Journal of Information and Technology

Extend the Ant Colony Optimization Algorithm for virtualization Technologies to improve the Resources utilization in On-Premises Datacenters

Jean Claude HABINSHUTI MUHUMUZA, Emmanuel BUGINGO & Célestin Tshimanga Kamanga

ISSN: 2617-3573

Stratford Peer Reviewed Journals and Book Publishing
Journal of Information Technology
Volume 8||Issue 1 ||Page 28-42||February |2024|
Email: info@stratfordjournals.org ISSN: 2617-3573

Extend the Ant Colony Optimization Algorithm for virtualization Technologies to improve the Resources utilization in On-Premises Datacenters

[1*]Jean Claude HABINSHUTI MUHUMUZA, [2]Emmanuel BUGINGO & [3]Célestin Tshimanga Kamanga

[1]Postgraduate Student, School of Computing, and Information Technology

University of Kigali, RWANDA

[2]Lecturer, School of Business and Economics, Department of Business and Information Technology, University of Rwanda, RWANDA

[3] School of Mathematics and Computer Sciences,

University of Kinshasa, Kinshasa, Democratic Republic of Congo

***How to cite this article*: MUHUMUZA, H.C.J., BUGINGO, E., & Kamanga, T.C. (2023). Extend the Ant Colony Optimization Algorithm for Virtualization Technologies to Improve the Resources Utilization in On-Premises Datacenters. *Journal of Information Technology,* 8(*1*), 28–42.**
https://doi.org/10.53819/81018102t7000

Abstract

Over the past few decades, there has been an increasing demand for computational power, which has fueled the growth of on-premises data centers. In recent years, virtualization techniques have been introduced to enhance data center resource utilization. These techniques consolidate multiple workloads onto fewer servers, reducing the need for physical devices to support an organization's IT infrastructure. Virtualization technologies have increased IT agility by allowing for quicker deployment of virtual machines (VMs), which in turn facilitates faster application and service rollouts, improves disaster recovery capabilities, and reduces carbon emissions, leading to significant cost savings for organizations. In this paper, we enhance the Ant Colony Optimization Algorithm (ACO) by applying it to virtualization. We simulate the ACO for virtual machine resource management. Our evaluation results demonstrate that the proposed algorithm can further improve resource utilization and reduce carbon emissions.

Keywords: *Virtualization, Datacenter, Resource utilization, Ant Colony Optimization.*

1.0 Introduction

Virtualization technologies aim to improve the utilization of Datacenter resources by allowing multiple Virtual machines (VMs) to share physical resources such as servers, storages, and networks. A VM is an isolated software container with operating system and applications. VMs allow organizations to run different operating systems and different applications independently by maximizing infrastructure resources such as Central Processing Units (CPUs), Radom Access Memory (RAM) while MINIMIZING the cost. Virtualization is nothing more than an increasingly efficient use of existing resources that delivers huge cost savings in a brief amount of time [1].
While virtualization allows us to better manage multiple workloads, it introduces considerable challenges to use it effectively. One of the concerns is the optimization of resources such as CPU and RAM while minimizing the Energy (Power), and in turn reducing the overall maintenance cost. The aim of this research is to extend the Ant Colony Optimization Algorithm by considering the Energy consumed by the physical host to enable VM placement and optimize the resource consumption.

1.1 Statement of the problem

Virtualization is the creation of virtual rather than actual version of something such as Operating System (OS), a Server, Storage device or network resources [2]. Virtualization is an increasing efficient as of the existing resources that delivers huge cost savings in a brief amount of time. It offers modern organizations new models of application deployment for greater uptime to meet the user's expectation or customer's needs. Virtualization provides new services in minutes instead of days, provides load balancing and scalability without downtime [3]. Traditional optimization techniques are sometimes insufficient to handle the huge demand of modern organizations, particularly in sectors like banking and finance where IT infrastructure plays a critical role in ensuring business continuity and meeting customer expectations.
We extended the ACO algorithm to optimize the utilization of Datacenter resources. The key challenge is to optimize the usage of the existing server resources such as CPU, RAM and Power with the goal of minimizing costs, increase efficiency and improve the overall performance of the Datacenter infrastructure. By deploying virtual machines, containers and other virtualization technologies, Datacenters can achieve greater flexibility, scalability, and utilization of resources. This can lead to significant cost savings as fewer physical servers will be needed to support the same workload.
Given the complexity and role of many organizations such as financial institutions or Banks, there is a clear demand upon IT for Improvement and enhancement of their Operational effectiveness and flexibility. Therefore, there is a huge demand and reliance on IT infrastructure to be able to achieve their objectives.
Financial institutions such as Banks understand that the IT infrastructure is increasingly critical to the services they provide, and their unavailability could have significant impact on the business and their customers. To ensure their business continuity, it is paramount that IT infrastructures are operating efficiently. The existing research have been focusing on developing models for cloud environments. Therefore, extending ACO algorithm for virtualization to improve the Datacenter resource utilization on-Premises Datacenters is very crucial to be able to adapt to the real-time demand for financial institutions such as Banks.

https://doi.org/10.53819/81018102t7000

1.2 Objective of the Study

This research aimed to extend the Ant Colony Optimization (ACO) algorithm to optimize resource utilization in on-premises Datacenters, specifically targeting the needs of financial institutions such as banks. The study sought to enhance the efficiency, scalability, and reliability of on-premises IT infrastructure, enabling banks to meet the evolving demands of their customers while minimizing operational costs and ensuring business continuity.

2.0 Literature Review

Virtualization is changing almost every aspect of how we manage systems, storage, networks, security, operating systems, and applications. A Datacenter may comprise of ten, hundreds or thousands of servers, storage systems, and network equipment. The amounts of computation power contained in these systems results in many interesting distributed systems and resource management problems. Modern Datacenters are designed to enable agile service-oriented IT models that are essential for success in the digital economy. The deployment of the virtualization helps to increase the agility of IT operations, enable faster deployment of new applications and services, and enhance the disaster recovery capabilities. In recent years virtualization has gained popularity in many different areas such as server consolidation, information security and cloud computing due to an increase in hardware performance about ten-fold in the past decade and the goal to reduce capital and operational costs within the Datacenter [4]. The resource utilization refers to the efficient use of resources such as CPU, memory, storage, and network bandwidth. Traditionally, Datacenters have re-lied on physical servers, storage devices, and networking equipment to run their workloads. However, this approach has resulted in underutilized hardware resources as each server typically runs a single workload. For example, the server virtualization enables multiple virtual servers to run on a single physical server which can im-prove CPU and memory utilization. Storage virtualization can enable the creation of a single pool of storage utilization. Network virtualization enables the creation of multiple logical networks that can be used traffic between different workloads which result in improvement of network bandwidth utilization [5]. To minimize the cost, these resources (CPU, RAM) should be made available to applications only as needed and not allocated statically based on the peak workload demand [6]. In [7], the Authors have proposed a new energy-aware approach based on the online bin-packing algorithm to improve the energy efficiency and resource utilization in cloud Datacenters. they have presented an over-provision method to deal with the varying resource demands from users. in [8], the authors have studied the problem of reduced data center resource utilization and increased network latency due to some virtual machines that are shutdown. They have proposed a network-aware virtual machine placement strategy to improve the overall network communication performance and user service quality through periodic virtual machine migration. the challenges are that the virtual machine migration process requires a large amount of network bandwidth, and the resources overheads on the physical machine where it is located increase, which in return will reduce the performance of other virtual machines. Even though virtualization helps with server consolidation and provide flexibility in resource management, it also brings new challenges. To determine where to run the applications in a shared environment remains a challenge due to the variable virtualization overheads seen by different applications and systems. Virtual Machine Migration methods are the foundation for managing computing resources, reducing performance overhead, saving energy, and balancing loads in cloud computing [9]. In [10], the authors have developed an ant colony-based algorithm which they combined with some local search approaches

https://doi.org/10.53819/81018102t7000

to minimize the number of active physical servers. They have demonstrated that the proposed algorithm will be able to combine virtual machines that are candidates together and will decrease the number of active utilized physical servers. To prevent overloading of physical machines, the same technique is applied in on-premises data centers. During the maintenance of physical machines, VMs can be transferred to another physical machine. VMs can also be migrated from a failing physical machine to a stable one, and idle VMs can be moved to another physical machine to optimize resource utilization. Live migration refers to the process of transferring a virtual machine from one physical server to another without causing any downtime [11].

Ant colony optimization (ACO) is a population-based algorithm that can be used to find approximate answers to complex optimization problems [12]. in ACO a set of software's agents called artificial ants search for good solutions to a given optimization problem. ACO can be explained using the Traveling Salesman Problem (TSP), where in TSP a set of locations and the distance between them are given. The problem consists of finding the closest tour of minimal length that visits each city once and only once. The idea of ACO is to simulate the behavior of ant colonies. When ants try to search for food, they use a chemical pheromone to communicate each other. Once they find the food source, they leave a pheromone in the path so that others can sense the pheromone on the ground until they find the food. As this process continues, most of the ants attract to choose the shortest path as there have been a huge amount of pheromones accumulated on this path [13]. The main reason behind the idea of im-proving the ACO algorithm is to be able to address the challenges faced in virtualization with the aim to improve resource utilization, reduce the cost and reduce the management overheads.

The literature review reveals that numerous efforts have been made to enhance resource utilization in data centers, yet challenges persist. Various approaches have been proposed to address this issue. The following is a summary of our main contribution:

1. An improved Ant Colony Optimization algorithm by integrating the Energy consumed by the physical server for effective resource allocation.
2. Examine the efficiency of the proposed ACO on two different On-Premises Datacenters with different physical servers.

3.0 Research Methodology

This section presents an overview of ant colony optimization algorithm methods.

3.1 ACO Algorithm to improve Virtual resource usage

The ACO Algorithm is a type of random search that uses groups of initial solutions that are put together at random to find solutions that are close to the best [14]. There are two different parts to the whole process: the adaptation phase and the planning phase. During the adaptation phase, each solution is mostly changed and adjusted by the way information changes during the evolution process. During the coordination phase, the information sharing method between each solution is mostly used to get the solution that is close to the best. Our research focuses on how we can improve the resource utilization on virtualized environment using the ACO algorithm.

https://doi.org/10.53819/81018102t7000

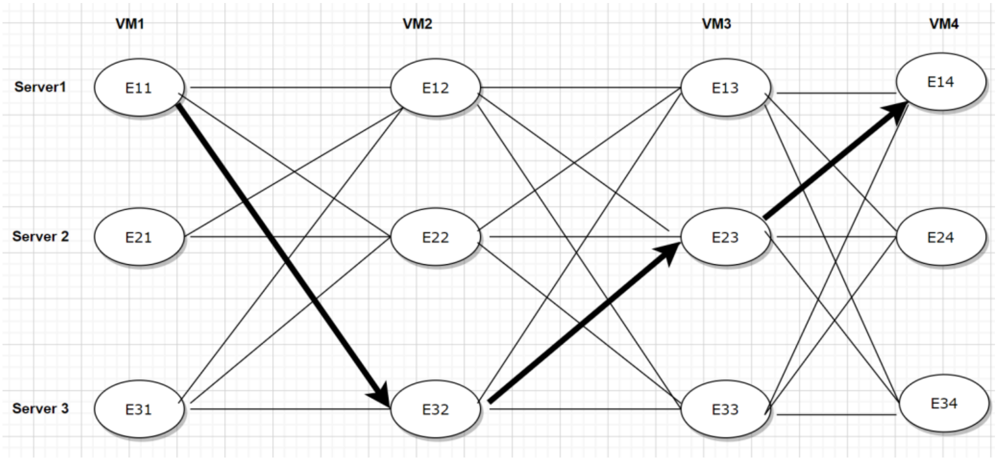


Figure 1. Challenges with virtual machine allocation

The ants construct solutions by moving through the search space and leaving the pheromone in the path. They build solutions by using a probabilistic rule using the formula below:

$$P_{(i,j)} = \frac{[\mathbf{T}_{(i,j)}{}^{\alpha} * \boldsymbol{\eta}_{(i,j)}{}^{\beta}]}{\sum[\mathbf{T}_{(i,k)}{}^{\alpha} * \boldsymbol{\eta}_{(i,k)}{}^{\beta}]} \quad (1)$$

Whereby i is the current node, j is the next node to be visited, T(i,j) is the pheromone trail between node i and node j, η(i,j) is the heuristic information that represents how the ant will move from node i to node j , α and β are the parameters that control the relative importance of the pheromone trails (levels) and the heuristic information in the decision-making process. To evaluate the solution, we consider the pheromone deposited between node i and node j (Δ T(i,j)) which is calculated by:

$$\Delta\mathbf{T}_{(i,j)} = \frac{C}{L} \quad (2)$$

Where C is defined as a constant that represents the amount of pheromone deposited by an ant and L to be the length of the solution found by the ant. Therefore, the pheromone trail will be updated based on the amount of pheromone deposited by the ant using the formula below:

$$\mathbf{T}_{(i,j)} = [(1 - \varrho) * \mathbf{T}_{(i,j)}] + [\varrho * \Delta\,\mathbf{T}_{(i,j)}] \quad (3)$$

Where ϱ is the evaporation rate.
In our improved Model for ACO Algorithm, we will integrate the Energy (Power) consumed by the physical server hosting the VMs. To be able to create a model for ACO in virtual machine resource utilization, let consider cluster of V virtual machines and a set of S physical servers. Let consider bij a binary variable which indicates whether VM j is allocated to a Server i, ri,j to be the resource utilization cost (CPU or RAM usage) for allocating VM j to Server i. P(i,j) is the pheromone level connecting the Server i and the VM j.
The same formula for ACO is used to calculate the probability P(i,j) of allocating the VM j to the Server i.

https://doi.org/10.53819/81018102t7000

Where:
T(i,j) = Pheromone level on the edge connecting Server i and VM j
η(i,j) = the inverse of the resource utilization cost of allocating VM j and Server i
α and β are the parameters that control the relative importance of the pheromone levels and the resource utilization cost.
On the end, the resource utilization Ri for each server will be calculated using the formula below:

$$\mathbf{R_i = \sum [r_{(i,j)} * b_{(i,j)}]} \quad (4)$$

Where r(i,j) is the resource utilization cost of allocating the VM j to the Server i and b(i,j) be in a binary variable.
The pheromone level will be calculated using the same formula:

$$\mathbf{T_{(i,j)} = (1 - \varrho) * T_{(i,j)} + \varrho * \Delta T_{(i,j)}} \quad (5)$$

Where:
Δ T(i,j) = Amount of pheromone deposited on the edge connecting the Server i and VM j and is calculated by:

$$\Delta \mathbf{T}_{(i,j)} = \frac{C}{\mathbf{R}_i} \quad (6)$$

Where C is a constant that controls the amount of the pheromone deposited and Ri be the resource utilization of the Server i.
The power consumed by the Virtual machines when they are idle in the active physical Server is a drawback and one of the major causes of resource wastage, therefore there might be a need to move it to another server to reserve the Energy.
Therefore, the Power consumption is calculated using the following formula:

$$\mathbf{P_{(u)} = (k * P_{max}) + [(1-k) * P_{max} * u]} \quad (7)$$

Where:
Pmax = Maximum Power consumed by a fully loaded Server
k = fraction power consumption in idle state
u = CPU utilization of the server

3.2 Energy Reduction for Ant-Colony Optimization (ER-ACO) algorithm

Our optimization goal is to adopt a virtualization technology which will allow the effective management of resources (CPU and RAM) and minimize the Energy consumed by On-Premises Datacenters which in turn will reduce the maintenance cost by effectively placing the VMs to the hosts with lower Energy consumption and with enough CPU, and Memory to accommodate the VM.

3.3 Proposed ER-ACO algorithm

Our proposed ER-ACO algorithm is shown in the Figure and table below and it can be stated as follows:
Input:
Virtual machine required resources Ri, Physical Machine resources S, Power utilized by the physical host P(u), the algorithm parameters α, β, ϱ

https://doi.org/10.53819/81018102t7000

Stratford Peer Reviewed Journals and Book Publishing
Journal of Information Technology
Volume 8||Issue 1 ||Page 28-42||February |2024|
Email: info@stratfordjournals.org ISSN: 2617-3573

Output:
The Virtual machine VMj mapped to the Physical host i, if P(u)< Pmax then the algorithm ends, otherwise, compute the Physical machine resources S and allocate the VMj to the Physical host i.

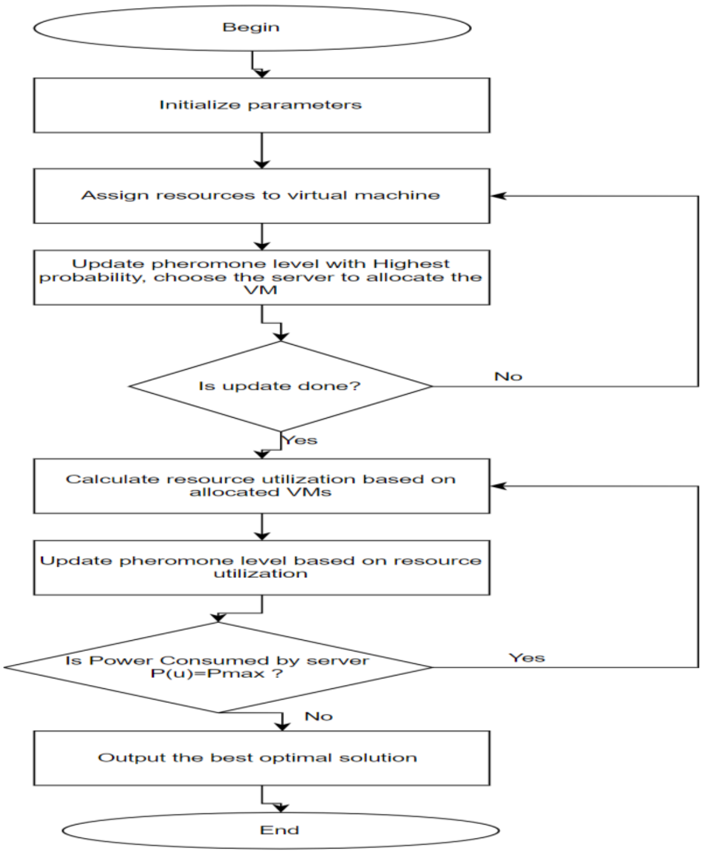


Figure 2. Proposed ER-ACO Algorithm flowchart

https://doi.org/10.53819/81018102t7000

Table 1. Proposed ER-ACO Algorithm

ALGORITHM1: ER-ACO ALGORITHM			
Input**: Virtual machine required resources **Ri**, Physical Machine resources S, Power utilized by the physical host **P(u)**, the algorithm parameters **α, β, ϱ			
Output:** The Virtual machine $\mathbf{VM_j}$ mapped to the Physical host **i			
***Initialization of Variables:** α, β, ϱ*			
for** VM_j=1 to N **do** //Deploy all unassigned VMs $\mathbf{VM_j}$ to the cluster host **i			
	*Assign resources **Ri** needed by the VM_j, $\mathbf{R_i} = \sum [\mathbf{r_{(i,j)}} * \mathbf{b_{(i,j)}}]$*		
	for** host **i** = 1 to **j do		
		Allocate VM_j to Host i	
		Calculate $P_{(u)} = (k * P_{max}) + [(1-k) * P_{max} * u]$	
		***if** $P_{(u)} = P_{max}$ **then** //Compute the resources of the remaining hosts S_i*	
			Allocate V_{Mj} to Host i+1
		end if	
	end for		
	update the pheromone $\mathbf{T_{(i,j)}} = (1 - \varrho) * \mathbf{T_{(i,j)}} + \varrho * \Delta \mathbf{T_{(i,j)}}$		
end for			
Return the best optimal solution //Allocate the VM to physical host			

4.0 Findings and Discussion

In this section, the simulation experiments are conducted to evaluate the performance of the proposed ER-ACO Algorithm.

4.1 Simulation Settings

The simulation experiments are conducted on Intel(R) Core (TM) i7-10750H CPU @ 2.60GHz (12 CPUs) using a Cloudsim platform named Workflowsim-1.0 and CloudReport. The hardware configuration of the experiment environment is based on Data collected from 2 Datacenters, 25 physical hosts with 4TB RAM and 1 TB of Storage on the primary Datacenter, and 12 physical hosts with 4TB and 1TB Storage in the secondary Datacenter. The system Architecture is x64, the Operating System is Windows, the Virtualization platform is Hyper-V. The computing capacity of physical servers is 1000 MIPS. The computing resources of the virtual machine is 1000 MIPS, RAM = 512 MB, CPUs =4. Different parameters in the simulation will have different results.

4.2 Simulation Results

The simulation is done using 30 VMs, the output from the CloudReport shows the following results:

https://doi.org/10.53819/81018102t7000

Stratford Peer Reviewed Journals and Book Publishing
Journal of Information Technology
Volume 8||Issue 1 ||Page 28-42||February |2024|
Email: info@stratfordjournals.org ISSN: 2617-3573

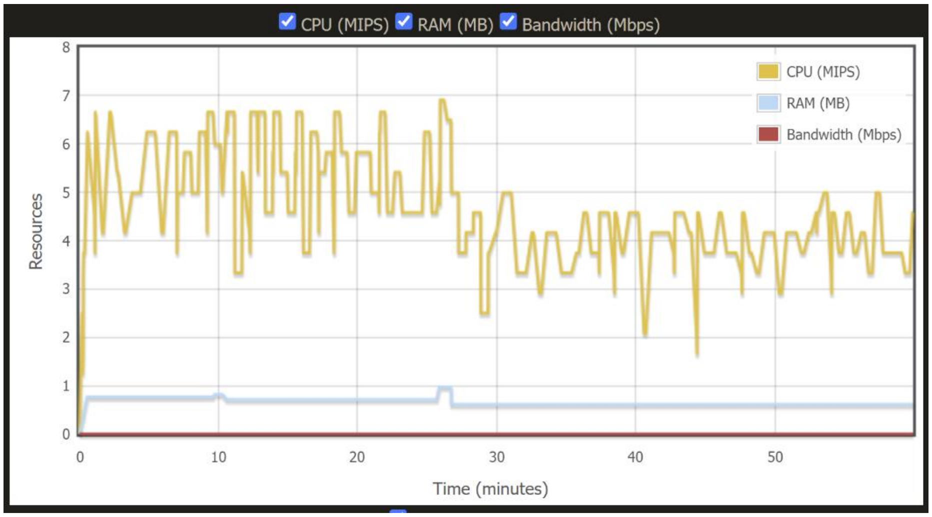


Figure 3. Overall resource utilization in Datacenter 1

Figure 3 shows the overall resource utilization in Datacenter 1 which has 25 hosts. It shows the CPU is highly in demand compared to other resources such as RAM and Network bandwidth. In the other hand, the Figure below shows the overall resource utilization for Datacenter 2 with 12 hosts.

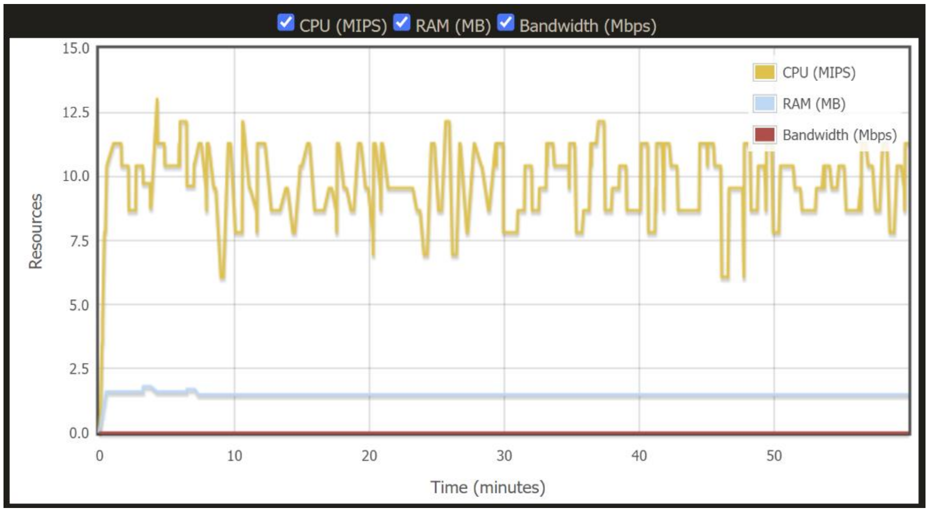


Figure 4. Overall resource utilization in Datacenter 2

The Figure below shows the overall Power consumption for Datacenter 1 for which has 25 hosts, and each host has maximum power utilization of 750 watts over a period of one hour.

https://doi.org/10.53819/81018102t7000

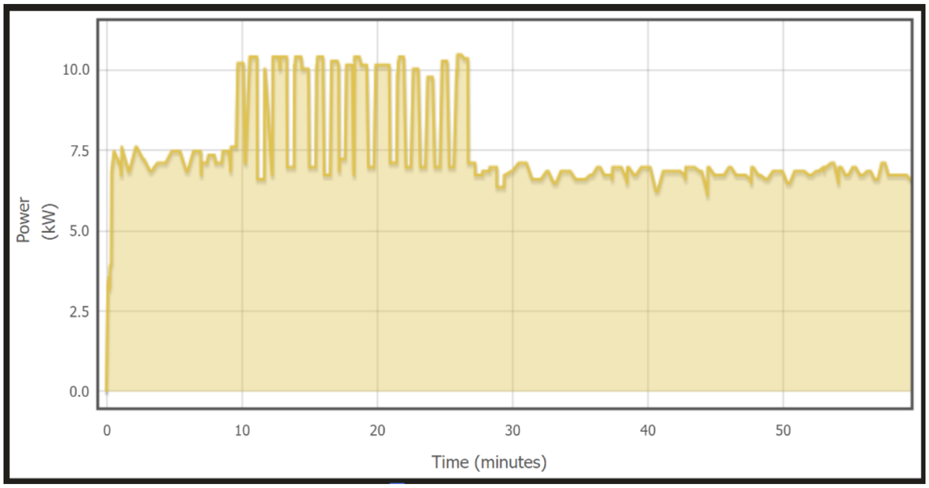


Figure 5. Overall Power Consumption for Datacenter 1

The figure below shows the overall power consumption for Datacenter 2 with 12 hosts each having power consumption of 750 watts over a period of one hour.

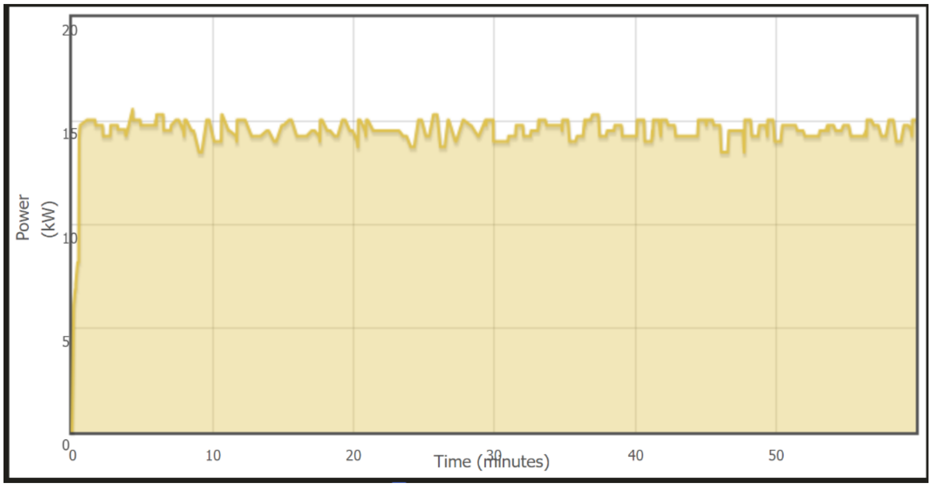


Figure 6. Overall Power Consumption for Datacenter 2

The result of the simulation shows that the number of VM migration in Datacenter 1 is 8; the first migration is related to distribution whereby the VM moved from Host0 to Host2 in 9.7 minutes where the source host (Host0) was consuming around 98% of Power and the destination host (Host2) was consuming 0.0% of power. Other migrations details are in the table below:

https://doi.org/10.53819/81018102t7000

Table 2. VM migration in Datacenter 1

Number of migrations: 8
Migration 0: Description: Distribution VM4-0 from Host0 to Host2 at 9.708333333333334 minutes. Source host was consuming 90.0% of CPU, 409600 GB of RAM and 98.125% of power. Target host was consuming 0.0% of CPU, 0 GB of RAM and 0.0% of power.
Migration 1: Description: Consolidation VM4-2 from Host0 to Host1 at 25.857666666666667 minutes. Source host was consuming 50.0% of CPU, 358400 GB of RAM and 85.625% of power. Target host was consuming 30.0% of CPU, 307200 GB of RAM and 79.375% of power.
Migration 2: Description: Consolidation VM4-4 from Host0 to Host1 at 25.857666666666667 minutes. Source host was consuming 50% of CPU, 358400 GB of RAM and 85.625% of power. Target host was consuming 30% of CPU, 30720 GB of RAM and 79.375% of power.
Migration 3: Description: Consolidation VM4-6 from Host0 to Host1 at 25.857666666666667 minutes. Source host was consuming 50% of CPU, 358400 GB of RAM and 85.625% of power. Target host was consuming 30% of CPU, 30720 GB of RAM and 79.375% of power.
Migration 4: Description: Consolidation VM4-8 from Host0 to Host1 at 25.857666666666667 minutes. Source host was consuming 50% of CPU, 358400 GB of RAM and 85.625% of power. Target host was consuming 30% of CPU, 307200 GB of RAM and 79.375% of power.
Migration 5: Description: Consolidation VM4-10 from Host0 to Host2 at 25.857666666666667 minutes. Source host was consuming 50% of CPU, 358400 GB of RAM and 85.625% of power. Target host was consuming 40% of CPU, 51200 GB of RAM and 82.5% of power.
Migration 6: Description: Consolidation VM4-12 from Host0 to Host2 at 25.857666666666667 minutes. Source host was consuming 50% of CPU, 358400 GB of RAM and 85.625% of power. Target host was consuming 40% of CPU, 51200 GB of RAM and 82.5% of power.
Migration 7: Description: Consolidation VM4-14 from Host0 to Host2 at 25.857666666666667 minutes. Source host was consuming 50% of CPU, 358400 GB of RAM and 85.625% of power. Target host was consuming 40% of CPU, 51200 GB of RAM and 82.5% of power.

On the other hand, The Simulation has shown the migration of only 3 VMs in Datacenter 2 to be able to effectively balance the load as shown in the table below:

https://doi.org/10.53819/81018102t7000

Stratford Peer Reviewed Journals and Book Publishing
Journal of Information Technology
Volume 8||Issue 1 ||Page 28-42||February |2024|
Email: info@stratfordjournals.org ISSN: 2617-3573

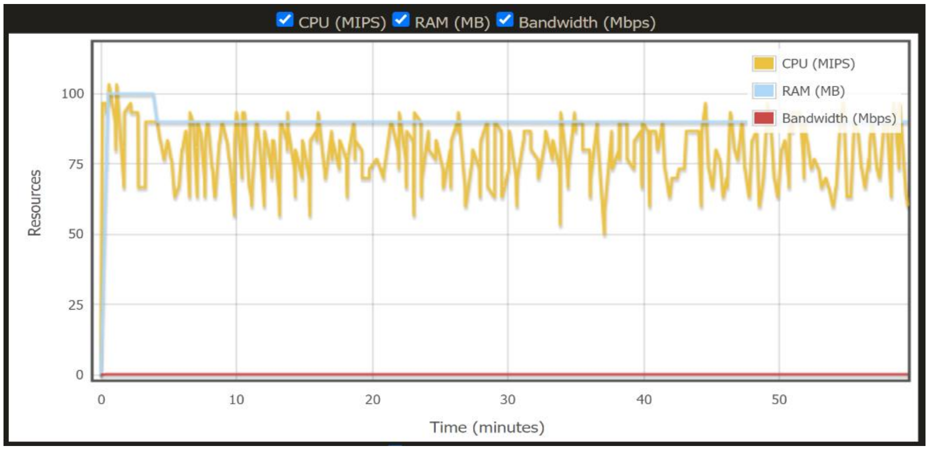
Table 3. VM migration in Datacenter 2

Number of migrations: 3
Migration 0: Description: Distribution VM4-1 from Host0 to Host1 at 3.275 minutes. Source host was consuming 90% of CPU, 563200 GB of RAM and 98.125% of power. Target host was consuming 30% of CPU, 204800 GB of RAM and 79.375% of power.
Migration 1: Description: Distribution VM4-3 from Host0 to Host1 at 3.275 minutes. Source host was consuming 90% of CPU, 563200 GB of RAM and 98.125% of power. Target host was consuming 30% of CPU, 204800 GB of RAM and 79.375% of power.
Migration 2: Description: Distribution VM4-5 from Host0 to Host1 at 6.508333333333334 minutes. Source host was consuming 80% of CPU, 460800 GB of RAM and 95.0% of power. Target host was consuming 30% of CPU, 307200 GB of RAM and 79.375% of power.

This demonstrates that if the ER-ACO Algorithm can be incorporated into on-premises virtualization, there will be an improvement in both the speed of VM Allocation and the efficiency of resource management. a good VM placement scheme decreases responding time of applications and energy consumption [15].

4.3 Performance of the existing ACO algorithm compared to the proposed ER-ACO algorithm

The purpose of this research is to improve Datacenter Resource utilization by reducing Cost and Energy consumption, more efficiently managing CPU and RAM in On-premises Datacenters, and optimizing the amount of time it takes for tasks to be completed. Simulating the existing ACO algorithm without considering the Energy consumption for example, we are getting different results as shown in Figure below where the CPU and RAM are remaining constantly high during the simulation thus more resources consumed.



https://doi.org/10.53819/81018102t7000

Figure 7. Overall resource consumption of 30 VMs in both 2 Datacenters without considering Power (with the existing ACO algorithm)

The results of the simulation where the power is not considered shows that per Data-center site, only 1 migration occurred as shown in Table below where VM4-0 moved from Host0 to Host2, and the migration time was 52 minutes. And the source host was consuming around 93.7% of its Power, meaning that the source Host was degrading.

Table 4. VM migration in Datacenter 1without considering the Power

Number of migrations: 1
Migration 0: Description: Distribution VM4-0 from Host0 to Host2 at 52.34166666666667 minutes. Source host was consuming 90% of CPU, 409600 GB of RAM and 93.75% of power. Target host was consuming 0.0% of CPU, 0.0% of RAM and 0.0% of power.

The table below depicts the migration happening at Datacenter 2 where only one VM4-1 migrated from Host0 to Host1 and took 3.2 minutes.

Table 5: VM migration in Datacenter 2 without considering the Power

Number of migrations: 1
Migration 0: Description: Distribution VM4-1 from Host0 to Host1 at 3.275 minutes. Source host was consuming 80% of CPU, 563200 GB of RAM and 83.33% of power. Target host was consuming 40% of CPU, 204800 GB of RAM and 41.67% of power.

The ER-ACO algorithm presented in Algorithm 1 is designed to reduce the Energy reduce the cost and maximize the overall resource utilization. We compared the existing basic ACO Algorithm, with the proposed ER-ACO algorithm, we found that the proposed ER-ACO algorithm is more efficient. The evaluation results are presented in the Figure 3, 4 and 5. The results achieved by our approach can be summarized as follows: For both Datacenters, there is a decrease in energy which would impact on the cost as the migration of the VM from the host helps to reduce the resource consumption, Therefore, the VMs can be distributed to other hosts to balance the load and this can impact positively the performance of the systems. In the end, the pro-posed Algorithm is beneficial as the number of virtual machines that needed to mi-grate is more resulting in better management of resources and the time it takes for the virtual machine migration is less. Unlike the existing ACO algorithm, the results (Figure 7) showed that there is no much reduction change on resources as the graph shows that the resources are still high and keep constant (90%). Therefore, the ER-ACO Algorithm that is proposed shows the potential to be effective in optimizing the resources of an On-Premises Datacenter.

5.0 Conclusion

In this study, the focus was on VM migration to minimize energy consumption, which, in turn, was anticipated to reduce maintenance costs in on-premises data centers. The ER-ACO Algorithm was introduced to optimize the performance of both physical hosts and VMs. To assess its

https://doi.org/10.53819/81018102t7000

effectiveness, simulation experiments were conducted, observing VM migration and resource consumption of physical hosts. Numerous algorithms have been developed to enhance resource utilization in cloud data centers. There was a recognized need to implement these algorithms in on-premises data centers to increase their efficiencies. Therefore, adopting and implementing the ER-ACO Algorithm to extend the use of virtualization technology was found to significantly improve the optimization of on-premises data center resources by enhancing the overall performance of data center infrastructures.

6.0 Acknowledgement

Words cannot express my gratitude to all those who have contributed to the completion of this research paper.
First and foremost, I extend my deepest appreciation to my supervisor, Dr. Emmanuel BUGINGO, for his unwavering support, invaluable guidance, and constructive feed-back throughout the entire research process. his expertise and encouragement significantly shaped the direction and quality of the work.
I want to acknowledge the University of Kigali Management for providing the necessary resources and knowledge that facilitated the successful execution of this re-search.
Many thanks to my family and friends for their unwavering support, understanding, and encouragement during the ups and downs of this research journey. For that I am truly grateful.

REFERENCES

1. Matthew P. (2016). Virtualization essentials (2nd ed). USA. John Wiley & Sons, Inc.
2. Kate B., Brian K. (2021). *Virtualization*. https://www.techtarget.com/searchitoperations/definition/virtualization
3. Matthew P. (2012). Virtualization essentials. USA. John Wiley & Sons, Inc.
4. Theresa V.S. (2018). Data center modernization for dummies, (VMware special edition). USA. John Wiley & Sons, Inc.
5. Samjhana R., Zinnia S. (2014). Comparative Performance Analysis of Virtualization Technologies in Cloud Computing. International Journal of Engineering Research & Technolo-gy. Vol. 3 Issue 9, September-2014.
6. Kate B., Brian K. (2021). Virtualization. https://www.techtarget.com/searchitoperations/definition/virtualization, last accessed 2023/9/12.
7. Xuwei X., Fulong Y., Kristif P., Fu W., Bitao P., Xiaotao G., Shaojuan Z., & Nicola C. (2019). A Reconfigurable and cost-effective architecture for high-performance optical data center networks. Journal of Lightwave Technology 10.1109/JLT.2020.3002735
8. Kusic,D., Kephart, J.O., Hanson, J.E., Kandasamy,N., Jiang,G. (2009). Power and Per-formance Management of Virtualized Computing Environments via Look ahead Control. Clustercomputing,12(1):1–15. https://doi.org/10.1007/s10586-008-0070-y
9. Wang, X., Liu, Z.: An energy-aware VMs placement algorithm in cloud computing envi-ronment. In: 2012 Second International Conference on Intelligent System Design and En-gineering Application (ISDEA), pp. 627–630. IEEE (2012). https://doi.org/10.1109/ISdea.2012.467
10. G. Y. Luo, Z. Z. Qian, and S. L. Lu, A network-aware VM re-scheduling algorithm, Chin. J. Comput., vol. 38, pp. 932–943, Jan. 2015.

https://doi.org/10.53819/81018102t7000

11. Muhammad Z. N., Syed A. H. (2019). Advances in computers, Vol. 112, Academic Press.
12. X.-F. Liu, Z.-H. Zhan, J. D. Deng, Y. Li, T. L. Gu, and J. Zhang, An energy efficient ant colony system for virtual machine placement in cloud computing, IEEE Trans. Evol. Comput., vol. 22, no. 1, pp. 113–128, Feb. 2018. https://doi.org/10.1109/TEVC.2016.2623803
13. Christopher C., Keir F., Steven H., Jacob Gorm H., Eric J., Christian L., Ian P., Andrew W. (2005). Live migration of virtual machines (273-285).
14. Dorigo, M. (n.d.). (2007). Ant Colony Optimization from Scholarpedia. https://doi.org/10.4249/scholarpedia.1461
15. Tawfeek, M. A., El-Sisi, A. B., Keshk, A. E., & Torkey, F. A. (2014). CCIS 488 - Virtual Machine Placement Based on Ant Colony Optimization for Minimizing Resource Wastage. In CCIS (Vol. 488). https://doi.org/10.1007/978-3-319-13461-1_16

https://doi.org/10.53819/81018102t7000