International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Resource Offloading (Balancing) in Cloud Environment using Particle Swarm Optimization and Improved Particle Swarm Optimization on Xen Server

[1*]Akash Dave, [2]Dr. Hetal Chudasama

Submitted:10/03/2024 **Revised**: 25/04/2024 **Accepted**: 02/05/2024

Abstract: Cloud has numerous strong servers to allot immense solicitation of clients. Load Balancing (Resource Offloading) is a strategy to disperse tasks on numerous VM's of Server to accomplish Asset usage, Reduce Reaction (cost) time and keep away from trouble. Resource offloading is a crucial factor in ensuring that the available resources are utilized efficiently, and the workload is distributed optimally. This paper presents novel research of resource offloading in a cloud environment and explores the different approaches to resource offloading, including Resource offloading, and Response time reduction of VM. In this article, A PSO & Improved PSO, has been introduced to track down the better arrangement for the issue of Allotment Resources (Assets) and load adjusting in CC. This Work was probed Xen Server and aftereffect of the Proposed Calculation Further developed PSO was extremely uplifting. Altogether the aftereffect of IPSO contrasted and PSO Calculation.

Keywords: *Cloud Environment, Particle Swarm Optimization, Improved Particle Swarm Optimization, Type 1 Hypervisor, Xen Server.*

1. Introduction

Cloud computing (CC) has arisen as a clever pattern in recent years [1,2,9]. It has prompted the movement of appropriated frameworks to an enormous scope of computing organizations. Cloud Computing intends to give calculations and assets over the web using dynamic provisioning of administrations. There are a few difficulties and issues related to the execution of Cloud Computing [2,9].

Factors of Cloud Computing [2,9]

- Strengthen Space Capacities
- Network Performance Improved
- Customizable Applications go forward.
- IoT inside Cloud Technology
- Information also reveals how Future Changes
- Greater Cloud hosting
- Secure
- Customizable Applications
- Financial
- Serverless Computation
- Elasticity & Scalability

Challenges of Cloud Computing [1,2,9]

- Protection of data
- Data retrieval and availability issue
- Load Balancing
- Execution monitoring
- Fault tolerance
- Migration
- Interoperability
- Portability

Load adjusting is one of the most basic and significant issue in cloud; Numerous VM's are utilized in the Networks so it is hard to dispense assets in cloud physically on the off chance that there are various solicitations all at once from modified clients for asset use, then, at that point, it makes a long queue up on server and builds the reaction time [5,6,7,9].

A fine load adjusting calculation is important that will assist with offering assets to all clients to utilize every virtual machine and ensure that there ought not be any more or lower Use machines [6,7,9].

1.1. Contribution

Motivated by the above analysis, to make full use of the advantages of cloud in computation offloading whilst to the tackle all those challenges, in this paper we discuss a general scenario with multiusers, 5 VMs and a remote cloud server.

[1*]*Ph.D. Scholar, Department of Computer Engineering, Madhuben and Bhanubhai Patel Institute of Technology, New Vallabh Vidhyanagar, Charutar Vidya Mandal University (CVMU),* India.
[2]*Associate Professor, Department of Information Technology, G H Patel College of Engineering and Technology, Vallabh Vidhyanagar, Charutar Vidya Mandal University (CVMU),* India.

[*]*Corresponding Author Email Id:* akashdave46@gmail.com

4 VMs are generated with 5 Applications are used for the analysis.

2. Related Work

In this examination, Xen-Server was utilized as a virtualization stage [5,6,7]. In Xen-Server, there are different settings for computer processor and memory portion [2,3]. Various types of utilizations might require different portion techniques to ideally work [6]. To find which asset designations are reasonable for explicit circumstances, tests and benchmarks are required [5,6,7]. Considering the calculation arranged model, expresses that there are three principal equipment elements to distributed computing server farms: CPU, RAM, and Hard Drives [7]. Accordingly, tests utilizing applications that are computer chip serious, memory escalated, and circle concentrated are fundamental. This investigation is being conducted in order to run benchmarks and identify the proper central processor and memory sections for optimal execution, considering use [1,2,3,4].

The instruments are utilized to divide the sum between VM with the assistance of a VM director [1,7]. Relocation of VM among VM Hosts one of the strategies for load equilibrium of VM Hosts which is to move the responsibility of VM's starting with one over-burden Virtual Machine Host then onto the next Virtual Machine Host attempting to take the responsibility of all Virtual Machine Host's equitably [5,6].Three periods of work are engaged with this cycle: discovery, choice, and activity. The recognition stage isto distinguish if a lopsidedness happens in a working ranch [1,2,3,4].

Hence, the responsibility of VMHs isn't static; the movement of VM might cause the responsibility of the objective VMH to be much more regrettable [6]. Subsequently, a powerful burden adjusting component is fundamental yet hard for the administration of VM Hosts [1,9].

One of the fundamental issues is load adjusting, which experiences numerous issues like the untimely combination, decreased union speed, from the outset picked arbitrary arrangements, and being stuck in localoptima [1,6,8,9].

Load balance is a regular yet fundamental exploration subject in equal or appropriated frameworks, and many AI-based strategies have been proposed [1, 2]. The accompanying talks about some new techniques for load adjusting in cloud computing [9].

Virtualization is a procedure that permits running diverse working frameworks (OS) together on one actualmachine and these OS are isolated from one another and the hidden actual framework through a unique middleware reflection called a VM (Virtual Machine) [5,6,7,8,9].

LB is the technique for expanding framework execution by moving the gadget trouble [6].

LB technique [6,9]

- Adaptability
- Throughput
- Scalable
- Response time
- Usability
- Availability
- Associated overheads.

Challenges of LB [6,9]

- Distributed Geographical Nodes
- Single Point of Failure
- VM Migration
- Nodes Heterogeneity
- Handling Data
- LB Scalability
- Complexity of Algorithm
- Automated Service Provisioning
- Energy Management

In [3], The Many-Objective PSO with Cooperative Agents algorithm is a novel algorithm that the author introduces. This algorithm makes two distinct uses of the Inverted Generational Distance (IGD) indicator. First, to determine the best solution in terms of convergence and diversity, IGD is employed as a leader selection technique. It is also used as an archival technique to identify the non-dominated solutions retained in a restricted archive. The algorithm's output demonstrates that the issue of numerous non-dominant solutions in Maops is resolved by IGD-based selection. Several experimental studies involving up to twenty objectives and 110 benchmark test cases verify the effectiveness of the proposed algorithm.

In [4], An Improved PSO algorithm is suggested in [4] to address task scheduling issues in cloud computing settings. Several adaptive learning techniques are used to shorten the original POS algorithm's execution time. During the first population phase, two types of particles are defined by the proposed Multi Adaptive Learning for PSO: ordinary particles and local best particles. In this phase, the likelihood of reaching a local optimum rise while the diversity of the population decreases. Four criteria are used to compare the proposed approach with different algorithms: time span, load balancing, stability, and efficiency.

In [6], the authors present a PSO algorithm to provide a better solution to the resource allocation and load balancing problems in CC. The proposed algorithm was experimented with Xen Server, and the results were very promising. The PSO algorithm's outcome is significant when compared to other balancing and comparison algorithms.

2.1 Open-Source Cloud Platforms – Review

Shows below that in table 1 Open-Source Cloud Platforms Comparison

Table 1: Open-Source Cloud Platforms Comparison

Parameters	XEN [5,6,7,8]	Open Stack [8]	Open Nebula[8]	Delta [8]
Tool DevelopedBy	Citrix Xenserver	Rackspace and NASA	Open Nebula Systems, OpenNebula Community	red hat in 2009 now Apache incubator project.
FacilitiesProvidedis.	Virtualization Platform	Open-Source Cloud Platform	compute (Nova), storage (Swift), image (Glance).	Allows A virtualized cloud Environment.
Written inlanguage.	Python, C	Python	Java and Ruby	Ruby, java, python
Service Providedis.	IAAS	IAAS	IAAS	IAAS
The hypervisorused is.	Xen	Xen, HyperV, KVM,Xen Server, VMWare,L XC	Xen, KVM, Hyper-V,VMWare	KVM, Xen
Cloud Implemented by which tool.	Public and private	Public and private	Public, private,Hybrid	Public and Private
AvailableThrough is.	Nimbus, Eucalyptus &Amazon EC2	Amazon EC2, S3	Amazon EC2	Available through AmazonEC2
GUI Available?	Command- line interface(CLI) ONLY	YES	YES	YES
which Licenseis needed?	GPL2	(ALV)Apache License Version2.0	ALV 2.0	ALV 2.0
Users Can Do	Done Simulation of Cloud Environment	Can be able to Build private &public clouds	Can Providescloud computing. services &resources management	View image status and statistics across clouds, Also, Use services likeHorizon. Done Simulation of Cloud Environment
OS Supported is.	Most Linux Distributionsand windows	Most Linux distribution	Most Linux distributions	Most Linux distribution

3. Proposed Work

3.1 Allocations of CPU in XenServer

XenServer permits the execution of server and datacenter Virtually [5]. Now and again, the default arrangement should be altered to allow VM assets and accordingly upgrade processor utilization. There are two kinds of computer processor asset designation in XenServer [7]. The first is the PSO asset designation, which allows a specific virtual machine to have more processing time than other virtual machines. The second is IPSO cap, which uses the maximum amount of CPU power that a virtual machine can have. [6].

3.1.1 Control of Memory in XenServer

It utilizes less power and unites server responsibilities, subsequently diminishing administration costs and establishing a well-disposed processing climate [6]. The first is the PSO asset designation, which allows a specific virtual machine to have more processing time than other virtual machines. The second is IPSO cap, which uses the maximum amount of CPU power that a virtual machine can have [5].

3.1.2 Benchmarks

In this examination, 4 programs were utilized separated from default Applications: [5]

1) stress-ng, 2) SysBench, 3) UnixBench, and 4) ApacheBench.

The stress-ng tool is designed to test various hardware subsystems and operating system kernel interfaces in Linux frameworks [5].

Customers can get an impression of the system performance using Sysbench, a benchmarking tool, without having to set up complicated benchmarks [5].

A set of benchmarks called Unixbench is used to evaluate the effectiveness of Unix-like systems. Compilers, operating systems, libraries, and hardware all have an impact on the benchmark's results [5].

Apachebench is a tool that lists web server hardware [5].

3.2. Issue Proclamation

Load (Balancing) Distribution is a major issue in CC and for that conveyed kind of arrangement is important. In any case, this technique isn't useful like clockwork and practical or cost adequate for supporting at least one dormant assistance to cover the required interest. Individual undertaking portion to meriting servers and clients isn't simple work. Subsequently, to successfully adjust the heap, load adjusting calculations are arranged into static and dynamic calculations.

3.3. Strategy

We demand the Cloud computing's goals which simply decrease Reaction Time as well as also cut the effect of the climate. This work is centered around the plan and execution of a programmed Asset Use. To serenely finish the examination points we will apply the Molecule Multitude Improvement calculation and IPSO in Xen Server [2,3,4,5].

3.4. Tools and Techniques

We will require different programming and Web access. The equipment necessities for this Environment are:

3.4.1. Minimum Required Hardware

1. Intel i5 or above & Any Generation with 3.00 or higher GHz Processor

2. 8 GB or above RAM

3. 500GB Minimum

3.4.2. Requirement of Softwares

1. Xen Server Above Version 7.0.0

2. Python above version 2.7

3.5. Algorithm Synopsis

PSO – This Algorithm is a heuristic optimization in view of swarm. Seen By the natural and social way of behaving of swarm insight [6]. A dispersed scanning strategy is used by birds to find food. There is always a bird searching for food that has a good sense of where to find food from the bird run. Additionally, the bird locates the area where all other birds are exchanging information about the food source and helps to approach it. What's more, the asset of food is generally ideal/agreeable arrangement [2,3,6].

IPSO - The original PSO algorithm has been widely used, but it may suffer from some issues such as premature convergence and low convergence speed. To address these issues, several improvements to the algorithm have been proposed. Here are some of the most used improvements: [4,5]

3.5.1. Velocity Clamping:

One of the issues with the original PSO algorithm is that the velocity of the particles can grow too large, leading to instability and divergence. Velocity clamping is a technique used to restrict the maximum velocity of the particles to a certain threshold value [5].

3.5.2. Inertia Weight:

Inertia weight is a boundary that controls the harmony among investigation and double-dealing in the pursuit cycle [5]. A high inertia weight value promotes exploration, while a low value promotes exploitation. The value of the inertia weight is dynamically adjusted during the search process to balance exploration and exploitation.

3.5.3. Constriction Factor:

Constriction factor is a parameter used to control the rate of convergence of the algorithm. Its foundation is the notion that the particle velocity should be limited by a factor that guarantees convergence. The constriction factor is dynamically adjusted during the search process to balance exploration and exploitation [5].

3.5.4. Neighborhood Topology:

The neighborhood topology defines the interactions between the particles in the swarm. Different neighborhood topologies have been proposed, such as the global neighborhood, ring neighborhood, and random neighborhood. The choice of neighborhood topology can essentially affect the exhibition of the calculation [5].

3.5.5. Hybridization:

PSO can be combined with other optimization techniques, such as Genetic Algorithm, Simulated Annealing, and Ant Colony Optimization, to improve its performance. The combination of PSO with other optimization techniques is called hybridization [5].

These improvements can significantly enhance the performance of the original PSO algorithm in terms of convergence speed, accuracy, and robustness. However, the choice of which improvement to use depends on the specific optimization problem being solved and the characteristics of the search space.

Algorithm Steps

PSO [3,6]

1. With arbitrary position and speeds Introduce particles populace.

2. Ascertain fitness capability an incentive for each molecule.

3. Utilizing Contrasting current particles fitness esteem and every molecule's fitness esteem track down Pbest and Gbest esteem

4. Analyze Local Best and Gloabal Best Qualities and Dole out the Assets to the best arrangement.

5. Update Fitness capability and populace for every one of the particles.

6. Check regardless of whether all the asset is dispensed, while possibly not then rehash from Step 2.

IPSO [4]

1. With arbitrary position and speeds Introduce particles populace.

2. Ascertain fitness capability an incentive for each molecule.

3. Utilizing Contrasting current particles fitness esteem and every molecule's fitness esteem track down Pbest and Gbest esteem by Multi Adaptive Learning for Particle Swarm Optimization

4. Analyze Local Best and Gloabal Best Qualities and Dole out the Assets to the best arrangement.

5. The diversity of the population is reduced and the probability of reaching the local optimum increases.

6. Update Fitness capability and populace for every one of the particles.

7. Check regardless of whether all the asset is dispensed, while possibly not then rehash from Step 2.

4. Implementation and Result

To affirm the proposed calculation, we lead an Examination with one server design of processor Intel i5 3.00 GHz with RAM 8 GB. In server we have introduced the Xen Server, On server VM is made having design of processor 1 Center, RAM 512 MB, and Storage 128 GB introduced operating system with ubuntu 12.04.

There are numerous assets accessible in the calculation we pick: computer processor Asset as it were. shows below in table 2 Notations and definitions

Table 2: Notations and definitions.

RAM= Random access memory	R_T = Response time
T = CPU utilization	RST = Request Submission time
k = Virtual Machine	RR = Request Reaction Time
Per = Percentage of load	T = Total available load
L= Generated Load	VM = Virtual Machine
CPU= central processing unit	CC = Cloud computing
IPSO=Improved Particle Swarm Optimization	PSO=Particle Swarm Optimization

To find different values we have created mathematical equations.

4.1. CPU Utilization

$$L_{(k)} = T_{(k)} / V_{(k)} \tag{1}$$

4.2. Response Time

$$R_T = RST_{(k)} - RR_{(k)} \tag{2}$$

Thus, we will presently have the heap of each VM in our data set. We have associated every one of the VM locally so there is no organization delay. XEN SERVER [2,3,4] gives all subtleties of the heap as indicated by central processor and Slam of VMs. Execution of calculation is in Python script. We utilize Various Applications to powerfully produce load. Beneath Table has the Estimations of the computer chip and Smash Use of 4 Unique Virtual Machines.

List Of Applications which are used for load generation and load analysis.

- Liber Office
- Firefox
- Gedit Text Editor

Note: Each time application is rearranged and afterward assigned.

Shows below that in table 3 Load Analysis of Different VM (Values are in average form)

Table 3. Load Analysis of Different VM (Values are in average form)

VM No.	VM 1		VM 2		VM 3		VM 4	
Application Name	Utilization of CPU	Utilization of RAM	Utilization of CPU	Utilization of RAM	Utilization of CPU	Utilization of RAM	Utilization of CPU	Utilization of RAM
Ideal State	1%	2%	1%	1%	1%	2%	2%	2%
Text Editor	10%	12%	13%	15%	14%	15%	17%	15%
Firefox	35%	27%	42%	39%	51%	49%	56%	55%
LibreOffice	69%	60%	72%	67%	75%	69%	78%	69%

To Calculate above percentage, we have used below formula.

$$Per(k) = T(k) / L(k) \quad (3)$$

In view of central processor (CPU) and RAM against heap of Diff. VMs We Assess Execution. Reaction time is chosen as metric. Reaction season of the applications running in VM PSO algorithm and IPSO algorithm is displayed in Figures 1,2,3,4 and shows below that in table 4,5,6,7 Response time analysis Of PSO for VM 1- VM 4.

Table 4: Response time analysis Of PSO for VM 1

VM Number	Algorithm Run	No. Of Application Executed at a time	No. of times algorithm runs	Response time (in Seconds)
1	PSO	1	1	1
		2	2	1.15
		3	3	1.37
		4	4	1.78
		5	5	1.83

Table 5: Response Time analysis Of PSO for VM 2

VM Number	Algorithm Run	No. Of Application Executed at a time	No. of times algorithm runs	Response time (in Seconds)
2	PSO	1	1	1.21
		2	2	1.25
		3	5	1.48
		4	6	1.80
		5	7	1.98

Table 6: Response Time analysis Of PSO for VM 3

VM Number	Algorithm Run	No. Of Application Executed at a time	No. of times algorithm runs	Response time (in Seconds)
3	PSO	1	1	1.25
		2	2	1.37
		3	3	1.58
		4	4	1.98
		5	5	2.31

Table 7: Response Time analysis Of PSO for VM 4

VM Number	Algorithm Run	No. Of Application Executed at a time	No. of times algorithm runs	Response time (in Seconds)
4	PSO	1	1	1.34
		2	2	1.42
		3	3	1.67
		4	4	1.99
		5	5	2.35

Shows below that in Table 8,9,10,11 Response Time analysis of IPSO for VM 1 – VM4

Table 8: Response Time analysis Of IPSO for VM 1

VM Number	Algorithm Run	No. Of Application Executed at a time	No. of times algorithm runs	Response time (in Seconds)
1	IPSO	1	1	1
		2	2	1.05
		3	3	1.27

		4	4	1.58
		5	5	1.48

Table 9: Response Time analysis Of IPSO for VM 2

VM Number	Algorithm Run	No. Of Application Executed at a time	No. of times algorithm runs	Response time (in Seconds)
2	IPSO	1	1	1.15
		2	2	1.10
		3	3	1.28
		4	4	1.41
		5	5	1.78

Table 10: Response Time analysis Of IPSO for VM 3

VM Number	Algorithm Run	No. Of Application Executed at a time	No. of times algorithm runs	Response time (in Seconds)
3	IPSO	1	1	1.11
		2	2	1.20
		3	3	1.34
		4	4	1.45
		5	5	1.87

Table 11: Response Time analysis Of IPSO for VM 4

VM Number	Algorithm Run	No. Of Application Executed at a time	No. of times algorithm runs	Response time (in Seconds)
4	PSO	1	1	1.15
		2	2	1.25
		3	3	1.42
		4	4	1.76
		5	5	2.15

While Execution of this calculation, we see that the performance of VMs depends on the quantity of assets that were dispensed at a specific time.

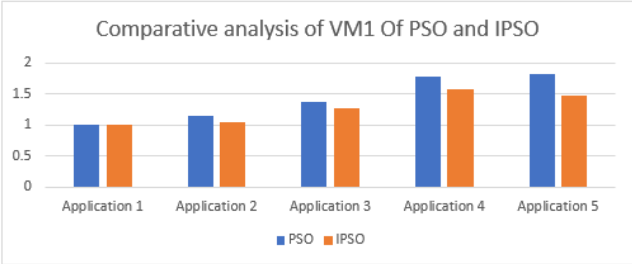

Fig 1: Comparative analysis of VM1 Of PSO and IPSO

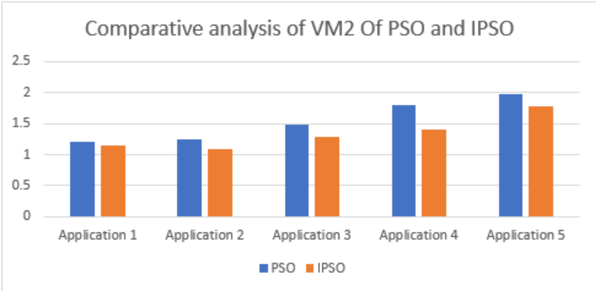

Fig 2: Comparative analysis of VM2 Of PSO and IPSO

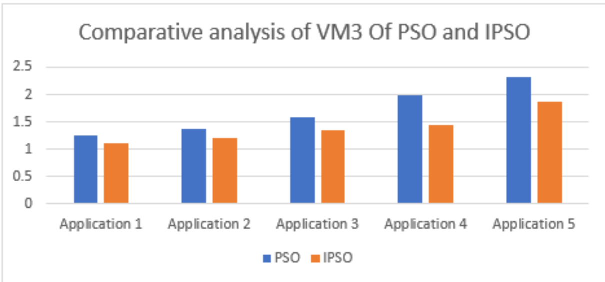

Fig 3: Comparative analysis of VM3 Of PSO and IPSO

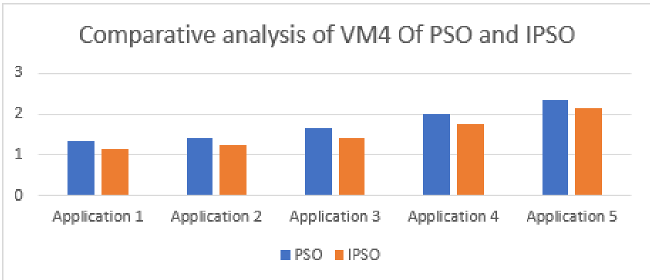

Fig 4: Comparative analysis of VM4 Of PSO and IPSO

5. Conclusion

CC handles productively the impending necessities for registering, it's fundamental for handle Issue ideally which emerge all over distributed computing climate. Load adjusting is difficult Issue which influences execution and usage of assets of the Server. Resource offloading is an essential aspect of a cloud environment, as it enables the efficient use of available resources and improves the overall performance of applications. There are various approaches to resource offloading, including task offloading, resource offloading, and computation offloading. While resource offloading provides several benefits, including reduced energy consumption and improved resource utilization, implementing resource offloading can be challenging due to latency and security-related issues. Nevertheless, resource offloading is an essential technology that will continue to play a critical role in cloud computing in the future. In this paper the authors have implemented real cloud environment using xen server with the Implementation of PSO and IPSO.

we have proposed a calculation for asset Usage which decrease the Reaction Season of the running applications in VM and assign (Offload) the heap on the servers. The examination has been finished on Xen Server. Diff. Applications has been utilized to produce the heap on virtual machines. The calculations are carried out in Python script. By the trial-and-error Outcome we saw that proposed calculation IPSO has impressively worked on the presentation, Asset Usage of running application in VMs.

Author contributions

Akash Dave, Conceptualization, Methodology, Software, Field study

Dr. Hetal Chudasama: Data curation, Writing-Original draft preparation, Software, Validation., Field study

Conflicts of interest

The authors declare no conflicts of interest.

Reference

[1] Kumaran K, Sasikala E. An efficient task offloading and resource allocation using dynamic arithmetic optimized double deep Q-network in cloud edge platform. Peer-to-Peer Networking and Applications. 2023 Feb 24:1-22.

[2] Zavieh H, Javadpour A, Li Y, Ja'fari F, Nasseri SH, Rostami AS. Task processing optimization using cuckoo particle swarm (CPS) algorithm in cloud computing infrastructure. Cluster Computing. 2023 Feb;26(1):745-69.

[3] Kouka N, BenSaid F, Fdhila R, Fourati R, Hussain A, Alimi AM. A novel approach of many-objective particle swarm optimization with cooperative agents based on an inverted generational distance indicator. Information Sciences. 2023 Apr 1;623:220-41.

[4] Pirozmand P, Jalalinejad H, Hosseinabadi AA, Mirkamali S, Li Y. An improved particle swarm optimization algorithm for task scheduling in cloud computing. Journal of Ambient Intelligence and Humanized Computing. 2023 Feb 15:1-5.

[5] Ismail HA, Riasetiawan M. CPU and memory performance analysis on dynamic and dedicated resource allocation using XenServer in Data Center environment. In2016 2nd International Conference on Science and Technology-Computer (ICST) 2016 Oct 27 (pp. 17-22). IEEE.

[6] Dave A, Patel B, Bhatt G, Vora Y. Load balancing in cloud computing using particle swarm optimization on Xen Server. In2017 Nirma University International Conference on Engineering (NUiCONE) 2017 Nov 23 (pp. 1-6). IEEE.

[7] Riasetiawan M, Ashari A, Endrayanto I. The analyses on dynamic and dedicated resource allocation on Xen server. TELKOMNIKA (Telecommunication Computing Electronics and Control). 2016 Mar 1;14(1):280-5.

[8] Bist M, Wariya M, Agarwal A. Comparing delta, open stack and Xen Cloud Platforms: A survey on open source IaaS. In2013 3rd IEEE International Advance Computing Conference (IACC) 2013 Feb 22 (pp. 96-100). IEEE.

[9] Dave A, Patel B, Bhatt G. Load balancing in cloud computing using optimization techniques: A study. In2016 International Conference on Communication and Electronics Systems (ICCES) 2016 Oct 21 (pp. 1-6). IEEE.