International Journal of Applied Mathematics

Volume 38 No. 5s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

ENHANCED TASK SCHEDULING PARTICLE SWARM OPTIMIZATION ALGORITHM (TSPSOA) BASED ON LOAD BALANCING FOR VIRTUAL MACHINE MIGRATIONS IN CLOUD COMPUTING

[1]A.Siva sankari, [2]Dr.S.Vimalanand

[1]Research Scholar, Department of Computer Science, Periyar University, Salem- 636 007, Tamil Nadu, India. Mail: sankarisiva001@gmail.com , **Orcid id: 0009-0006-8312-4069**

[2]Research Supervisor, Principal, Achariya college of arts and science Puducherry, India
Mail: aaschead@achariya.org , **Orcid Id: 0009-0007-2479-1790**

Abstract:

In the contemporary data centers, virtual machine migration is a crucial technique for gaining scalability through optimum resource use. The majority of earlier publications focused on how to move virtual machines in Cloud environments with high traffic needs. It mainly concentrated on the internet activity between virtual computers and the energy required by those machines. None of the preceding algorithms gave consideration to the client's experience, such as the wait time they experienced to have their services handled. Among the most significant problems that Cloud technology is now experiencing is load balancing. All the nodes should receive an equitable distribution of the load. In dynamic and diverse contexts, dynamic algorithms produce superior outcomes. This research suggests an enhanced particle swarm optimization approach for Cloud Computing job scheduling optimization. First, a scheduling model utilizing an upgraded particle swarm technique is suggested based on the Cloud Computing scheduling algorithm concept to prevent the optimization approach from devolving into local optimization. Three load balancing algorithms are used for comparison: the Honeybee Foraging Behavior Load Balancing Algorithm, the Throttled Load Balancing Algorithm, and the ESCE (Equally Spread Current Execution) Task Allocation Algorithm. It is demonstrated through the simulation results that the suggested TSPSOA is superior to the other state-of-the-art algorithms.

Keywords: Migration, Virtual machine, Cloud computing, Load balancing.

Introduction

The cloud is a collection of mobile Computers that provides an organisation with on-demand access to computational resources. Recently, cloud computing has become a cutting-edge breakthrough. It is a methodically distributed framework where processing resources obtained from the organisation (Cloud) and services are gathered to provide for the clients on a pay-based on circumstance basis.

Received: August 04, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Distributed computing provides assistance for everything and is represented by open, private, local, and mixed-breed mists. There are 3 important support layers for cloud computing: Software as a Service (SaaS), such as Google Online Workplace, Google Documents, Email clouds, etc., means that the customer is not responsible for the installation and configuration of any hardware or software; Platform as a Service (PaaS), wherein help is the delivery of a web-based application development platform allowing users to create and launch their own apps as needed. A merchant or cloud provider is in charge of setting up the server and calculation platform. Google App Engine is an example of PaaS. Foundation as a Service (IaaS), in which the cloud provider provides servers, software, and organizational equipment on a non-request basis [1].

Distributed computing utilizes the pool of server farms to convey admittance to the figuring assets, wherein help is the delivery of a web-based application development platform allowing users to create and launch their own apps as needed. A merchant or cloud provider is in charge of setting up the server and calculation platform. Google App Engine is an example of PaaS. Foundation as a Service (IaaS), in which the cloud provider provides servers, software, and organizational equipment on a non-request basis [2].

In Cloud processing, admittance to the asset depends on Virtualization [3]. Virtualization is Load unevenness might cause framework bottleneck. To accomplish asset usage and no postpone in offering support, asset portion ought to be finished in an effective manner. Hubs of the framework can be coherently gathered into group and undertaking of burden adjusting is disseminated among bunches. Every individual group will allot burden to the hubs having a place with that bunch. This can be set up in progressive structure. For Cloud climate different burden adjusting approaches have been carried out to give productive conveyance of burden among accessible machines.

Assignment of Cloud assets to clients on request brings about the issue of burden adjusting. In the event that responsibility isn't conveyed as expected, then a few hubs in Cloud will be vigorously stacked and a few hubs will be under stacked. Similarly in the event that the assets given by the Cloud are not distributed proficiently, it prompts defer in offering support to the clients [4]. Load unevenness might cause framework bottleneck. To accomplish asset usage and no postpone in offering support, asset portion ought to be finished in an effective manner. Hubs of the framework can be coherently gathered into group and undertaking of burden adjusting is disseminated among bunches. Every individual group will allot burden to the hubs having a place with that bunch. This can be set up in progressive structure. For Cloud climate different burden adjusting approaches have been carried out to give productive conveyance of burden among accessible machines.

Dynamic calculations give improved brings about conditions. These calculations are more adaptable. Virtualization is Load unevenness might cause framework bottleneck. To accomplish asset usage and no postpone in offering support, asset portion ought to be finished in an effective manner. Hubs of the framework can be coherently gathered into group and undertaking of burden

Received: August 04, 2025

adjusting is disseminated among bunches. Every individual group will allot burden to the hubs having a place with that bunch. This can be set up in progressive structure. For Cloud climate different burden adjusting approaches have been carried out to give productive conveyance of burden among accessible machines.Dynamic calculations can think about the unique changes to the characteristics. Notwithstanding, these calculations are more complicated [5]. Fundamental benefit of this is that determination of errand depends on present status and this will assist with working on the exhibition of the framework. In Distributed System, every one of the hubs communicate with one another and load adjusting calculation is executed by every one of the hubs conveyed among every one of the hubs. Communication among hubs can be agreeable or non-helpful [6]. Assuming that any hub bombs in the framework, it won't stop the usefulness.

I) In agreeable appropriated framework, all hub cooperates.

ii) In non-helpful appropriated framework, every hub works freely.

The principal contrast is, albeit in view of a straightforward rule where more loads are evoked on servers and bringing about imbalanced traffic. Though in powerful burden adjusting is anticipated on a question that can be made much of the time on the servers, yet some of the time existed traffic will forestall questions to be responded to and correspondingly more added above.

Literature Survey:

The main component of going with choices halfway is straightforwardness. Nonetheless, unified calculations experience the ill effects of the issue of the bottleneck and single point disappointment. Dispersed load adjusting calculations are liberated from these issues. The choices in regards to the utilization of their assets autonomous of the impact of their choice on the remainder of the framework. In the previous one every processor has the obligation to do its own piece of the booking errand to accomplish a typical framework wide objective [7].

Honey bee Behavior Load Balancing Algorithm - This computation was made using real bumble bee behavior when looking for food sources. The bumble bees search for food sources and then come back to a apiary to light the food source. They carry out bunch development to achieve this. The term "waggle Dance" is another name for this gathering development. They waggle dance to signal to other honey bees the precise location of the food supply. This waggle dance displays the caliber, quantity, and proximity of the food supply to the colony [8]. Advantage Self-sorting, aliened by nature, calculation. (ii) Efficiency will be achieved by increasing the size of the framework. (iii) Adaptable to a variety of climates. negative is The overall throughput won't increase with an increase in assets.

Throttled Load Balancing Algorithm-Calculations for throttled load adjustment are best suited for virtual machines. The load balancer maintains up with the list of all the framework's virtual machines. To accomplish asset usage and no postpone in offering support, asset portion ought to be finished in an effective manner. Hubs of the framework can be coherently gathered into group and undertaking of burden adjusting is disseminated among bunches. Every individual group will allot burden to the hubs having a place with that bunch. This can be set up in progressive structure. For Cloud climate different burden adjusting approaches have been carried out to give productive conveyance of burden among accessible machines examines the ordering table. The occupation is assigned to that virtual machine if it is available. Following each allocation and de-portion of an item, the load balancer updates the ordering table [8]. Advantages include I keeping track of the status of each VM together with its list, (ii) good execution, and (iii) improved asset usage. The load is continuously scanning the list of VMs without considering the on-going strain on the VM.

ESCE - The listing of all virtualization and jobs is kept up to date by the load balancer. The listing of virtual machines is filtered by the load balancer when it gets a request. The request is routed to that specific VM in the unusual chance that even one is found that can execute the client's request. The identical burden is uniformly distributed across all VMs using this approach [9].Benefits include maintaining an even workload across all VMs and increasing throughput. The weakness of the central government is obstruction, which is also a closed-minded weakness.

A work flow control design calculation based on the ancestral computations model has been put out by J. Huang [10] in the context of cloud computing. The trial's outcomes showed that the asset distribution's proficiency had been met, and at the same time, they had shortened the completion period.

Cloud Services Pricing Models have been presented by May Al-Roomi, et al [11]. This study focuses on contrasting several used and suggested valuing models approaches and includes the benefits and drawbacks of each. The correlation relies on a variety of factors, including decency, valuing strategy, as well as usage duration.

Booking calculations in the era of cloud computing highlight the necessity of finding a potential schedule for the tasks. The purpose of the undertaking planning problem is to obtain the base typically execution time, and subsequent papers have demonstrated that this problem is NP-Complete. The problem of scheduling a lot of errands with a lot of groupings to the a lot of m mists is addressed by Panda, S.K., et al. [12] in such a way that the overall delay time is constrained. Keep in mind that the size of the temporal gaps between matched assignments is the overall delay time. We now provide a control design calculation for the cloud computing environment that is based on the Hungarian calculation, a considerable improvement calculation.The suggested formula balances the tasks and takes into account a variable amount of tasks and mists before deciding on the booking option. In 22 distinct datasets, we replicate the suggested calculation and compare it to three other calculations: the first-start items out served calculation, the Hungarian computation with rent time, and the Hungarian calculation with talk rent time.

International Journal of Applied Mathematics
Volume 38 No. 5s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

The presentation assessment shows that the proposed calculation delivers better delay time in contrast with existing calculations. The proposed calculation is dissected hypothetically and displayed to require O(kpl2) time for k cycles, p redundancies and l assignments.

RESEARCH METHODOLOGY:

The virtual machine deployments in the proposed system enhance network performance by using cloud computing. The lifetime of the system will be enhanced by the load balancing strategy that we are utilising.

When a real computer is overloaded, virtualization can be used to generate a virtual or duplicated version of a device or resource, such as a server, memory stick, network, or operating system. The primary difficulty here is to move virtual machines around to dynamically disperse the load from such a overloaded physical machine to another side in order to prevent bottlenecks in the cloud computing system.

It is preferable for load balancing algorithms to be less complicated and must be simple to use and comprehend. Complexity concerns will have a detrimental impact on performance. Every load balancing decision maker (scheduler) in the cooperative scheme is responsible for completing its own share of the scheduling task, but everyone is striving to achieve the same system-wide objective.

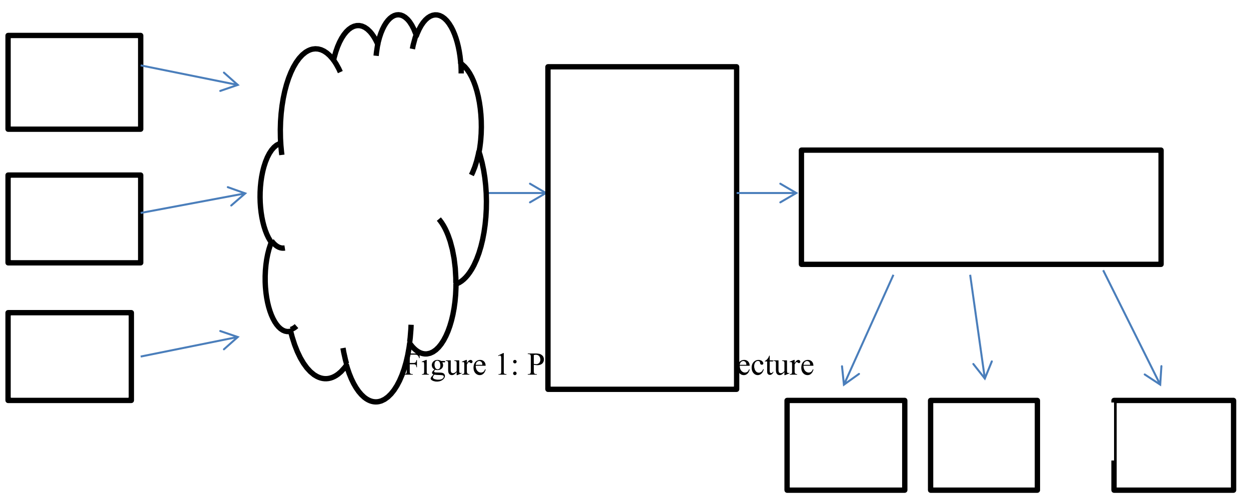

Figure 1: P ecture

A Task Scheduling Particle Swarm Optimization Algorithm (TSPSOA) Based on Load Balancing: a two-level work scheduling approach built on task scheduling to accommodate users' changing needs and achieve high resource utilisation. By first assigning tasks to virtual servers and would then virtualization software to host resources, it achieves load balancing while also enhancing task response times, resource usage, and overall cloud computing environment

International Journal of Applied Mathematics
Volume 38 No. 5s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

performance. The extra jobs were transferred to the new host VMs using the Particle Swarm Optimization (PSO) optimization model.

An abstract model that describes tasks without identifying the precise location of the resources used to carry out the tasks is typically used as the input to task scheduling algorithms. There are non-deterministic and deterministic abstract task-scheduling models. The dependencies of jobs and I/O data are known at run time in a nondeterministic paradigm, but only in advance in a deterministic one.

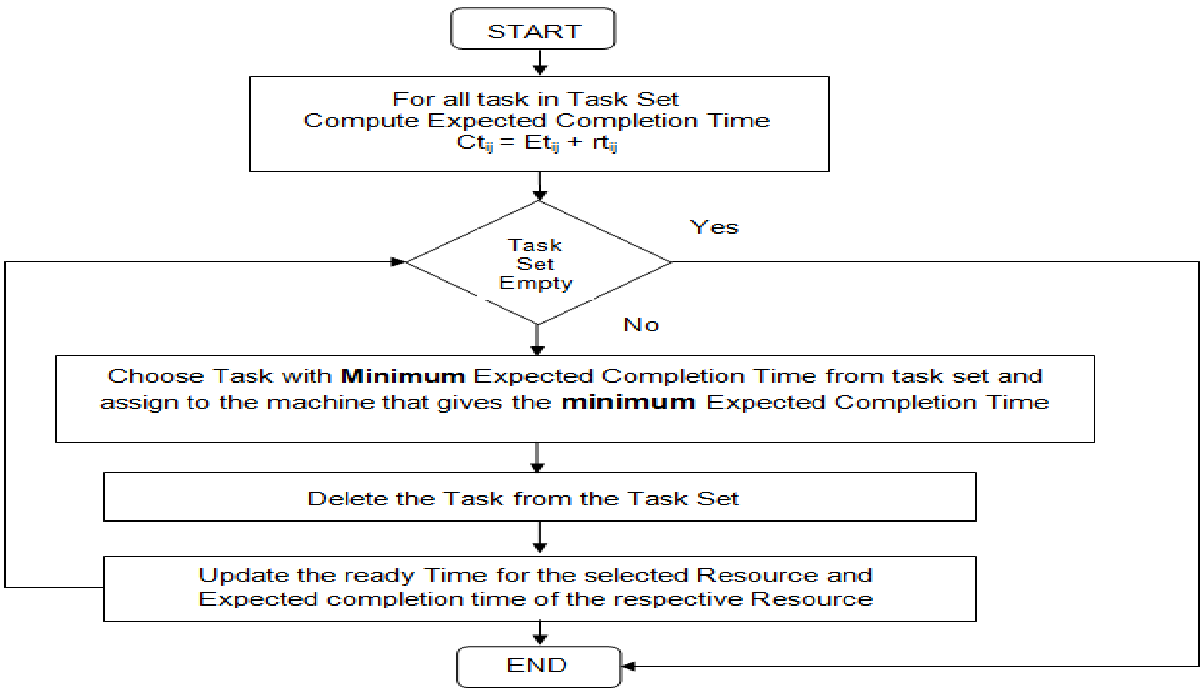

Figure 2: Task scheduling flow chart

Following the gathering of data on the Cloud's resources (during the discovery phase), a list of suitable candidates is identified. The resource selection process chooses the potential solution that satisfies all criteria and makes the best use of the infrastructure. Particle Swarm Optimization (PSO) for Clouds is an optimization approach that might be used to choose the resources. Additionally, optimization tactics may be used to enhance certain Cloud features like energy conservation or to deal with security as well as fault tolerance regulations.

Kennedy and Eberhart first presented the Particle Swarm Optimization (PSO) method in 1995, which relies on the social behavior of the particles. PSO uses guidelines inspired by the behavior of flocks of birds seeking for food to find the best solution to a challenge, which is known as a fitness task. PSO, a population-based meta-heuristic, has lately become quite popular because to its dependability, simplicity, and efficacy in a variety of applications with little computational

Volume 38 No. 5s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

overhead. As a result, creating effective PSO implementations is a challenge with significant application.

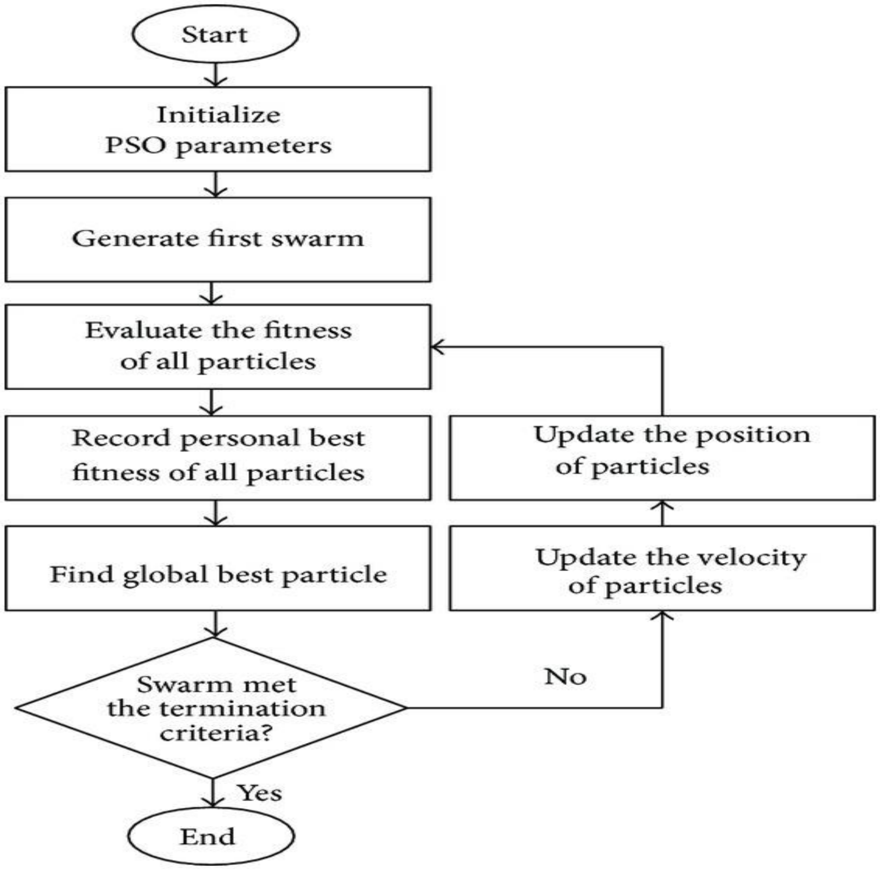

Figure 3: Flowchart of PSO

PERFORMANCE ANALYSIS: (Implementation in progress)

The proposed TSPSOA's performance in comparison in this study to that of the Honeybee Foraging Behavior Load Balancing Algorithm, the Throttled Load Balancing Algorithm, and the ESCE Load Balancing Algorithm. The analysis's characteristics include throughput, fast response, resource use, and reliability.

Throughput

Throughput is the number of data blocks. It is frequently utilized in devices ranging from businesses to various device and network components.Table 1 comparesthe throughput of existing Honeybee Foraging Behavior Load Balancing Algorithm, Throttled Load Balancing Algorithm and ESCE with proposed TSPOA.

International Journal of Applied Mathematics
Volume 38 No. 5s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Table 1: Throughput Comparison

System Load	Throughput (%)			
	Honeybee Foraging Behavior Load Balancing Algorithm	**Throttled Load Balancing Algorithm**	**ESCE**	**Proposed TSPSOA**
2	92	91	90	98
4	90	89	89	98
6	89	87	87	97
8	84	85	85	95
10	83	84	82	94
12	81	81	81	94
14	80	76	79	92

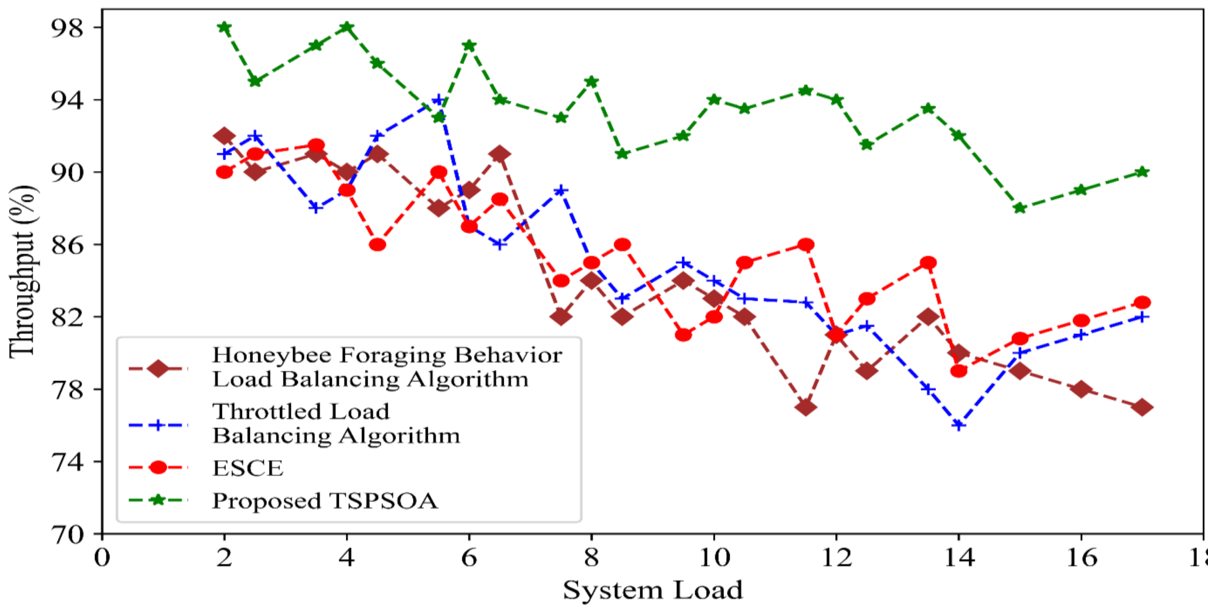

Figure 4: Throughput Analysis

Figure 4 shows a throughput analysis comparison with the Throttled Load Balancing Algorithm, the Honeybee Foraging Behavior Load Balancing Algorithm, and ESCE. System load is depicted on the X-axis, while throughput as a percentage is shown on the Y-axis. Red shows ESCE, blue suggests Throttled Load Balancing Algorithm, brown indicates Honeybee Foraging Behavior Load Balancing Algorithm, and green indicates suggested TSPSOA. The graphic demonstrates that the suggested strategy outperforms the currently used methods in terms of outcomes.

Response time

Response time, is a measure of the amount of time required for a packet to travel across a path from a sender to a receiver.

Table 2 compares the Response time of existing Honeybee Foraging Behavior Load Balancing Algorithm, Throttled Load Balancing Algorithm and ESCE with proposed TSPOA.

Table 2: Response time Comparison

System Load	Response time (ms)			
	Honeybee Foraging Behavior Load Balancing Algorithm	**Throttled Load Balancing Algorithm**	**ESCE**	**Proposed TSPSOA**
2	95	92	90	85
4	98	97	96	86
6	100	105	103	90
8	108	102	100	92
10	110	100	95	95
12	115	105	100	98
14	120	112	110	100

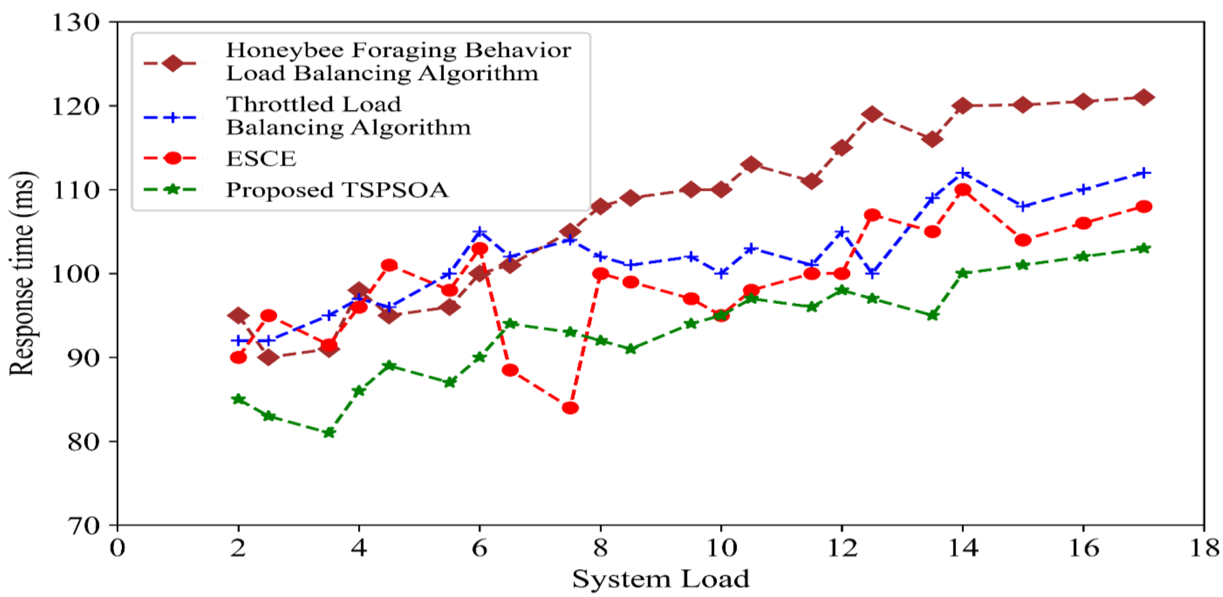

Figure 5: Response time Analysis

International Journal of Applied Mathematics
Volume 38 No. 5s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Figure 5 shows a comparison of the Response Time Analysis using the Throttled Load Balancing Algorithm, the Honeybee Foraging Behavior Load Balancing Algorithm, and ESCE. System load is depicted on the X-axis, while response time in milliseconds is shown on the Y-axis. Red shows ESCE, blue suggests Throttled Load Balancing Algorithm, brown indicates Honeybee Foraging Behavior Load Balancing Algorithm, and green indicates suggested TSPSOA. The graphic demonstrates that the suggested strategy outperforms the currently used methods in terms of outcomes.

Resource Utilization

The performance evaluation as well as effort over a time period is known as resource utilisation (or capacity).

Table 3 compares the Resource utilization of existing Honeybee Foraging Behavior Load Balancing Algorithm, Throttled Load Balancing Algorithm and ESCE with proposed TSPOA.

Table 3: Resource utilization Comparison.

System Load	**Resource Utilization (%)**			
	Honeybee Foraging Behavior Load Balancing Algorithm	**Throttled Load Balancing Algorithm**	**ESCE**	**Proposed TSPSOA**
2	76	79	81	83
4	79	81	83	85
6	81	83	85	86
8	83	85	87	90
10	85	87	89	92
12	87	89	90	95
14	89	90	92	96

Received: August 04, 2025

International Journal of Applied Mathematics
Volume 38 No. 5s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

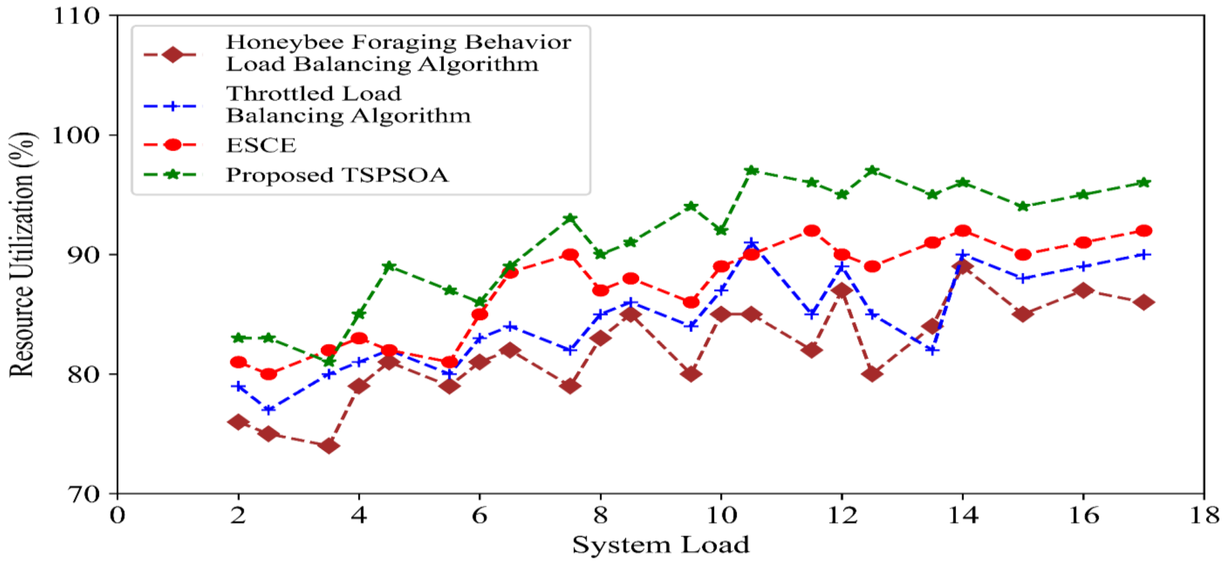

Figure 6: Resource utilization Analysis

Figure 6 compares resource consumption using the Throttled Load Balancing Algorithm, the Honeybee Foraging Behavior Load Balancing Algorithm, and ESCE. System load is shown on the X-axis and the Y-axis. The percentage of resources used is shown. Red shows ESCE, blue suggests Throttled Load Balancing Algorithm, brown indicates Honeybee Foraging Behavior Load Balancing Algorithm, and green indicates suggested TSPSOA. The graphic demonstrates that the suggested strategy outperforms the currently used methods in terms of outcomes.

Stability

The process of using commands to make sure every router and other devices being used to connect an intranet to the Internet.

Table 4 compares the Stability of existing Honeybee Foraging Behavior Load Balancing Algorithm, Throttled Load Balancing Algorithm and ESCE with proposed TSPOA.

Table 4: Stability Comparison

System Load	**Stability (%)**			
	Honeybee Foraging Behavior Load Balancing Algorithm	**Throttled Load Balancing Algorithm**	**ESCE**	**Proposed TSPSOA**
2	80	82	84	85
4	82	84	85	86

International Journal of Applied Mathematics
Volume 38 No. 5s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

6	84	85	86	90
8	85	86	90	92
10	86	90	92	95
12	90	92	95	96
14	92	95	96	98

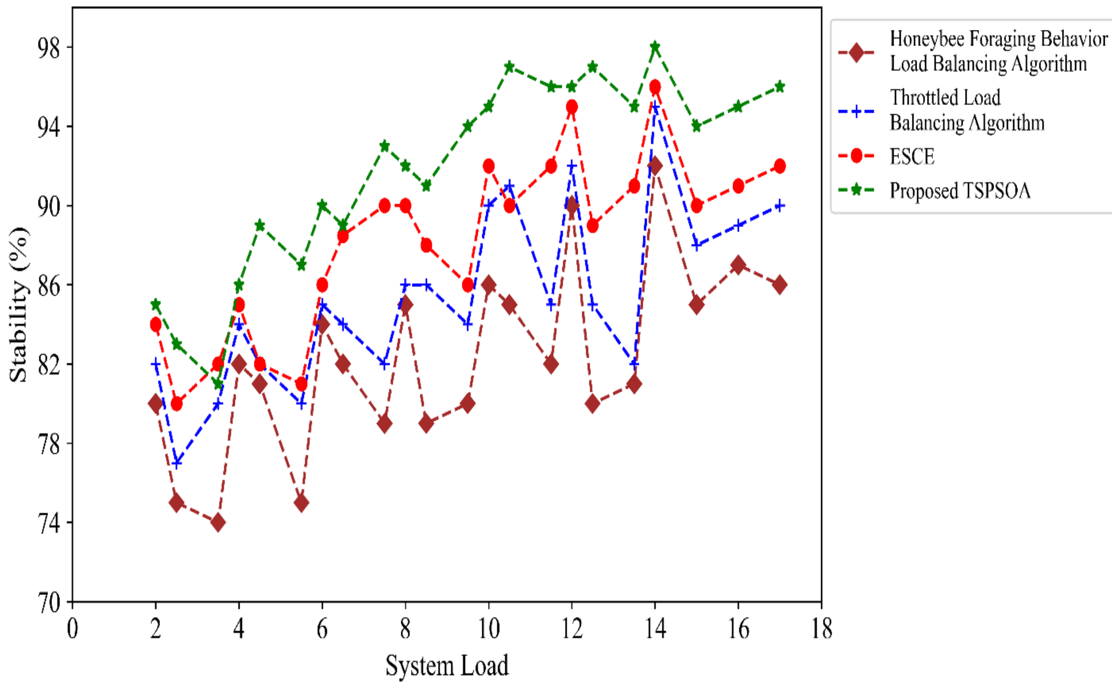

Figure 7: Stability Analysis

Figure 6 shows a stability comparison using the Throttled Load Balancing Algorithm, the Current Honeybee Foraging Behavior Load Balancing Algorithm, and ESCE. System load is shown on the X-axis and the Y-axis. Percentage stability is demonstrated. Red shows ESCE, blue suggests Throttled Load Balancing Algorithm, brown indicates Honeybee Foraging Behavior Load Balancing Algorithm, and green indicates suggested TSPSOA. The graphic demonstrates that the suggested strategy outperforms the currently used methods in terms of outcomes.

Conclusion:

This work proposes an enhanced particle swarm optimization technique for job scheduling optimization in cloud computing. First, a scheduling model utilizing an upgraded particle swarm method is suggested based on the concept of cloud computing job scheduling in order to prevent the optimization approach from succumbing to local optimization. Three load balancing algorithms are used for comparison: the Honeybee Foraging Behavior Load Balancing Algorithm,

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

the Throttled Load Balancing Algorithm, and the ESCE (Equally Spread Current Execution) Load Balancing Algorithm. The simulation results demonstrate which the proposed TSPSOA outperforms existing cutting-edge algorithms.

Reference:

1. A. Beloglazov, and R. Buyya, Energy efficient resourcemanagement in virtualized Cloud data centers, Proc. 10thIEEE/ACM international conference on cluster, Cloud andgrid computing, 2010, 826-831.
2. Morabito, Roberto, and NicklasBeijar. "Enabling data processing at the network edge through lightweight virtualization technologies." Sensing, Communication and Networking (SECON Workshops), 2016 IEEE International Conference on. IEEE, 2016.
3. Rajwinder Kaur and PawanLuthra," Load Balancing in Cloud Computing", Association of Computer Electronics and Electrical Engineers, 2014, DOI: 02.ITC.2014.5.92
4. Ashalatha R; J. Agarkhed, "Dynamic load balancing methods for resource optimization in Cloud Computing environment ", 2015 Annual IEEE India Conference (INDICON) , Pages: 1 - 6, DOI: 10.1109/INDICON.2015.7443148
5. R. Kanakala; V. K. Reddy; K. Karthik, " Performance analysis of load balancing techniques in Cloud Computing environment", 2015 IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT), Pages: 1 - 6, DOI: 10.1109/ICECCT.2015.7226052
6. A. A. Jaiswal, Dr. Sanjeev Jain," An Approach towards the Dynamic Load Management Techniques in Cloud Computing Environment", 2014 International Conference on Power, Automation and Communication (INPAC
7. K. Garala; N. Goswami; P. D. Maheta, " A performance analysis of load Balancing algorithms in Cloud environment ", 2015 International conference on Computer Communication and Informatics (ICCCI), Pages: 1 - 6, DOI: 10.1109/ICCCI.2015.7218063
8. Sidra Aslam, Munam Ali Shah, "Load Balancing Algorithms in Cloud Computing: A Survey of Modern Techniques", 2015 National Software Engineering Conference (NSEC 2015)
9. SurbhiKapoor, Dr. ChetnaDabas," Cluster Based Load Balancing in Cloud Computing", 2015 Eighth International Conference on Contemporary Computing (IC3)
10. J.Huang. "The Workflow Task Scheduling Algorithm Based on the GA Model in the Cloud Computing Environment." Journal of Software, vol. 9, No 4, PP. 873- 880, April 2014.
11. M.Al-Roomi, S.Al-Ebrahim, S.Buqrais and I.Ahmad, "Cloud Computing Pricing Models: A Survey", Vol.6, No.5 (2013), pp.93-106, International Journal of Grid and Distributed Computing.
12. Panda, S.K., et al. A pair-based task scheduling algorithm for Cloud Computing environment. Journal of King Saud University – Computer and Information Sciences (2018).