ISSN: 2348 - 2117

International Journal of Engineering Technology and Computer Research (IJETCR)

Available Online at www.ijetcr.org

Volume 11; Issue 03; May-June; 2023; Page No. 10-13

Analysis of Hybrid Load Balancing Strategy Used In-cloud Computing

Shaharyar Ahmad[1], Sunil Vishwakarma[2], Abhishek Yadav[3]

[1,2,3] Department of Computer Science and Engineering of BBD University, Lucknow, India

Conflicts of interest: Nil

Corresponding author: Shaharyar Ahmad

Abstract

This exploration paper presents a relative investigation of two load balancing procedures in cloud computing: Min-Max algorithm and Counterfeit Honey bee Settlement (ABC) algorithm. The review planned to assess the exhibition of these algorithm concerning adjusting jobs among various servers in a cloud climate. The Min-Max algorithm was carried out by doling out assignments with the greatest handling time to the server with the base accessible limit. The ABC algorithm, then again, was executed by considering different boundaries like handling time, asset use, and energy utilization to decide the best server to dispense an errand to. The exhibition of these algorithm was assessed utilizing measurements, for example, reaction time, asset usage, and energy utilization.

Keywords: Cloud computing, Load Balancing, Modified Honeybee, Minmax, Virtual Machine.

Introduction

Load adjusting in cloud computing is a course of conveying jobs and organization traffic across numerous servers in a cloud climate. The objective is to disperse approaching solicitations equitably among accessible servers to guarantee most extreme usage of assets and limit reaction time. This assists with trying not to over-burden a solitary server and decrease the gamble of margin time.[1]

LOAD BALANCING

Load Balancing is basically divided into two parts – static and dynamic load balancing and further both of them together form a hybrid load balancing algorithm.

***Hybrid Load Balancing algorithm**:* The principal objective of load balancing is to ensure that not a single solitary hub isover-burden nor any hub ought to be kept inactive. load adjusting make hub in adjusted way and keep a track neither any centre point is over-burden or under stacked this is significant issue enormous blast in market for climate figuring key job of load balancing is to forestall server or organization by being over-burden or under stacked, load adjusting is fundamentally separated by two sections. Static and dynamic load balancers in static load balancing working respects to past information data information has been moved by past status as opposed to thinking about present status further once the information is designated it can't move or redistributes there can't be any change whenever designation has been done, instances of static load balancing are min-max max-min and cooperative effort.[2]in dynamic load balancing in light of three primary rules flowed non-coursed and semi-circled approaches in circled strategies all hubs are dispatched similarly in non-circled technique there is one in the middle between which assembles all solicitations and where all hubs are separated what's more here each group is fundamental centre point which then disperses information then at that point move information to all servers in semi-circle strategy there are gathering of group illustration of dynamic load balancing isjoin inactive line bumble bee and subterranean insect settlement and so on.[3]

Modified Artificial Bee Colony Algorithm: Modified Artificial Honey bee algorithm (ABC) is a nature-motivated streamlining calculation utilized in cloud computing for loadadjusting and asset portion. The calculation depends on the way of behaving of bumble bees and mimics the rummagingconduct of honeybees to look for nectar. In cloud computing,the ABC algorithm is utilized to convey responsibilities among different servers in a cloud climate. The algorithm works by considering different boundaries like handling time, asset use, and energy utilization to decide the best server to distribute an undertaking to. The ABC algorithm is viewed asa successful burden adjusting method as it can track down close ideal arrangements in a short measure of time and can adjust to evolving responsibilities. It is broadly utilized in cloud computing conditions for effective asset use and high framework execution.[4]

Min-Max algorithm- Min-Max algorithm is a heap adjustingprocedure utilized in cloud computing to disperse jobs among various servers in a cloud climate. The algorithm works by allotting the undertaking with the most extreme handling timeto the server with the base accessible limit. Along these lines, the heap is adjusted such that the complete handling time is limited while guaranteeing that no server is overpowered. The Min-Max algorithm assists with forestalling over-burdening a solitary server and guarantee high accessibility and executionof the cloud climate. It is generally utilized in cloud computing conditions where the handling season of assignments isn't known ahead of time and can fluctuate significantly.

LITERATURE REVIEW

SN	LITERATURE REVIEW		
	AUTHOR	**TITLE**	**PROPOSE SOLUTION**
1-	Vinay Kumar Prasad, Dr. Dharamveer Singh, Vikas Gupta.	Round robin Based Dynamic Load Balancing in Cloud Computing	Make span, Resource Utilization
2-	Syed Abuthahir Jaganathan, Ramesh Saha.	An Efficient Enhanced Dynamic Load Balancing Weighted Round Robin Algorithm for Virtual Machine in Cloud Computing	Adaptive fault tolerant algorithm
3-	Bivasa Ranjan Parida, Amiya Kumar Rath, Hitesh Mohapatra.	Binary Self- Adaptive Salp Swarm Optimization- Based Dynamic Load Balancing in Cloud Computing	Resource Utilization Makespan
4-	Abderraziq Semmoud, Mourad Hakem, and Badr Benmammar.	Distributed Fault Tolerant Algorithm for Load Balancing in Cloud Computing Environments	Makespan, Energy Efficiency
5-	T.P.Latchoumi, Latha Parthiban.	Quasi Oppositional Dragonfly Algorithm for Load Balancing.	Makespan, Energy Efficiency

6-	Santosh Waghmode, Bankat M. Patil	Adaptive Load Balancing in Cloud Computing Environment	Divisible weighted round- robin, round- robin, and weighted least connection algorithm
7-	SsTripathy, Mishra, DS Roy, K Yadav	State-of-the-Art Load Balancing Algorithms for Mist- Fog-Cloud Assisted Paradigm	presents a deep meta- analysis and research scopes
8-	MSumathi, Vijayaraj, SP Raja	HHO-ACO hybridized load balancing technique in cloud computing	provide better throughpu t and performan ce to requestor

PERFORMANCE EVOLUTION METRICS

Hybrid load balancing algorithm in cloud computing are designed to improve the allocation of resources to tasks while minimizing the execution time, energy consumption, and cost. The performance evaluation of such algorithms involves the use of several metrics that assess their efficiency, effectiveness, and quality. Here are some common performance evaluation metrics for hybrid balancing in cloud computing.[5]

Makespan: Makespan is the total time taken to completeall the tasks in a job. The lower the make span, the more efficient the balancing algorithm is. The algorithm metric is a crucial performance evaluation metric for hybrid load balancing algorithm in cloud computing as it measures the efficiency of the algorithm.[6]

Energy Consumption: Energy consumption is the amount of energy consumed by the cloud infrastructure to execute the tasks in a job. The lower the energy consumption, the more energy-efficient the balancing algorithm is. Energy consumption evaluation metric for hybrid load balancing in cloud computing, as it measures the effectiveness of the algorithm in reducing energy consumption.[7]

Cost: Cost is the total amount of money spent on executing the task in a job , including the cost of resources, energy consumption, and maintenance. The lower the cost, the more cost-effective the balancing algorithm is. Cost is an important performance evaluation metric for hybrid load balancing in cloud computing as it measures the efficiency ofthe algorithm in reducing the cost of executing a job.[8]

Resource utilization: Resource utilization is the percentage of resources that are being utilized by the tasks. The higher the resource utilization, the more efficient the balancing algorithm is. Resource utilization is an important performance evaluation metric for hybrid load balancing in cloud computing, as it measures the efficiency of the algorithm in utilizing the available resources.[9]

Throughput: Throughput measures the number of tasks completed per unit of time. The higher the throughput, the more efficient the balancing algorithm is. Throughput is an important performance evaluation metric for hybrid load balancing in cloud computing, as it measures the efficiency

of the algorithm in executing the tasks.[10]

In conclusion, the performance evaluation of hybrid load balancing algorithm in cloud computing requires the use ofmultiple metrics that measures the efficiency, effectiveness, and quality of the algorithm. The selection of appropriate metrics should be based on the specific requirements of the job and the characteristics of the cloud infrastructure.[10]

Conclusion

Many researchers have worked on static and dynamic load balancing approaches but still efficient results have not beenachieved which result in evolution of hybrid algorithm.

This algorithm is the combination of static and dyammicstrategy to overcome the lacking facilities

- This paper is proposing of algorithm in cloud computingfor better result in hybrid environment.

References

1. S. Sharma, S. Kumar, S. Mohapatra, and R. Rani, "Discrete gravitational search algorithm for virtual machine placement in cloud computing," *Int. J. Adv. Sci. Technol.*, vol. 29, no. 8 Special Issue, pp. 1261–1267, 2020.
2. S. Sahu, M. Cse, J. Vidyapeeth, A. Saini, S. Kumar, and S. Rathod, "Security into Cloud Storage with Adding Integrity and Timed Ephemerizer," 2014.
3. P. Sharma Ȧ, S. Singh Ȧ, and P. Baheti Ḃ, "Digital Video Watermarking with Multi-level Discrete 3-D Wavelet Transform," 2014.
4. S. Kumar, K. K. Dubey, A. K. Gautam, S. Verma, V. Kumar, and U. Mamodiya, "Detection of recurring vulnerabilities in computing services," *J. Discret. Math. Sci. Cryptogr.*, vol. 25, no. 4, pp. 1063–1071, 2022, doi: 10.1080/09720529.2022.2072432.
5. P. Joshi, S. Kumar, and A. S. Raghuvanshi, "A performance efficient joint clustering and routing approach for heterogeneous wireless sensor networks," *Expert Syst.*, no. July, pp. 1–26, 2022, doi: 10.1111/exsy.13121.
6. S. Kumar *et al.*, "Protecting location privacy in cloud services," *J. Discret. Math. Sci. Cryptogr.*, vol. 25, no. 4, pp. 1053–1062, 2022, doi: 10.1080/09720529.2022.2072430.
7. M. Malik *et al.*, "High level browser security in cloud computing services from cross site scripting attacks," *J. Discret. Math. Sci. Cryptogr.*, vol. 25, no. 4, pp. 1073–1081, 2022, doi: 10.1080/09720529.2022.2072434.
8. A. Asha *et al.*, "An optimized DEEC approach for efficient packet transmission in sensor based IoTs network," *Microprocess. Microsyst.*, vol. 96, no. October 2022, p. 104714, 2023, doi: 10.1016/j.micpro.2022.104714.
9. S. Kumar, N. Ranjan, B. B. Parashar, and R. Kumar, "Data Gathering and Forwarding in Wireless Sensor Network with Mobile Sink by Using Shortest Path Algorithm to Improve the Life of Sink," no. 15354, pp. 15354–15361, 2020.
10. C. Singh Ȧ, S. Singh Ȧ, and P. Baheti Ḃ Ȧ, *Human Face Recognition and Face Detection using Skin Colour Model*, vol. 4, no. 4. 2014