Load Scheduling Of Tasks in Cloud Computing Using Honey Bee Behavior

Monika Rathore, Sarvesh Rai

***Abstract*— Cloud computing has recently rising technology, day by day having wide scope in future. Cloud computing is outlined as an oversized scale distributed computing paradigm that's driven by social science of scale during which a pool of abstracted virtualized energetically. the amount of users in cloud computing is growing exponentially. sizable amount of user requests tries to designate the resources for several applications that together with to high load shortly abroad faraway from cloud server. Whenever sure VMs square measure full then no additional tasks ought to be send to full virtual machine if below loaded virtual machines square measure out there. For optimize answer and higher time interval the load must be balanced among full and below loaded virtual machines. during this paper, Associate in Nursing formula is planned named honey bee behavior based mostly load reconciliation (HBB-LB), that targets to attain well balanced load across virtual machine.**

I. INTRODUCTION

It is a method of reassigning the entire load to the individual nodes of the collective system to form resource utilization effective and to enhance the time interval of the duty, at the same time removing a condition during which a number of the nodes square measure over loaded whereas some others square measure below loaded. A load reconciliation formula that is dynamic in nature doesn't think about the previous state or behavior of the system, that is, it depends on this behavior of the system. The necessary things to think about whereas developing such formula square measure : estimation of load, comparison of load, stability of various system, performance of system, interaction between the nodes, nature of labor to be transferred, choosing of nodes and lots of different ones [4] .

This load thought of is in terms of processor load, quantity of memory used, delay or Network load.

II. GOALS OF LOAD SCHEDULING SQUARE MEASURE

- To improve the performance considerably
- To have a backup set up just in case the system fails even part
- To maintain the system stability
- To accommodate future modification within the system

TYPES OF LOAD SCHEDULING ALGORITHMS

Manuscript received July 18, 2015

Depending on United Nations agency initiated the method, load reconciliation algorithms is of 3 categories as given in [4]:

- **Sender** : If the load reconciliation formula is initialized by the sender
- **Receiver**: If the load reconciliation formula is initiated by the receiver
- **Symmetric**: it's the mix of each sender initiated and receiver initiated

III. TYPES OF LOAD SCHEDULING

Depending on this state of the system, load reconciliation algorithms is divided into two categories as given in [4]:

- **Static**: It doesn't rely on this state of the system. previous data of the system is required
- **Dynamic**: choices on load reconciliation square measure supported current state of the system.

No previous data is required. therefore it's higher than static approach. Here we'll discuss on numerous dynamic load reconciliation algorithms for the clouds of various sizes.

3.1 STATIC LOAD SCHEDULING

In static setting the cloud supplier installs solid resources. additionally the resources within the cloud aren't versatile once setting is formed static. during this state of affairs, the cloud needs previous data of nodes capability, process power , memory, performance and statistics of user needs. These user needs aren't subjected to any amendment at run-time. Algorithms planned to attain load reconciliation in static setting cannot adapt to the run time changes in load. though static setting is simpler to simulate however isn't compatible for heterogeneous cloud setting.

Round Robin formula [13] provides load scheduling in static setting. during this the resources square measure provisioned to the task on first-cum-first-serve (FCFS- i.e. the task that entered 1st are going to be 1st allotted the resource) basis and scheduled in sharing manner. The resource that is least loaded (the node with least range of connections) is allotted to the task. Eucalyptus uses greedy (first-fit) with round-robin for VM mapping. Radojevic planned Associate in Nursing improved formula over spherical robin known as CLBDM (Central Load reconciliation call Model) [14]. It uses the idea of spherical robin however it additionally measures the length of association between consumer and server by calculative overall execution time of task on given cloud resource.

3.2 DYNAMIC LOAD SCHEDULING

In a distributed system, dynamic load scheduling is exhausted 2 completely different ways: distributed and non-distributed. within the distributed one, the dynamic load scheduling formula is dead by all nodes gift within the system and also the task of load reconciliation is shared among them. The

interaction among nodes to attain load scheduling will take 2 forms: cooperative and non-cooperative [4]. within the 1st one, the nodes work side-by-side to attain a standard objective, for instance, to enhance the time interval, etc. within the second kind, every node works severally toward a goal native thereto, for instance, to enhance the time interval of a neighborhood task. Dynamic load scheduling algorithms of distributed nature, typically generate additional messages than the non-distributed ones as a result of, every of the nodes within the system must act with each different node. A benefit, of this can be that though one or additional nodes within the system fail, it'll not cause the entire load scheduling method to halt, it instead would have an effect on the system performance to some extent. Distributed dynamic load scheduling will introduce large stress on a system during which every node must interchange standing info with each different node within the system. it's additional advantageous once most of the nodes act severally with only a few interactions with others In non-distributed kind, either one node or a gaggle of nodes do the task of load reconciliation. Non-distributed dynamic load reconciliation algorithms will take 2 forms: centralized and semi-distributed. within the 1st kind, the load reconciliation formula is dead solely by one node within the whole system: the central node. This node is exclusively chargeable for load scheduling of the full system. the opposite nodes act solely with the central node. In semi-distributed kind, nodes of the system square measure partitioned off into clusters, wherever the load reconciliation in every cluster is of centralized kind. A central node is nonappointive in every cluster by applicable election technique that takes care of load scheduling at intervals that cluster. Hence, the load scheduling of the full system is finished via the central nodes of every cluster [4]. Centralized dynamic load reconciliation takes fewer messages to succeed in a choice, because the range of overall interactions within the system decreases drastically as compared to the semidistributed case. However, centralized algorithms will cause a bottleneck within the system at the central node and additionally the load reconciliation method is rendered useless once the central node crashes. Therefore, this formula is most fitted to networks with little size.

IV. DISTRIBUTED LOAD SCHEDULING FOR THE CLOUDS

Introduction In complicated and huge systems, there's an incredible would like for load scheduling. For simplifying load reconciliation globally (e.g. during a cloud), one issue which may be done is, using techniques would act at the elements of the clouds in such the way that the load of the full cloud is balanced. For this purpose, we tend to square measure discussing 3 varieties of solutions which may be applied to a distributed system [7]: Apis mellifera hunt formula, a biased sampling on a stochastic process procedure and Active bunch.

V. PROBLEM FORMULATION AND OBJECTIVES

In the past, variety of load reconciliation formulas are developed specifically to suit the dynamic cloud computing environments like INS (Index Name Server) algorithm, WLC (Weighted Least Connection) formula, LBMM (Load reconciliation Min-Min) formula, ACO(Ant Colony Optimization) formula and Bee-MMT(Artificial Bee Colony algorithm- negligible Migration time). we tend to square measure aiming to use the Active bunch and Resource Aware planning Algorithm(RASA) for load reconciliation in dynamic cloud environments and compare it with hymenopteran Colony optimisation (ACO) formula. Performance of Active bunch and RASA has additionally been approved higher in distributed system than ACO.

VI. METRICS FOR LOAD RECONCILIATION IN CLOUDS

Various metrics are going to be thought of in load reconciliation techniques in cloud computing square measure mentioned below

1. output is employed to calculate the no. of tasks whose execution has been completed. It ought to be high to enhance the performance of the system.
2. Overhead Associated determines the quantity of overhead concerned whereas implementing a load-balancing formula. it's composed of overhead thanks to movement of tasks, inter-processor and inter-process communication. this could be decreased so a load scheduling technique will work with efficiency.
3. Fault Tolerance is that the ability of Associate in Nursing formula to perform uniform load scheduling in spite of absolute node or link failure. The load scheduling ought to be an honest fault tolerant technique.
4. time interval is that the quantity of your time taken to retort by a specific load scheduling formula during a distributed system. This parameter ought to be decreased .
5. Resource Utilization is employed to visualize the employment of re-sources. It ought to be optimized for Associate in Nursing economical load scheduling.
6. measurability is that the ability of Associate in Nursing formula to perform load scheduling for a system with any finite range of nodes. This metric ought to be improved.
7. Performance is employed to visualize the potency of the system. This must be improved at an affordable price, e.g., cut back task time interval whereas keeping acceptable delays.

VII. HONEY BEES HUNT BEHAVIOR

This formula springs from the behavior of honey bees for locating and reaping food. there's a category of bees known as the forager bees that forage for food sources, upon finding one, they are available back to the beehive to advertise this employing a dance known as waggle dance. The show of this dance, provides the thought of the standard or amount of food and additionally its distance from the beehive. Scout bees then follow the foragers to the placement of food and so began to reap it. They then come to the beehive and do a waggle dance, which supplies a concept of what proportion food is left and thence ends up in additional exploitation or abandonment of the food supply. just in case of load scheduling, because the webservers demand will increase or decreases, the services square measure appointed dynamically to control the dynamic demands of the user. The servers square measure classified below virtual servers (VS), every VS having its own virtual

service queues. every server process a call for participation from its queue calculates a profit or reward, that is analogous to the standard that the bees show in their waggle dance. One live of this reward is the quantity of your time that the {cpu|central process unit|CPU|C.P.U.|centralprocessor|processor|mainframe|electronicequipment|hardware|computer hardware} spends on the processing of a call for participation. The flooring just in case of honey bees is analogous to an advertisement board here. This board is additionally wont to advertise the profit of the whole colony. every of the servers takes the role of either a forager or a scout. The server when process a call for participation will post their profit on the advert boards with a chance of pr. A server will opt for a queue of a VS by a chance of post exchange showing forage/explore behavior, or it will check for advertisements (see dance) and serve it, therefore showing scout behavior. A server serving a call for participation, calculates its profit and compare it with the colony profit and so sets its post exchange. If this profit was high, then the server stays at this virtual server; posting a billboard for it by chance pr. If it had been low, then the server returns to the forage or scout behavior.

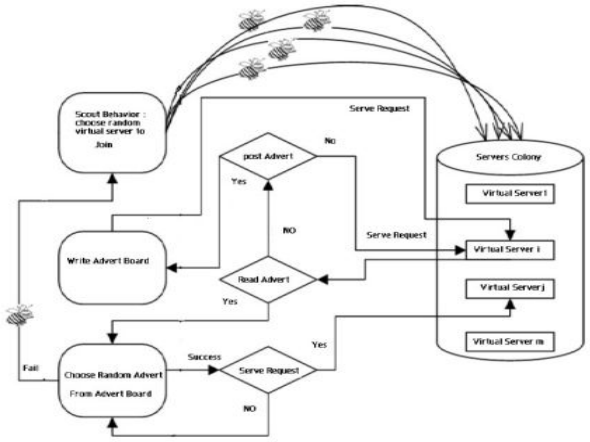

Figure 1: Server Allocations by hunt in Honey bee technique (adopted from [7])

A colony of honey bee will extend itself over long distances on notice several food sources like flower patches and so these bees harvests nectar or spore from these sources. alittle fraction of the colony finds the setting longing for national capital patches. once food supply is encountered the scout bees come in the sector close the hive and check for quality helpful. after they come to the hive, the scouts collect the food harvested. there's a vicinity within the hive known as because the “dance floor”, wherever waggle dance is performed by the bees that found a awfully helpful food. Through the waggle dance a scout bee passes the position of its search to idle spectator, that helps within the victimization of the flower patch. Here the length of the dance is in keeping with the scout’s rating of the food supply, to reap the simplest rated flower patches additional foragers get recruited. once dance is finished, the scout come to the food supply it found to check additional food. until the food is profitable, food sources are going to be announce by the scouts after they come to their hive. Foragers United Nations agency square measure recruited recently might waggle dance moreover, which can increase the accomplishment for extremely profitable flower patches. This contact action method can persist to search out most helpful flower patches [14].

In cloud computing, load scheduling is needed to distribute the dynamic native work equally across all the nodes. It helps to attain a high user satisfaction Associate in Nursingd resource utilization magnitude relation by guaranteeing an economical and honest allocation of each computing resource. correct load scheduling aids in minimizing resource consumption, implementing fail-over, sanctionative measurability, avoiding bottlenecks and over provisioning etc.

- Input: - needed parameters for cloudlets and VM’s square measure taken from user.
- Output: - Improves load scheduling at cloud with higher time interval, processing time and throughput.

Bee colony optimisation truly, there square measure numerous natural systems (i.e. social insects colonies) like the Ants Colony and also the Bees Colony during which easy individual organisms will produce systems that square measure able to perform extremely complicated tasks by dynamically interacting with one another. In general, the honey bee colony consists of 3 sorts of adult bees: staff, drones, and a queen. though every member within the honey bee colony features a definite task to perform, lots of employee bees have to be compelled to get together to complete complicated jobs, like nest building, food finding and assortment, and brood rearing. additionally, individual bees (workers, drones, and queens) cannot survive while not the support of the colony. Therefore, living and reproducing have to be compelled to mix the efforts of the whole colony.

For the forage, the honey bee colony can by selection forage from nectar sources out there within the field. the method is initiated by scout bees being sent to look for promising flower patches. Scout bees move willy-nilly from one patch to a different. Scout bees after they come to the hive, those scout bees that found a patch that is rated on top of an exact quality threshold deposit their nectar or spore and attend the ‘‘dance floor’’ to perform a dance referred to as the ‘‘waggle dance’’. This mysterious dance is crucial for colony communication, and contains 3 items of knowledge relating to a flower patch: the direction during which it'll be found, its distance from the hive and its quality rating (or fitness) [2]. Authors in [2,18] summarize the construct of the Bee Colony optimisation whereas authors in [6,19,20] justify it in additional details. every bee can follow a nestmate United Nations agency has already discovered a patch of flowers. Upon arrival, the hunt bee takes a load of nectar and returns to the hive relinquishing the nectar to a food storer bee. when she relinquishes the food, the bee will

- Abandon the food supply and become once more uncommitted follower,
- Continue to forage at the food supply while not recruiting the nestmates, or
- Dance and therefore recruit the nestmates before the come to the food supply.

The bee opts for one among the on top of alternatives with an exact chance. at intervals the dance space, the bee dancers ‘‘advertise’’ completely different food areas. The mechanisms by that the bee decides to follow a selected dancer aren't well understood, however it's thought of that ‘‘the accomplishment among bees is often a perform of the

standard of the food source''. it's additionally noted that not all bees begin hunt at the same time [18].
Bee Colony optimisation (BCO) [18] could be a metaheuristic capable to unravel tough combinatorial optimisation issues. The metaheuristic could be a swarm intelligence approach, which means it's characterised by people doing repetitive actions and an easy communication technique between people, leading to repetitive improvement of answer quality. Therefore, organisation of bees is predicated on some comparatively easy rules of individual insect's behavior [19]. The authors in [19] and [20] were used the collective bee intelligence in resolution combinatorial optimisation issues.

VIII. PROPOSED FLOW CHART FOR LOAD SCHEDULING

In this analysis work proposes a flow chart that describes the mechanism of load scheduling among full virtual machines and underloaded virtual machines. The flow chart is impressed by the Honey Bee Behaviour approach to balance the load in cloud computing. The key plan is to submit the tasks to the virtual machine until the machine gets full i.e. load thereon virtual machine become quite threshold worth .The threshold worth could also be thought of seventy five. we tend to don't submit tasks to the full machine and that we send the remaining tasks to underloaded virtual machines which will be finding by willy-nilly SEARCH technique. Here in flow chart Cloudlets square measure thought of as tasks/jobs that user requests to the Virtual Machine (VM).

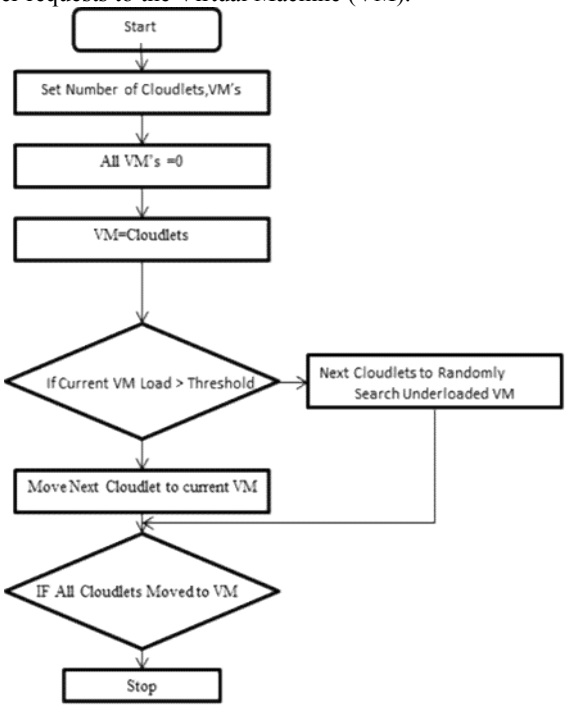

Fig.2 planned Flow Chart for Honey Bee Behavior approach

CONCLUSION

Load scheduling is a vital task in Cloud Computing setting to attain most utilization of resources. during this paper, we tend to mentioned Honey Bee shaping load scheduling schemes. The tasks square measure to be send to the underloaded machine and like hunt bee subsequent tasks also are sent thereto virtual machine until the machine gets full as flower patches exploitation is finished by scout bees. Honey bee behavior impressed load scheduling improves the output of process and priority based mostly scheduling focuses on reducing the quantity of your time a task must assist a queue of the VM. Thus, it reduces the response of your time of VMs. we've compared our planned formula with different existing techniques.

REFERENCES

[1] Chen Wu, Tharam Dillon and Elizabeth Chang," Cloud Computing: Issues and Challenges" *24th IEEE International Conference on Advanced Information Networking and Applications* 2010.
[2] ,Junjie Pang, Gaochao Xuand ,Xiaodong Fu, Tsinghua "Load balancing model based on Cloud partitioning for the public cloud", *science and technology* 2013,vol.18, pp34-39.
[3] Der-Jiunn Deng,Hui-Hsin Chin, and Chun-Cheng Lin, IEEE members, "*Dynamic Multiservice Load balancing in Coud –Based Mutimedia System*", IEEE Systems Journal,2013.
[4] Yi Chung,Haiying Shenand Yu-Chang Chao ,HungChang, Hsueh-, " *Load Rebalancing for Distributed File System in clouds*", IEEE Transactions on Parallel And Distributed Systems.2013,vol.24, pp951-961.
[5] R. Hunter, The why of cloud, http://www.gartner.com/ DisplayDocument?doccd=226469&ref= g noreg, 2012
[6] and Dr. R. Shanka, Asst. Prof. Atesh Kumar Singh Namrata Swarnkar, A Survey of Load Balancing Techniques in Cloud Computing. Vol. 2 Issue 8, August – 2013
[7] E.D. Lazowska, D.L. Eager, J. Zahorjan, Adaptive load sharing in homogeneous 647 distributed systems, The IEEE Transactions on Software Engineering 12 (5) 648 (1986) 662–675.
[8] L.D. P. Venkata Krishna, Dhinesh Babu (2013), '*Honey bee behavior inspired Load Balancing of tasks in cloud computing environments*', ELSEVIER, vol.13, pp 2292-2303.
[9] Changjun Jiang,Shu Li ,Dongliang Zhang, "A fast adaptive load balancing method for parallel particlebased simulations", Simulation Modelling Practice and Theory 17 (2009) 1032–1042.
[10] Xiuqiao Li, Qimeng Bin Dong Wu, Limin Xiao, Li Ruan, "A dynamic and adaptive load balancing strategy for parallel file system with large-scale I/O servers", J. Parallel Distribution Computing. 72 (2012) 1254–1268.
[11] Rynson W.H. Lau, Yunhua Deng, "Heat diffusion based dynamic load balancing for distributed virtual environments", in: Proceedings of the17th ACM Symposium on Virtual Reality Software and Technology, ACM, 2010, pp. 203–210.
[12] Eric Tobias, Markus Esch, "Decentralized scale-free network construction and load balancing in Massive Multiuser Virtual Environments", in: Collaborative Computing: Networking, Applications and Worksharing, Collaborate Com, 2010, 6th International Conference on, IEEE, 2010, pp. 1–10.
[13] K. Lakshminarayanan, B. Godfrey, S. Surana, R. Karp, I. Stoica, "Load balancing in dynamic structured P2P systems", in: INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies, vol. 4, IEEE, 2004, pp. 2253–2262. About Bee Behaviour and algorithm from " http://en.wikipedia.org/wiki/Bees_algorithm"
[14] M. Malathi, "Cloud Computing Concepts", IEEE, 2011
[15] Paul, M., Sanyal, G., "Survey and analysis of optimal scheduling strategies in cloud environment", IEEE, 2012

[16] R. Vasanth, Nagaveni, N., Jeyarani, R., Ram, "Design and Implementation of an Efficient Two-Level Scheduler for Cloud Computing Environment", IEEE, 2010
[17] Huang Ting-lei, Huang Qi-yi, "An optimistic job scheduling strategy based on QoS for Cloud Computing", IEEE, 2010
[18] Haiyang Wang, Yanbing Bi, Meng Xu, Lizhen Cui,"A Multiple QoS Constrained Scheduling Strategy of Multiple Workflows for Cloud Computing", IEEE, 2009
[19] Hao Li, Huixi Li, "A Research of Resource Scheduling Strategy for Cloud Computing Based on Pareto Optimality M×N Production Model",, IEEE, 2011
[20] Meng-Hsuan Fu, Yau-Hwang Kuo, Kuan-Rong Lee, "A hierarchical scheduling strategy for the composition services architecture based on cloud computing", IEEE, 2011
[21] Byung-Gon Chunz, Randy H. Katzy, Gunho Leey, "Heterogeneity-Aware Resource Allocation and Scheduling in the Cloud", University of California
[22] Kuo-Qin Yan, Shun-Sheng Wang, Ching-Wei Chen, Shu-Ching Wang, "A Three-Phases Scheduling in a Hierarchical Cloud Computing Network", IEEE, 2011
[23] M.J., Estrella, Peixoto, M.L.M., Santana, J.C., Tavares, T.C., Kuehne, B.T., Santana, R.H.C., "A Metascheduler architecture to provide QoS on the cloud computing", IEEE, 2010
[24] Fei Teng, "Resource allocation and scheduling models for cloud computing", Paris, 2011
[25] Xiaolin Li, Han Zhao, "AuctionNet: Market oriented task scheduling in heterogeneous distributed environments", IEEE, 2010.