INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X
Impact factor: 4.295
(Volume3, Issue4)
Available online at www.ijariit.com

Resource Allocation Utilizing Enthalpy Based Krill Herd Optimization Algorithm in Cloud Computing

Chenni Kumaran .J
Research Scholar, Dept. of Cse,
Manonmaniam Sundaranar University, Tirunelveli.
chennikumaranj988@gmail.com

Dr. M. Aramudhan
Associate Professor & Head,
Dept. of IT, PKIET, Karaikal.
aramudhanm@gmail.com

Abstract**: Cloud computing has been a sort of Internet-related computing which proffers distributed computer processing resources, data computers and other devices upon requirement. The prior methodology of resource assignment in cloud computing is accomplished through Queuing Theory Based Cuckoo Search algorithm and it is having few limitations including failure in processing plus knapsack issue. Moreover scheduling energy consumption plus the computational cost is high. Our proposed work of resource assignment through workflow scheduling is purely carried in two stages. The first stage comprises of two phases, for every available task we measure task reward, delay, transmission probability, communication cost and reputation. According to the task measure value computed, we calculate the Enthalpy values. In the Second stage of our proposed work, we employ enthalpy based krill herd optimization algorithm for allocating resources that increase trade makespan and for minimizing the resource usage. It also reduces computational complexity by enhancing the computing efficiency of processing elements. The implementation of our suggested algorithm reduces the knapsack issue of energy consumption, VM usage, PM usage, computational time, task migration and resource utilization which proffers to cost reduction.

Keywords: Cloud Computing, Resource Allocation, Enthalpy, Krill Herd, Task Measure, Scheduling, Queuing Theory, Cuckoo Search.

I. INTRODUCTION

The progress of cloud computing has provoked several explorations in the retrieval of information for huge data informatics in contemporary years [4]. The design aim of the clouds was proffering distributed resources to multi-tenants and maximizes the efficacy and cost [1]. Cloud computing assists developers plus companies for overcoming the lack of hardware capability by allowing the user to access on-demand resources by the Internet [2]. Mainly, The cloud computing proffers three sorts of services viz: Infrastructure as a Service (IaaS), Software as a Service (SaaS) and Platform as a Service (PaaS), [6]. It has created a competitive market where consumers pay providers for using resources and are usually billed using a pay-as-you-go model. Resource Allocation (RA), in cloud computing, is the methodology of apportioning possible resources to the required cloud applications through the internet [15]. Resources in the cloud are usually geographically distributed; may be heterogeneous and possessed by several organizations with variant utilization and also cost policies [3]. The availability of these distributed resources also facilitated the adherence to existing business regulations (e.g., privacy and security) which impose geographical constraints on the storage and management of their data [5,14]. Efficient task scheduling and resource management is a challenging issue of distributed computing [10]. Since the issue of task scheduling on Cloud Computing environments is rectified by utilizing various Genetic Algorithms, PSO (Particle Swarm Optimization) [17], Ant Colony Optimization, and CSO (Cat Swarm Optimization algorithms). However, the issue is still tough because of the highly powerful environment in Cloud Computing plus to the variability plus confliction of the users' requisitions [16]. There are two significant categories for scheduling and resource allocation approaches that can be found in the literature; throughput maximization [12] and delay minimization [13]. When a request is submitted by the client, it's firstly divided into numerous subtasks and then scheduling of resources and its deployments will be carried out efficiently [7]. The number of requests is based on the number of VM usage such that one VM cannot meet both great performance and also low price [9]. VM is managed by its VM manger that requests permission to get resources from the resource provider and also resource provision [11]. Once when the permission is granted from a resource provider, the resource provisioner offers resources to VM manager according to least cost of resources [18]. Application-level optimization techniques along with topology based VM placement overture can be carried out to offer better chances of performance improvement [8]. The former method of resource allocation through Queuing

Theory Based Cuckoo Search algorithm is having some drawbacks such as failure processing and knapsack problem of resource allocation which increases the energy absorption and hence the computational cost is high [2].

II. RELATED WORK

Xinjie Guan *et.al* [19] have designed a novel application oriented Docker container (AODC)-based resource allocation framework to minimize the application deployment cost in data centers and to support automatic scaling while the workload of the cloud applications varies. Also, they modeled the AODC resource assignment issue regarding components of Docker, various applications' requirements, and available resources in cloud data centers, and proposed a scalable algorithm for data centers with diverse and dynamic applications and massive physical resources.

Alok Gautam Kumbhare *et.al* [20] have developed the concept of "dynamic data flows" which applies for alternating works as extra power over the data flow's cost plus QoS. Furthermore, they generated an optimization issue for representing execution and runtime resource provisioning which permits us for balancing the QoS application value and the resource cost. Also, they pontificate two greedy heuristics, centralized plus shared, regarding variable-sized bin packing algorithm then comparison opposite to a Genetic Algorithm (GA) related heuristic which proffers an adjacent-optimal solution.

Bhaskar Prasad Rimal and Martin Maier [21] have suggested an innovative cloud-related workflow scheduling (CWSA) policy to compute-intensive workflow applications in environments of numerous tenant cloud computing, that assists to reduce the complete workflow accomplishment time, cost of implementation of the workflows, tardiness, and also utilize inactive cloud resources efficiently. Moreover, a proof-of-concept analysis of applications of actual-world scientific workflow was implemented for demonstrating the measurability of the CWSA, which validates the efficacy of the suggested solution.

Sukhpal Singh *et.al* [22] have proffered a systematic literature evaluation of resource handling in cloud computing, including resource provisioning, resource scheduling, and autonomic resource provisioning plus schedule. They elucidated the recent status of resource handling in cloud computing and have proffered further evaluation of its methodologies as enhanced by the variant industry as well as academic groups.

Xiaolong Xu *et.al* [23] have proposed an Energy-aware Resource Assignment technique, termed EnReal, for addressing the above challenge. Fundamentally, they leverage the powerful execution of virtual machines to scientific workflow implementations. Particularly, an energy consumption model was provided for applications utilized through cloud computing platforms, and also an associating energy-aware resource assignment algorithm was suggested for scheduling of virtual machine for accomplishing scientific workflow implementations.

III.PROPOSED METHODOLOGY

Our proposed effort of resource allocation through workflow scheduling is purely based on two stages. The first stage comprises of two phases, we measure task reward**,** delay**,** transmission probability**,** communication cost and reputation for every possible task. According to the task measure values computed, we compute the Enthalpy value. In the 2nd stage, we employ enthalpy based krill herd optimization algorithm for allocating resources that increase trade makespan and for reducing the resource usage. It also reduces computational complexity which increases the computing capability of processing elements. The implementation of our suggested algorithm reduces the knapsack problem (energy consumption, PM usage, VM usage, computational time, task migration) that yields to the reduction of cost. Our intended scheme comprised of totally three phases namely, i) Computation of task measure values, ii) Enthalpy computation, iii)Enthalpy optimization utilizing KH. The architecture of our suggested Enthalpy-based KH Optimization is presented in Figure 1.

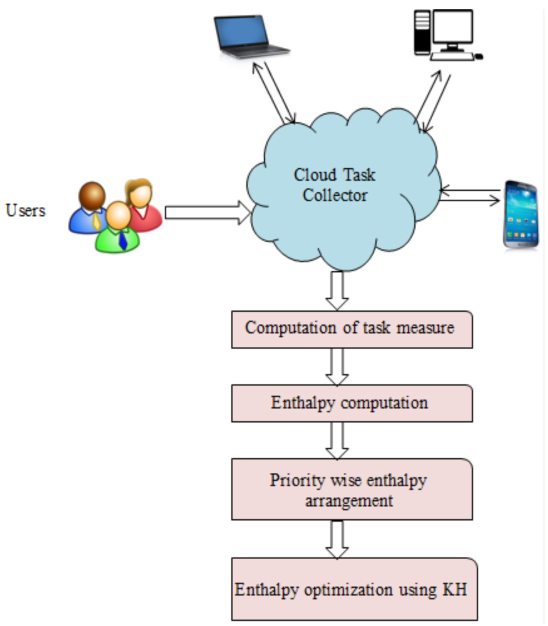

Fig 1: Architecture of our suggested Enthalpy-based KH Optimization

3.1 Computation of Task Measure Values

Task scheduling algorithm is a technique by which tasks are compared, or assigned to data center resources. Due to conflict scheduling goals, absolutely no perfect scheduling algorithm exists. For evading this situation the calculation of task measure values is implemented. In this phase, the number of the task is computed individually by task reward**,** delay**,** transmission probability**,** communication cost and reputation.

3.1.1Task Reward

The computation duration of a task is separated into two parts: a mandatory part p and an optional part r .Here, Mandatory part have to attain its deadline for proffering less agreeable service, while the optional part is implemented for extra enhancement of quality if the system resource still exists. Such enhancement is signified through the reward function T_R. While a job of r implements t time units of its optional part then the reward $T_R(t)$ is signified as follows.

$$T_R(t) = \begin{cases} h_r(t) \text{ if } 0 \le t \le lr \\ h_r(lr) \text{ if } t > lr \end{cases} \tag{1}$$

Whatever $T_R(t)$ is a no decreasing function; thus, the reward of a task r cannot decrease by permitting the task to run longer. Likewise, the reward accrued by a task r halts to increase when its optional part's implementation time t attains its optimum value l_r.

3.1.2 Delay

A network delay denotes how long it takes for a bit of data to travel across the network from one node or else endpoint to another. It is measured typically in multiples or else fractions of the seconds. Delay may slightly vary, regarding the location of the particular pair of communicating nodes. Also, Delay time is defined as the addition of extra time taken to complete the task and actual or correct time taken to complete the task within allotted time. It is represented as follows,

$$D_T = E_T + C_T \tag{2}$$

Whereas, D_T denotes the delay time, E_T denotes additional time taken for accomplishing the task and C_T denotes accurate time taken for accomplishing the task within assigned time.

3.1.3 Transmission Probability

The performance of any network can be characterized by successful transmission probability, which is the probability that a receiver can catch the packets effectively from its transmitter in a shot.

A wave packet with probability amplitude V_1 hits a potential barrier. A part of the wave is reflected back and the rest will be transmitted through the barrier. The probability amplitude of the transported part is signified by V_2. The transmission possibility $|P_T|^2$ is denoted as the square of the ratio betwixt transferred and also incoming probability amplitudes.

$$|P_T|^2 = \frac{|V_2|^2}{|V_1|^2} \tag{3}$$

In which, V_1 denotes the transmitted probability amplitude, V_2 denotes the incoming probability amplitude.

3.1.4Communication Cost

Communication cost of a service measures the resource usage during its execution time and might refer to the service fees. This is inversely proportional to the general quality of service. Higher the costs then lower will be the Quality of Service.

User allocates the task to cloud coordinator (CC) through an agent which acts as intermediate. CC segments the task into same sized cloudlets and sends to the data center. Every data center will be having its own Virtual Machine Manager (VMM) which manages the Virtual Machine Pool (VMP). VMM Appeals for the resources to the Resource Provisioner (RsP) and also it then further appeals for accessing permission from the Resource Provider (RP) and it will be responding back to the VMM. When the permissions are granted from RP, the RsP proffers resources according to less cost of resources to VMM for creating VM. The communication cost is computed by utilizing the below expression.

$$T_C = aC_H + bD_N + cB + dC_S \tag{4}$$

While, T_C signifies the communication cost, C_H is Hop Count, D_N is Network Delay, B is Bandwidth and C_S is Security Cost and a , b , c , d are the variables respectively.

3.1.5 Reputation

Reputation mechanisms are broadly embraced for trust handling during resource assignment. Regarding cloud computing, the reputation value of a cloud service denotes the expansion to which clients may believe the service, according to feedback aggregation proffered by former consumers. Reputation is computed by utilizing the subsequent expression.

$$P(w) = \frac{\sum_{z=1}^{N} q(z,w) * T_Y}{N} \tag{5}$$

Whereas, $P(w)$ is the value of the local reputation of cloud service T_C, $q(z,w)$ is the feedback to the w^{th} cloud service from the z^{th} cloud consumer, T_Y is the credibility of the z^{th} cloud consumer, which is computed according to similarity betwixt the requestor to the w^{th} cloud service.

3.2Enthalpy Computation

Enthalpy is a methodology to atomize an application into smaller forms that support bigger data and meanwhile satisfies what user requires most. It is possible in achieving the same level of quality of service as provided by the application before atomized.

$$E = Z + qY \tag{6}$$

Where, E is the enthalpy, Z is the enthalpy of original system before being serviced, q is the data required by the system and Y is the processing time to run the system.

The change of enthalpy ΔE is the one that can be measured as defined in the subsequent equation

$$\Delta E = E_{final} - E_{initial} \tag{7}$$

Where, ΔE is the enthalpy change, E_{final} is the enthalpy once the system has been atomized into smaller services, and $E_{initial}$ is the enthalpy of the service before being atomized.

If the response time, data and the number of functionalities reduces, its enthalpy change ΔE will be negative, and if the response time, data and number of functionalities increase then ΔE will be positive. Performance improvement in enthalpy uses one metric known to be the quality of service (QoS), i.e. processing time. The average of processing time is used as enthalpy modification of performance improvement for creating a new atomic service. However, such enthalpy of performance improvement is useful to avoid timeout that usually occurs during transfer of big data between services.

3.3 Enthalpy-based KH Optimization

Antarctic krill (Euphausia superba) is a species of krill found in the Southern Ocean. These species consume phytoplankton. A differentiating component of krill is the capability for individual krill for being molded amidst a huge group of organisms that are even hundreds of meters in length. The process of KH algorithm starts with the theory of enthalpies, i.e. selected enthalpy and a total number of enthalpies. However, initialization of enthalpy represents a krill means situating it at the solution space by giving it a set of coordinates. The enthalpy based KH Algorithm deducts the energy consumption, the amount of VM usage, computational time, PM usage, resource utilization, task migration and also demonstrating that enthalpy related KH is enhancing the performance of the allocating resources in cloud computing.

3.3.1 Steps for Enthalpy-Based KH Optimization Algorithm

The Enthalpy-based Krill Herd Algorithm is presented as follows,

Step1: Initially assume the total number of enthalpies as K and iterations as J_{mx}.

Step2: Randomly generate the initial enthalpy value M_A, where A = 1, 2, 3....K individual enthalpies respectively. Establish the following parameters for the subsequent functions:

- Foraging Speed R_A
- Maximum induced speed $M_{\max}$
- Maximum iteration number J_{mx}

Step3: Compute the fitness function through determining all individual enthalpies based on its present position.

Step4: Analyze the motion by considering the three elements which are mentioned below:

i) Accordance with other Enthalpy values

Selection of the j^{th} individual enthalpy, such that M_j can be defined as

$$M_j^{new} = M_{\max} Z_j + \Omega_s M_j^{old} \tag{8}$$

Where, $M_{\max} \rightarrow$ maximum induced speed, $M_j^{old} \rightarrow$ previously induced motion, $\Omega_s \rightarrow$ Inertia weight,

$$Z_j = Z_j^{\ L} + Z_j^T \tag{9}$$

$Z_j^L \rightarrow$ Local effect delivered by the vicinity enthalpy value, $Z_j^T \rightarrow$ Target effect delivered by the vicinity enthalpy values.

ii) Foraging motion

Foraging motion is nothing but the motion induced to an individual enthalpy due to the presence of resources and its localities in the explanation. The foraging motion value for the j^{th} enthalpy is estimated by the succeeding equation:

$$R_j = R_f \mu_j + \Omega_f R_j^{\ old} \tag{10}$$

Where, $R_j \rightarrow$ Foraging Motion, $R_f \rightarrow$ Foraging Speed, $\Omega_f \rightarrow$ Inertia weight of the foraging motion, $R_j^{old} \rightarrow$ Last foraging motion value

$$\mu_j = \mu_j^{\ evalue} + \mu_j^{best} \tag{11}$$

While, $\mu_j \rightarrow$ Effect of enthalpy values, $\mu_j^{\ evalue} \rightarrow$ Effect of available enthalpy values, $\mu_j^{best} \rightarrow$ Effect of best enthalpy value

iii) Physical diffusion

This is nothing but the random diffusion of the enthalpies in the solution space. The enthalpy computation is estimated from the speed of the enthalpies and a randomly created directional vector. The expression to compute this diffusion effect of enthalpy is signified as follows:

$$S_j = S_{\max} \left({(J_{mx} - J)} \big/ {J_{mx}} \right) \tau \tag{12}$$

Whereas, $S_{\max} \rightarrow$ Maximum diffusion speed, $\tau \rightarrow$ Random directional vector between [-1, 1]

Step5: Suppose, if the situation of the optimization issue is not satisfied, then move to step3.

Step6: Improve the new positions of the selected enthalpy among the available total enthalpies correspondingly,

Employing all the efficient parameters (M_j, R_j, S_j) of the motion attained over time, the random position of the enthalpy during the interval f and Δf can be completed by the subsequent equation and the position of the each enthalpy value is updated. According to uniform distribution function,

$$(f + \Delta f) = X_j(f) + \Delta f.(dX_j / df) \tag{13}$$

$$X_j = M_j + R_j + S_j \tag{14}$$

Whereas, Δf, is termed as a Scale factor,

Step7: Suppose, if the termination criterion is not met, then move to step3 and repeat the process duly.

Step8: Else, if the termination criterion is satisfied, then determine the finest possible solution in the search space.

IV. RESULT AND DISCUSSION

Our proposed overture is utilized to decrease the knapsack problem of energy consumption, resource utilization, the number of Virtual Machine usage, Primary Memory usage, computational time, task migration, and hence the computational cost is getting reduced automatically. However, the performance of resource allocation i.e., the Quality of Service (QoS) is excellent due to a reduction in aforementioned parameters. Our suggested methodology is engaged in the the.NET platform with machine configuration as follows: Processor: Intel core i3OS: Windows 7CPU Speed: 3.20 GHz RAM: 4GB. The performance of the proposed technique is related to the former QTCS algorithms.

4.1 Energy Consumption

Energy consumption of servers depends on a CPU usage, network card, memory, and disk. CPU has been the major energy consumption unit and hence the CPU utilization of a server typically signifies its resource utilization. The results of energy consumption have to be minimum for the efficient algorithm. Comparison analysis of our suggested Enthalpy-based KH is explained in Figure 2.

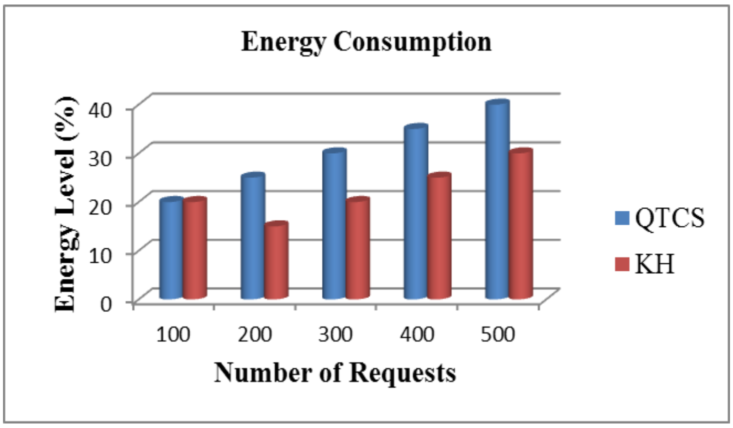

Fig 2: Comparison analysis of proposed KH with QTCS for energy consumption

When the number of request increases, then the energy absorption will considerably decrease as per graphical observation. Thus, our proposed Enthalpy-based KH diminishes the energy conception, thereby providing improved performance than the former QTCS algorithm outcomes.

4.2 Virtual Machine Usage

An efficient system should have less number of VM and we varied the number of virtual machine requests from 10 to 30 with a step value of 10. Figure 3 explains the VM usage comparison of Enthalpy-based KH to that of former QTCS algorithm. From the Fig it is observed that amount of VM usage is considerably reduced while using Enthalpy-based KH algorithm than the former technique of QTCS.

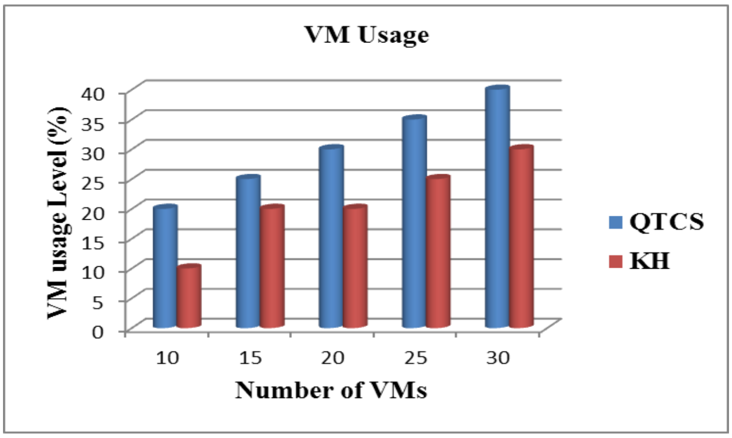

Fig 3: Comparison analysis of proposed KH with QTCS for VM usage

4.3 Primary Memory Usage

Primary Memory (PM) utilizations have to be minimized to attain a better performance. When the number of request from VM increases for example, in the range of 10, 15, 20 then the PM usage is decreasing which is lesser than the previous approach of QTCS.

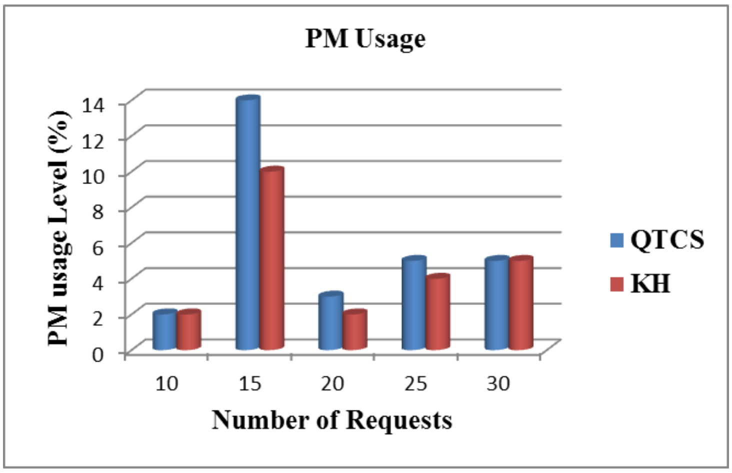

Fig 4: Comparison analysis of proposed KH with QTCS for PM usage

Figure 4 shows the comparison of the observed outcomes of QTCS and KH optimization with respect to PM usage and it is observed that the usage of PM is considerably getting reduced when the number requests increased, thus proves that our Enthalpy-based KH algorithm proffers improved performance than earlier QTCS technique.

4.4 Resource Utilization

The resource usage of our suggested task is designated by the comparison results of Enthalpy-based KH algorithm with former QTCS and it is presented in Figure 5. Here, the result is estimated by mapping the number of virtual machine requests against the utilization time.

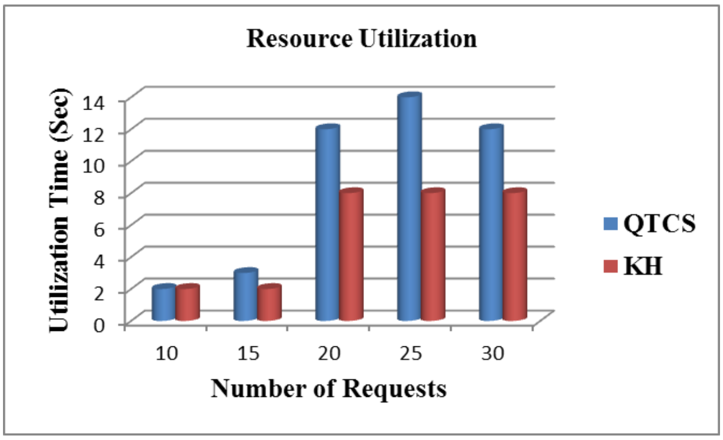

Fig 5: Comparison analysis of proposed KH with QTCS for resource utilization

From the figure, it is observed that our Enthalpy-based KH utilizes minimum resources with the increase in arrival instants of time partitions than that of erstwhile QTCS algorithm.

4.5 Computational Time

Computational duration of the desired approach has to be minimized for accomplishing excellent performance within less time. From the Figure 6 it is observed that when the number of request from VM is increasing then the computational time is getting reduced considerably while employing KH approach.

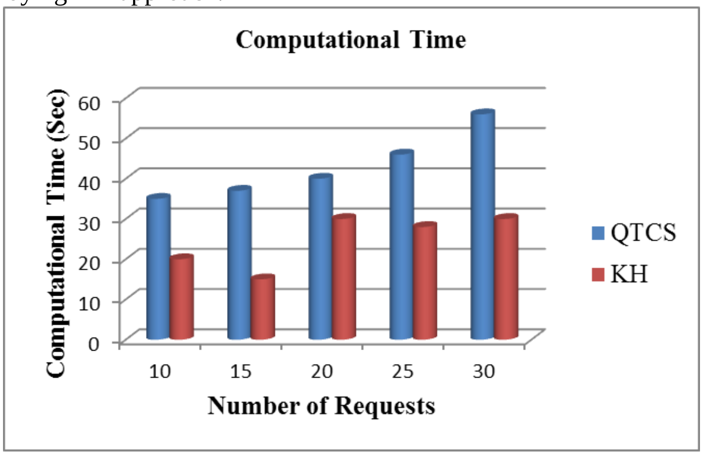

Fig 6: Comparison analysis of proposed KH with QTCS for computational time

This indicates that our approach attained reasonable scalability with an approximately optimal solution than that of previous QTCS.

4.6 Task Migration

The comparison between KH and QTCS is presented in Figure 7. with respect to task migration. Task migration is directly related to time period for accomplishing the task, since from the below figure when the number of request from VM is increasing i.e., for instance, 10, 15, 20 etc. Then the time taken for completing the task is considerably reducing as per the observation.

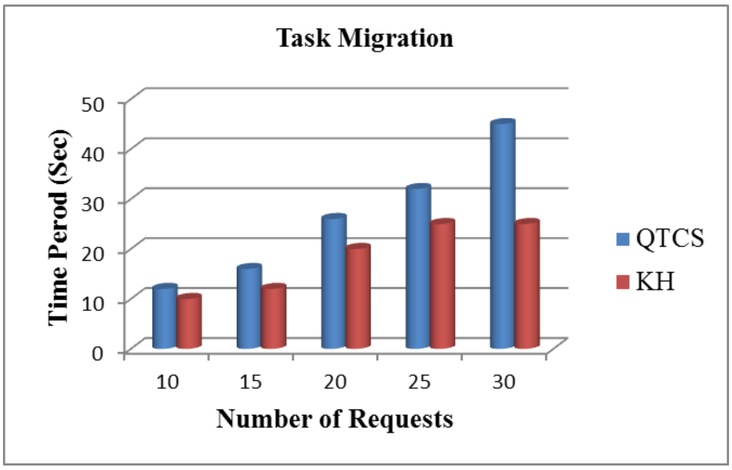

Fig 7: Comparison analysis of proposed KH with QTCS for task migration

Thus the task migration is minimal while utilizing KH optimization which is massive in earlier QTCS optimization.

CONCLUSION

In our suggested methodology of resource allocation, we first compute task measure values and according to such computed values, we calculate the enthalpy values. Then by utilizing enthalpy based KH optimization algorithm, we can achieve excellent performance over previous approaches. Since the former approach of resource assignment in cloud computing is utilizing Queuing Theory Based Cuckoo Search algorithm yet it is having few limitations including failure in processing and also knapsack issue. Also, the scheduling energy consumption and computational cost are high. Hence for overcoming these disadvantageous, we engage an enthalpy related krill herd optimization algorithm to assign resources that augment trade makespan and also to diminish the resource utilization. It also reduces the computational complexity by improving the computing capability of processing features. The implementation of our recommended algorithm diminishes the knapsack issue of energy consumption, PM usage, VM usage, computational time, task migration and resource utilization that automatically results in cost reduction.

REFERENCES

[1] I. Sadooghi, J. H. Martin, T. Li, K. Brandstatter, Y. Zhao, K. Maheshwari, T. P. P. d. L. Ruivo, S. Timm, G. Garzoglio, and I. Raicu,"Understanding the performance and potential of cloud computing for scientific applications, " IEEE Transactions on Cloud Computing, Vol. 5, No. 2, pp. 358 - 371, 2015.

[2] S. Vakilinia, B. Heidarpour, and M. Cheriet, “Energy Efficient Resource Allocation in Cloud Computing Environments”, IEEE Access, Vol. 4, pp.8544 - 8557, 2016.

[3] X. Wang, X. Wang, H. Che, K. Li, M. Huang, and C. Gao, "An intelligent economic approach for dynamic resource allocation in cloud services", IEEE Transactions on Cloud Computing, Vol.3, No. 3, pp. 275-289, 2015.

[4] K. Gai, M. Qiu, and Hui Zhao, "Cost-aware multimedia data allocation for heterogeneous memory using a genetic algorithm in cloud computing", IEEE Transactions on cloud computing, No. 99, pp. 1-11, 2016.

[5] B. Wanis, N. Samaan, and A. Karmouch, "Efficient modeling and demand allocation for differentiated cloud virtual-network as-a-service offerings", IEEE Transactions on Cloud Computing, Vol. 4, No. 4, pp. 376-391, 2016.

[6] C. Awasthi, and P. Kanungo, "Resource allocation strategy for cloud computing environment," In Computer, Communication and Control (IC4), 2015 International Conference, pp. 1-5, 2015.

[7] S. Goutam, and A. K. Yadav, “Preemptable priority based dynamic resource allocation in cloud computing with fault tolerance”, In Communication Networks (ICCN), 2015 International Conference on, pp. 278-285, 2015.

[8] P. S. Pillai, and S. Rao, "Resource allocation in cloud computing using the uncertainty principle of game theory", IEEE Systems, Vol. 10, No. 2, pp. 637-648, 2016.

[9] W. Dai, L. Qiu, A. Wu, and M. Qiu, "Cloud Infrastructure Resource Allocation for Big Data Applications", IEEE Transactions on Big Data, No. 99, pp.1-10, 2017.

[11] G. Portaluri, D. Adami, A. Gabbrielli, S. Giordano, and M. Pagano, "Power Consumption-Aware Virtual Machine Allocation in Cloud Data Center", In Globecom Workshops (GC Wkshps), 2016 IEEE, pp. 1-6, 2016.

[12] H. Al-Tous, and I. Barhumi, "Resource allocation for multiple-sources single-relay cooperative communication OFDMA systems", IEEE Transactions on Mobile Computing, Vo15, No. 4, pp. 964-981, 2016.

[13] Y. Cui, V. K .N Lau, and E. Yeh, "Delay optimal buffered decode-and-forward for two-hop networks with random link connectivity", IEEE Transactions on Information, Vol. 61, No. 1, pp. 404-425, 2015.

[14] N. Grozev, and R. Buyya, "Multi-cloud provisioning and load distribution for three-tier applications”, ACM Transactions on Autonomous and Adaptive Systems (TAAS), Article. 13, Vol. 9, No. 3, pp.1-21, 2014.

[15] V. P. Anuradha, and D. Sumathi, "A survey on resource allocation strategies in cloud computing", In Information Communication and Embedded Systems (ICICES), 2014 International Conference on, Vol.3, No.6, pp. 1-7, 2014.

[16] K. Bousselmi, Z. Brahmi, and M. M. Gammoudi, "QoS-aware scheduling of Workflows in Cloud Computing environments", In Advanced Information Networking and Applications (AINA), 2016 IEEE 30th International Conference on, pp. 737-745, 2016.

[17] M. A. Rodriguez, and R. Buyya, "Deadline based resource provisioning and scheduling algorithm for scientific workflows on clouds", IEEE Transactions on Cloud Computing, Vol. 2, No. 2, pp. 222-235, 2014.

[18] P. Sun, Y. Dai, and X. Qiu, "Optimal Scheduling and Management on Correlating Reliability, Performance, and Energy Consumption for Multiagent Cloud Systems," IEEE Transactions on Reliability, Vol. 66, No. 2, pp. 547 - 558, 2017.

[19] X. Guan, X. Wan, B. Y. Choi, S. Song, and J. Zhu, "Application-Oriented Dynamic Resource Allocation for Data Centers Using Docker Containers," IEEE Communications Letters, Vol. 21, No. 3, pp. 504 - 507, 2016.

[20] A. G. Kumbhare, Y. Simmhan, M. Frincu, and Viktor K. Prasanna, "Reactive resource provisioning heuristics for dynamic data flows on cloud infrastructure, " IEEE Transactions on Cloud Computing, Vol. 3, No. 2, pp. 105-118, 2015.

[21] P. B. Rimal, and M. Maier, "Workflow Scheduling in Multi-Tenant Cloud Computing Environments", IEEE Transactions on Parallel and Distributed Systems, Vol. 28, No. 1, pp.290-304, 2017.

[22] S. Singh, I. Chana, and M. Singh, "The Journey of QoS-Aware Autonomic Cloud Computing", IT Professional, Vol. 19, No. 2, pp. 42-49, 2017.

[23] X. Xu, W. Dou, X. Zhang, and J. Chen, "EnReal: An energy-aware resource allocation method for scientific workflow executions in a cloud environment", IEEE Transactions on Cloud Computing, Vol. 4, No. 2, pp.166-179, 2016.

