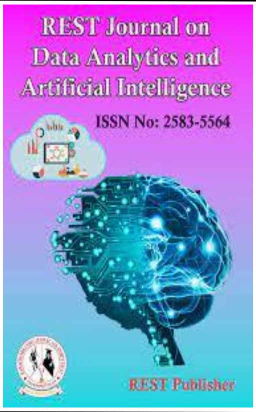
REST Journal on Data Analytics and Artificial Intelligence
Vol: 2(1), 2023
REST Publisher; ISSN: 2583-5564
Website: http://restpublisher.com/journals/jdaai/
DOI: https://doi.org/10.46632/jdaai/2/1/8



Perspicacious Fuzzy Clustering Franchised by Grasshopper Optimization Dexterity for Robust Resource Scheduling with Effective Load Balancing in Cloud Precedent

***S. Sree Priya, T. Rajendran**
Government Arts & Science College, Kangeyam, Tamilnadu, India.
*Corresponding Author Email: sspriya.sivakumar@gmail.com

Abstract. *Utilizing cloud resources become promising in encroachment of internet technology, countenancing everyone to use resources for little or no cost. It will be very important to have task scheduling for sharing resources in cloud environment. To maintain effective resource usage, cloud technology equally divides workload among shareable resources when it receives task requests. Machine learning and metaheuristic algorithms afford dynamic component for equitable task distribution in cloud paradigm. The current state-of-art unsupervised models-based load balancing arbitrarily selects centroid locations and struggles to achieve incorrect task requests. Using an optimization technique that takes inspiration from behavioral science, study aims to build well-balanced clustering model-based task scheduling system. In order to efficiently schedule tasks among virtual servers in cloud environment, this proposed work styles aids of perspicacious fuzzy and Grass Hopper algorithms. The results showed that PFC-GOD upsurges cloud resources usage while lowering make-span, execution time, and high balance load scheduling.*

Keywords: *Machine Learning, Task Scheduling, Clustering, Grasshopper Optimization, Cloud Computing, Resources, Load, Perspicacious Fuzzy C-Means.*

1. INTRODUCTION

Cloud Computing, known as distributed computational model brands use of virtualized computer resources dispersed over copious common pools [1]. Accessibility of requesting self-management, resource sharing, broad area networks furthermore load balancing was major areas of focus of cloud computing paradigm [2]. The most challenging job in cloud computing exists resource allocation subsequently thither mismatch amid number of service requests and number of resources that are available. Purpose of dividing available resources among incoming requests, an effective mechanism is therefore required. New strategies for resource provisioning with allocation laid out in an exertion to enhance general performance of cloud amenities for cloud users. These include Drip Irrigation centered Resource Allocation technique, Priority-based Queue (PQ) scheduling strategies, and Roulette Wheel Selection (RWS) technique (DRA). The back end has huge network of data centers by means of wide variety of different applications, system programs, and data storage technologies [3]. Offering different services to internet consumers is what cloud computing stands for. It is possible to divide back end and front end of cloud computing architecture [4]. The program's front-end is represented by entities like web browsers, companies, and different cloud users. It demanded that as result of this process, Cloud Service Providers (CSPs) have nearly ceaseless computational storage capacity. According to requirements of client, cloud service providers competently consign and manage resources [5]. Foremost responsibility about cloud resource governance consists of allocating besides carrying out tasks provided by client users. Two chief cloud procedures are allocation of resources and their configuration. The provisioning task means of finding right resources for certain activity that is purely dependent on degree of service that cloud client's demand. According to resources chosen for resource provision, planning, projecting besides implementation about client task entreaties within cloud were associated to resource scheduling process. Load balancing targets to deliver cloud users with extraordinary level of satisfaction by accelerating task completion and making best expenditure of obtainable resources. The ability of virtual computers to be clustered in accordance with their capacities enables clustering-based load balancing to efficaciously address heterogeneous environment, meet resource demand, and minimize overhead of screening process.

The preeminent goal of this work endures to deliver quality-related services to cloud users while resonant out optimized load balancing in cloud environment by prudently managing dissimilar kinds of virtual machines. With aid of clustering, both incoming requests and virtual machine resources manageable categorized in accordance with availability of resources. This research effort contributes clustering model based on metaheuristics, even in situations when stimulating to assign incoming task request precisely. The major aspiration of this exertion affords quality-based services to cloud consumers while performing optimized load balancing in cloud milieu by carefully managing innumerable types of virtual machines. By gathering, mutually arriving requests as well as obtainable virtual machine assets grouped according to level of resource availability. Even in circumstance when it is difficult to assign incoming task request precisely, this study work contributes metaheuristic-based clustering model. By describing them with degrees of membership, non-membership, and indeterminacy, *perspicacious fuzzy clustering* selects related task requests and resources that are available. By using grasshopper optimization to choose cluster centroids rather than random selection, load-balancing process augmented to cloud environment. The *perspicacious fuzzy clustering* chooses associated task requests in addition to obtainable resources through depicting them about degrees of association, non-membership, in addition to indefiniteness. Load-balancing procedure in cloud environment enhanced by employing grasshopper optimization to choose cluster center hubs rather than random selection. The proposed load balancing methodology for cloud environment thoroughly elucidated in Section3, followed by discussion of outcomes also comparison of proposed exertion towards prevailing consignment balancing techniques in Section4. The paper's results depicted in Section5, conclusions summarized in Section6, furthermore deals with associated work.

2. RELATED WORK

Rantonen et al. [6] proposed load-balancing method for multiprocessor system built based on demand. The typical periodic load balancing algorithm was considered to be insufficient to keep all the processors active as well as to maintain stable load upon that processor. The measurements revealed until CPU reaches idle phase, periodical task scheduling incapable to identify any demand drop. Pradeep and Jacob provided comparison analysis of efficacy of various task-scheduling techniques by means of various service-based scheduling systems and their advantages [7]. Kiruthiga et al. [8] proposed to identify patterns of resource requests that are similar, and those tasks clustered as well as cloud-based VMs according to features by means of brand-new pattern known as intuitionistic fuzzy C-means clustering. By indicating grasshopper optimization behaviors to choose centroids in research rather than choosing preliminary random, clustering process itself further enriched. To improve work effectiveness and output, Raju et al. [9] developed optimal Time approach. Through use of map-reduce task scheduling strategy, workload duration is decreased. Ge et al. [10] creation of job scheduling system on cloud makes use of genetic algorithm. They used every job in work queue to accomplish balanced scheduling between virtual machines, resource assignment conceded out while taking decrease in makespan into account. Naveen Durai et al. [11] suggested HIWIGOA-LB technique exploits IWOA and GOA algorithms to ensure that expected grade of examination and utilization further well-adjusted method, hence achieving potential task scheduling besides load balancing. It combines GOA as well as IWOA utilizing stochastic process and clustering approach to enhance capacity to explore during optimizing and to increase exploitation causes majority of escalation techniques to become trapped in local solutions, outcomes demonstrated that suggested HIWIGO-LB architecture featuring varying sets of fog nodes enlarged resource utilization efficiency. Jang et al. [12] created genetic strategy for task scheduling with QoS benefits besides financial welfares for cloud providers. Qiang Guo et al. [13] suggested ACO to plan job cutting-edge cloud environment. Computer used variables like pheromone collecting and fitness function changes to determine best scheduling method. Their efforts meant to boost productivity, cut costs, and keep cloud workload balanced. Zuo et al. [14] predicted scheduling method by means of ACO paradigm. Utilizing financial constraints, reviewed feedback from previous clients regarding caliber of service. This typical prevents local optima problem of ACO and ensuing negative feedback. [15] Srinivasa Rao Gampa et al. [15] proposed fuzzy GOA-based approaches created for such best assignment to afford peak demand of distribution system and EV charging load at the same time. To optimize size and distribution as well as to enhance reliability with reduced actual loss of power in distribution system, fuzzy multiobjective based GOA shaped in former stage. Hongbo Liu et al. [16] created scheduling problem centered on particle swarm intelligence compared performance of PSO with that of genetic algorithm and simulated annealing Srinivasa et al. [17] suggested evolutionary algorithm-related resource strategy by means of genetic algorithm and control factors like mutation and crossover rate. They showed that their model outdoes batch queuing heuristic by wide margin. Juan et al. [18] developed swarm intelligence-based job scheduling system using cost vector technique to get over cloud network problems. It built model based on input jobs and necessary QoS restrictions and evaluates scheduling arrangements through cost. Despite being effective, it makes things further multifaceted. In order to reduce average running time using few resources, Krishnasamy [19] presented hybrid PSO-based work scheduling system. Alkayal et al. study [20] included application of PSO to ranking-based multi-objective task assignment. Requests are divided among virtual machines based on rank. It performed really well, resulting in less waiting time being experienced. The Rao et al Teaching-Learning.'s Optimization approach [21] is indeed two-phase process that interrelates with spectacle of teaching-learning environment. Dipesh et al. [22] clustering-based resource scheduling method was developed to provide effective service delivery in cloud environments of cloud computing. In order to fairly distribute user requests, these two different load balancing technique clusters scattered cloud computing services during first phase and assigns clients to clusters in second phase. Amer et al. [23] developed dominant series grouping based work scheduling through prejudiced slightest connection towards load balancing. Jobs of consumers organized through dominant series grouping, apiece job organized by means of amended varied premature finish, in edict to achieve superior outcomes. The Hungarian technique is used in Malinen et al. [24] balanced clustering k-means algorithm to optimize average square error for designated number of clusters whilst preserving balanced upkeep of assets. Geetha et al. employed fuzzy C-means clustering, hires simplified scanning technique, to create clustered based load balancing [25]. Overhead of scanning list of available virtual machines congregated as per various competencies, as well as concern's resource requirements are substantially satisfied.

3. PROBLEM STATEMENT

Most linear approaches solely consider homogenous virtual servers but ignore resource demands, which results to unnecessary latency when users examine all-inclusive tilt of virtual machines aimed at every job request besides an uneven workload. Virtual servers clustered approach makes good use of massed-load balancing principle effectively manages and satisfy resource utilization that effectively eliminate complexity.

4. RESOURCE SCHEDULING WITH GRASSHOPPER BEHAVIOR CENTERED PERSPICACIOUS FUZZY CLUSTERING AND OPTIMIZED LOAD BALANCING

In proposed work, multiple service requests submitted to cloud servers are organized using sophisticated unsupervised learning model in accordance with various resource requirements assessed. Numerous task requests are sent out as cloudlets that use maximum resources in cloud. It must be possible to equitably distribute cloud resources among cloudlets quickly. The effective scheduling of work in cloud environment is made possible by numerous optimized methods. This suggested strategy creates clustering model utilizing behavioral inspiration to achieve optimum task scheduling and efficient usage of cloud resources. In mandate to progress service quality in cloud computing, this research effort provided resilient perspicacious fuzzy C-means with grasshopper optimization that combines benefits of perspicacious fuzzy and Grass Hopper algorithms.

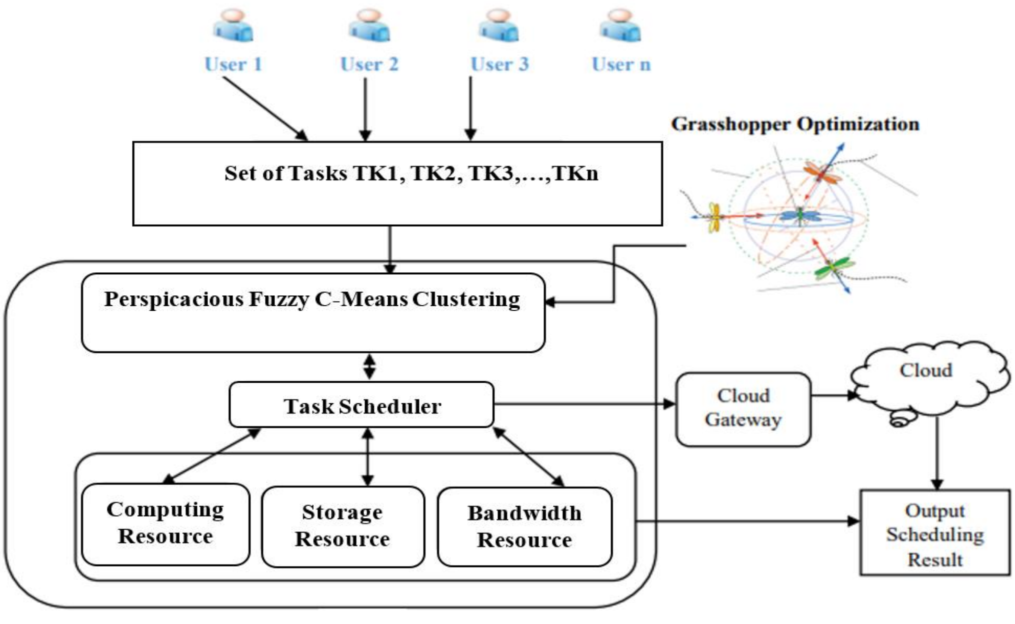


Figure 1: Overall Framework of Grasshopper Behavior Enabled Perspicacious Fuzzy C-Means Clustering in Cloud Exemplar

Figure 1 shows how incoming task scheduling requests are categorized based on amount of storage needed, whether resources must be used, and how much bandwidth is required to finish each task. In cloud environment, accessible virtual machines grouped according to their configuration and all resources made available to them. The membership, non-membership, and amount of hesitation of clustering parameters expressed and perspicacious fuzzy C-Means employed in cloud environs. In order to cluster similar virtual machines and jobs in cloud environment, certain of them must be identified as centroids. To do this, best center node selected via grasshopper optimization. The load distribution across virtual machines balanced by means of this strategy.

Perspicacious Fuzzy Clustering Prelude: A generalization of fuzzy theory is Atanassov's perspicacious fuzzy theory [26]. In fuzzy theory, degree of membership μ_F(z), whose value falls within range [0-1], used to define set F. The membership μ_F(z) and non-membership w_F(z) of set F is represented by perspicacious fuzzy as two independent degrees in representation of set F. According to Equation 1's limitation, μ_F(z) and w_F(z) value ranges amid

$\mu_F(z) + W_F(z) <= 1$……………………..………. (1)

The incorporation of grade of resistance in perspicacious fuzzy greatly reduces issue of ambiguity and imprecision in selecting optimum assets aimed at inward job requirements. For every request arriving, criteria weighed for membership μ, non-membership W, hesitation σ [27]. We can calculate value of using w and σ by means of μ. As stated below, Equation 2 establishes non-membership of perspicacious fuzzy.

$\sigma_F(z) = \frac{1-\sigma_F(z)}{1+\gamma\sigma_F(z)}, \gamma > 0$……………………….. (2)

Equation 3 mentions perspicacious fuzzy hesitation degree as follows:

$\vartheta_F(z) = 1 - \mu_F(z) - = \frac{1-\sigma_F(z)}{1+\gamma\sigma_F(z)} | z \in Z$………..(3)

Cloud environs ideal virtual machine assignment depends upon managing work scheduling uncertainty. Equation 4 below illustrates value of PFC membership calculation.

$\sigma_h^*(z) = \sigma_F(z) + \vartheta_F(z)$ .... (4)

To establish clustering procedure's objective goal, membership and reluctance degrees both taken into consideration while selecting best center node. Below logical fuzzy c-means target function, given by Equation 5 as follows:

$fobj(fu, cs_{1,\dots} cs_n) = \sum_{t-1}^{m} \sum_{k=1}^{n} fobj_n \binom{n}{k} x^k a^{n-k} \binom{1}{n} fu^k a^{n-k} (z_q | cs_q)$.....(5)

Perspicacious Fuzzy C-Means Clustering Technique:

Input: -Requesting Job RJ = {rj1,rj2,rj3….rjn}
Output-Clustering related jobs
Begin

1. Consign 'do' as number of clusters
2. Initialize f>1 //perspicacious fuzziness-degree
3. Initialize np>0 //Perspicacious fuzzy-Negation-parameter
4. Initialize Perspicacious fuzzy Matrix

$$uf^{t(1)} = \{\mu_{pq}{}^{(1)}\}_{doXN} \ \forall p \in \{1,2, \dots, do\} \& \ \forall q \in \{1,2, \dots, N\}$$

5. Assign M ← 1
6. Modify swarm centers

$$w_p^{NFSS(M)} = \langle \mu_w(z_q^{(m)}), v_w(z_q^{(m)}), \pi_w(z_q^{(m)}) \rangle$$

7. Compute $\| z_q^{NFSS} - w_p^{NFSS} \|$
8. Apprise Perspicacious fuzzy partition matrix $up^{(m+1)} = \{\mu_{pq}{}^{(m+1)}, w_{pq}{}^{(m+1)}, \pi_{pq}{}^{(m+1)}\}_{doXN}$
9. If $\| uf^{t(m+1)}\} - uf^{t(m)} \| < \varepsilon$ then $w = \{w_j^{NFSS}\}_{doXN} uf^t = \{\mu_{pq}, w_{pq}, \pi_{pq}\}_{doXN}$
10. Else compute m ← m +1 goto step 12
11. Goto step 6
12. End {Process}

Algorithm 1. Perspicacious Fuzzy C-Means Clustering

Prior to calculating work count, negation parameter > 0 and perspicacious fuzziness e>1 is confirmed in accordance with Algorithm 1. The fuzzy matrix primed based on values once IFS (L) cluster centers updated. Several jobs organized each time fuzzy partition matrices (U (l+1)) are changed.

Knowledge Of Grasshopper Optimization Algorithm: Due of damage they inflict to agricultural crops, grasshoppers are regarded as nuisance pest. These insects have ability to live solitary lives, yet stereotypically congregate in huge swarms. For farmers, swarms that disproportionately big are nightmare. Both as adults and nymphs, they demonstrate distinct swarm behavior traits. When large number of nymphs, they move in cylinder-like motion.

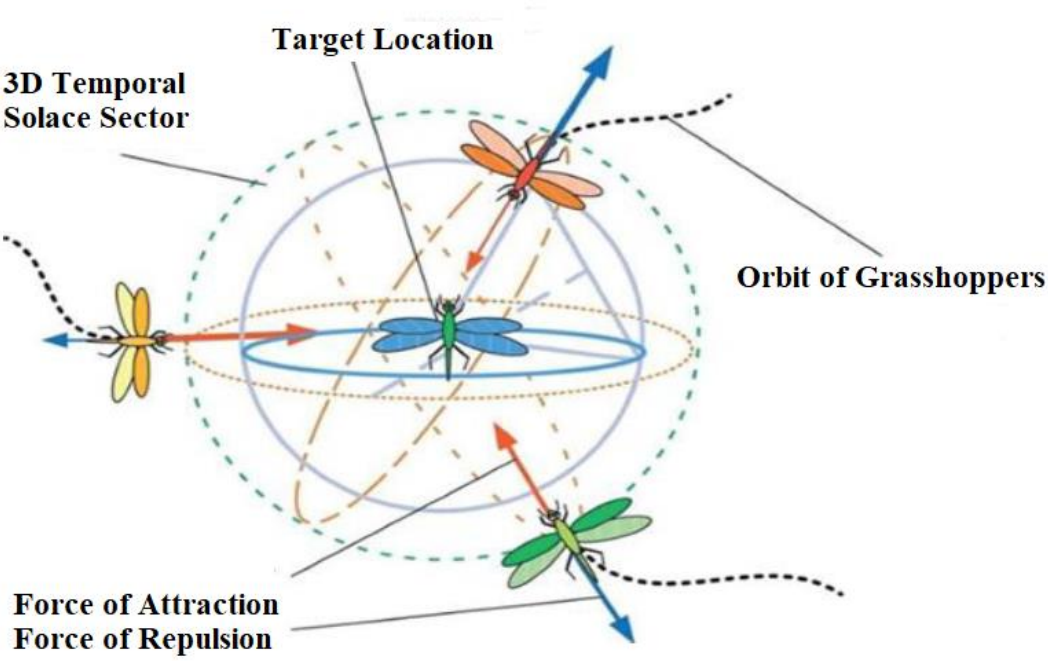


Figure 2:ViewPoint of Grasshopper Behavior Optimization

The majority of flora along their route consumed while they are moving. They form an airborne frame swarm and fly farther [24]. Distinctive characteristic of grasshopper at this stage is swarm, flies very gradually. However, as they age, their motion becomes abrupt. As they search for food, they will gather in swarm. Behavior of Figure 2 modified in this study to locate likely center hubs to edge cloud resource clusters for scheduling optimization. Following procedure illustrates mathematical representation of artificial grasshopper's behavior when searching for food.

Perspicacious Performance: According to fuzzy C's specifications, early centroids in this work are clustered using grasshopper optimization. The grasshopper model customs its food-finding method and determined predicted value from each virtual machine to choose best center hub hearts used for scheduling pertinent tasks. Virtual machine with highest fitness value designated by grasshopper as best competent individual for post.

Grasshopper Optimization Technique

1. Set the starting values of swarm $sm_q = (q = 1,2,3, \dots n)$
2. Determine $ct_{high,}$ $ct_{low,}$opr
3. Evaluate agent Perfect_fit value for each.
4. Perfect fit = Greatest(hunt_agent)
5. While (s<high_itr) Current CT(1) = C high – s $\frac{ct\ high - ct\ low}{opr}$
6. for each hunt_agents : Place simulated grasshoppers' distances together to normalize them.
7. The following is a position update for Search Agents:

TSB_p^a

$$\sum_{p=q=1}^{a}\left(c_n\, UpperB_n \frac{n\pi x}{M} - LowerB_n \sin\frac{n\pi x}{M}\right) tg.\left(tsb_q^a - tsb_p^a\right).\left(\frac{tsb_q^a - tsb_p^a}{etu_{pq}}\right) + Bestsol_a$$

The optimal option so far identified as "$Bestsol_a$" with "$c_n\ UpperB_n \frac{n\pi x}{M}$ " standing for the dimension's upper bound, "$LowerB_n \sin\frac{n\pi x}{M}$" for the dimension's lower bound.

8. Restore contemporary local hunt spinal to its preliminary locus if it leaves barrier zone.
9. End for
10. If an efficient search_agent found, modify perfect fit as s = s+1
11. End {While}

Algorithm 2. Grasshopper Optimization Technique

Algorithm2 primes swarm occupants, cmax and cmin search bounds, and number of recapitulations required. Following each amendment in position of exploration agent based on lower and upper parameters, best agent taken into consideration.

Grasshopper Optimization Driven Cloud Resource Management Through Perspicacious Fuzzy Clustering:

Input:

Jobs Unreserved {JS}, VirtualMachines Unallocated {VMS}

Output: Jobs JS Obligation ← VMS,Time, Resource Utilization,

Procedure:

Start

1. ∀ jobs (p=1…n)
2. Emphasis-value = preference (job_p) * significance (job_p) * preference (job_p) * price(job_p)
3. End ∀
4. Use intuitive fuzzy cluster analysis to put individuals in groups based on how related they are.
5. Use Grasshopper Optimization to choose cluster coordinates.
6. ∀ VMS {p…n}
7. DecideVMs(traits) traits such as bandwidth, RAM capacity, Memory
8. End ∀
9. Smear PFC-GOD to VMS and syndicate them to constellations.
10. VM knots and job knots divided into base, intermediate and high preference type.
11. ∀ jobs (p=1…n)
12. Contingent on kind of ensemble, allocate consideration VM.
13. End ∀
14. End

Algorithm 3. Grasshopper Optimization Supported Perspicacious Fuzzy Clustering

The duration, deadline, and cost of each unallocated assignment prioritized each activity, demonstrated in method 3. The grasshopper optimization selects best-fitting virtual machine, groups them into low, medium, and high priority clusters

5. RESULTS AND DISCUSSIONS

This section discusses suggested Perspicacious fuzzy Clustering algorithm (PFC-GOD) as well as its performance evaluation in order to allocate resource to arriving workloads in cloud scenario. CloudSim serves as cloud simulator in Java to replicate PFC-GOD. Five datacenters being employed in this study, with multiple hosts placed beneath every one of them for total of ten hosts. Between 350 and 1500, tasks/cloudlets salvaged. The proposed model's performance examined using metrics of makespan, resources use, execution time, power consumption, as well as imbalance degree. Additional scheduler approaches that applied to compare performance comprised Fuzzy C-means, K-means gathering, standard Perspicacious fuzzy C Means (PFC-GOD).

Comparability Of Performance by Means of Makespan: Figure3 compares to three currently used clustering models—k-means, FCM, and PFC—suggested model, PFC-GOD, has significantly shorter makespan. K-means approach just utilizes preconfigured cluster centers that randomly picked and commences by consorting related job requests using Euclidean distance alone. Fuzzy C implies only consider job request's possessions when evaluating comparability, but PFC completely ignores job request's non-belongingness. Even though initially cluster chosen at randomness, subsequent centroid decision simply affected by proximity, PFC-GOD takes each centroid's belongingness and non-belongingness into consideration. Therefore, only with aid of grasshopper optimization, centroids given superfluous weight to group best chore request with pertinent constellation.

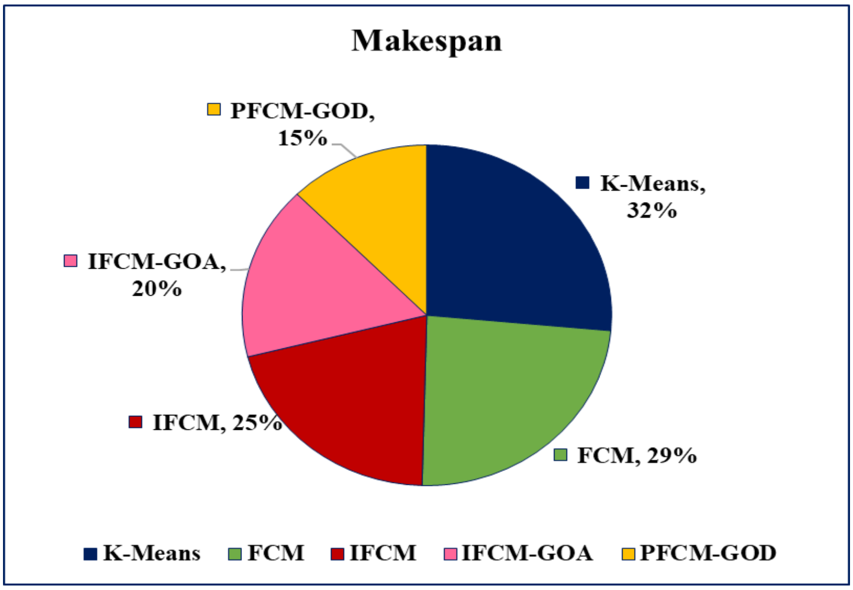


FIGURE 3. Analysis of Five Dissimilar Clustering Techniques in Cloud Environs Centered on Makespan for Resource Management.

Comparability Of Performance by Means of Imbalance Degree: The disparity degree in cloud exemplar acts as barometer for workload differential across virtual machines.

$$\text{Imb\_deg} = \frac{Exec_{[max]} - Exec_{[min]}}{Exec_{[average]}} \ldots\ldots\ldots\ldots\ldots\ldots\ldots (6)$$

Those terms $Exec_{[max]}$, $Exec_{[min]}$ and $Exec_{[average]}$ refer to maximum, minimum, and average execution times, respectively, for virtual machines. The five different clustering algorithms each handle different degree of imbalance between virtual machines in cloud environment, as shown in Figure 4. Owing to its comprehension of how to determine degree of hesitation, suggested PFC-GOD greatly poises load, and centers hubs selected prudently by grasshopper's searching activity. Leftover load counterbalancing clustering exemplars just considered how similar task demands were current models do not place much focus on balancing between virtual machines.

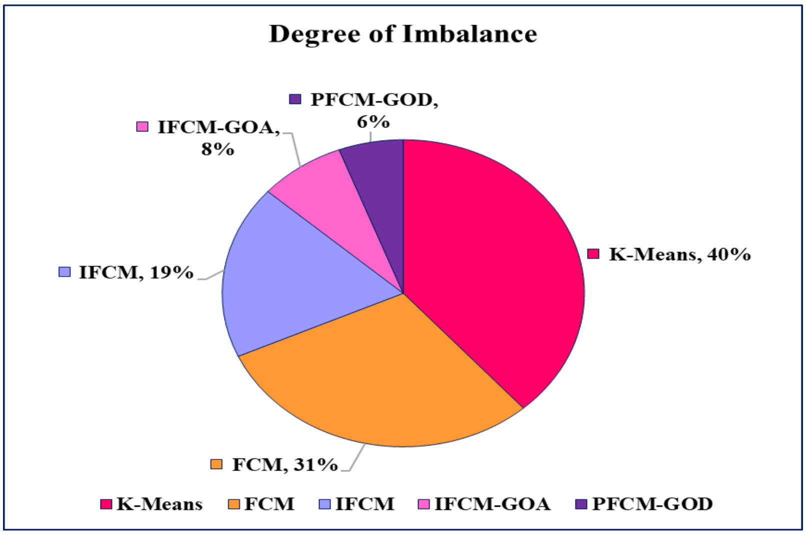


FIGURE 4. Analysis of Five Various Clustering techniques in Cloud Environs Centered on Degree of Imbalance through Resource Management

Comparability Of Performance by Means of Energy Consumption: Figure 5 indicates how much energy of five dissimilar gathering techniques consumed. Most energy utilized when k-means grounded work scheduling implemented cutting-edge cloud location. This is due to fact that all three models randomly choose their center hubs currently in use—k-means, PFC, and PFC-GOD. The new cluster centroids and those throughout each phase of cluster are selected using conceptual approach, PFC-GOD, using metaheuristic model grasshopper optimization. Its food-searching nature takes into account best match task scheduling between incoming task demands and existing virtualization software. Extremely little energy used in cloud environment where resources used in balanced way to fulfil needs of both customer and cloud service provider.

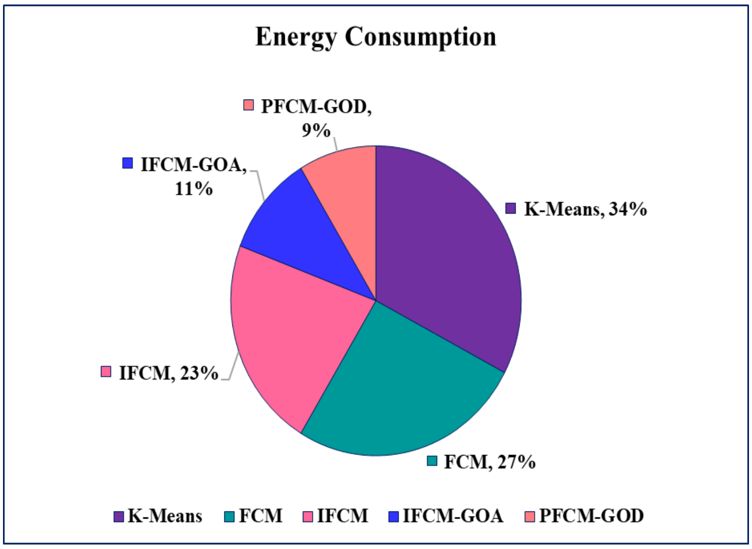


FIGURE 5. Analysis of Five Various Clustering techniques in Cloud Environs Centered on Energy Consumption through Resource Management.

Comparability Of Performance by Means of Resource Utilization: Figure 6 uses four various clustering models to show how resource usage in cloud environment plays out. The present models do not sufficiently take into account selection of suitable virtual machines to complete specified task in an adversarial environment, as well as necessities of incoming job requests not precisely known. The jobs categorized using perspicacious fuzzy clustering into groups like large, intermediate as well as low. Virtual machines consigned towards these jobs in accordance with their requirements, and resources consigned to virtual servers depending on obtainability of resources, therefore when compared to other clustering models, this method succeeds higher rate of resource utilization.

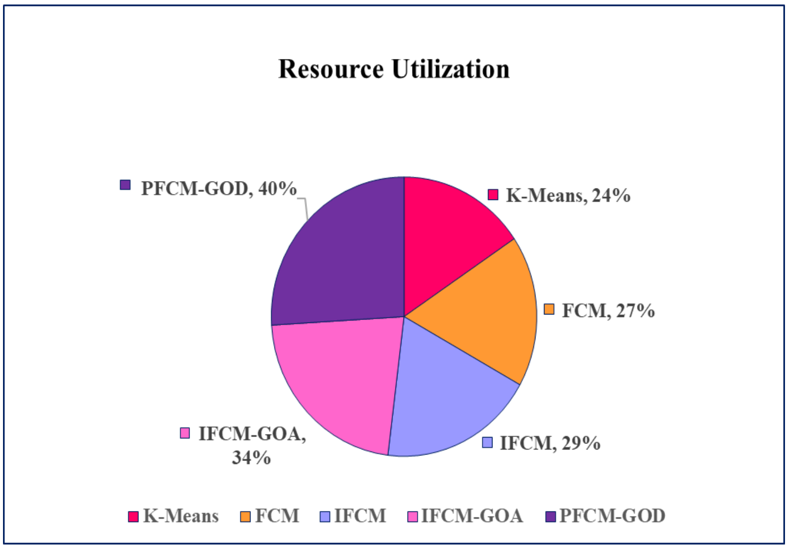


FIGURE 6. Analysis of Five Various Clustering techniques in Cloud Environs Centered on Resource Utilization through Resource Management.

Comparability Of Performance by Means of Execution Time: Figure7 shows four distinct clustering models' execution time performance. When compared to previous clustering-based task scheduling rules castoff in cloud environment, suggested PFC-GOD executes with much shorter time. Relying on throughput, execution time, reaction time, as well as turnaround time for each task, exemplification created. With help of an efficient cluster centroid selection, it is possible to represent each job depending on degree of membership and non-membership, which speeds up clustering and reduces amount of center hubs need to reassign, resulting in shorter execution time.

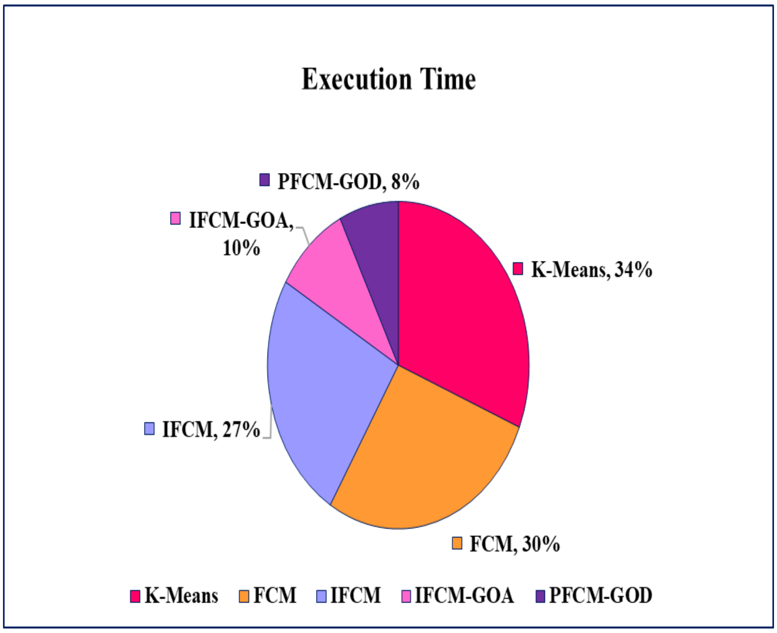


FIGURE 7. Analysis of Five Various clustering techniques in Cloud Environs Centered on Execution Time through Resource Management

6. CONCLUSION

The furthermost perplexing issue will indeed be efficiently scheduling cloud resources because it is thought provoking to determine how many new cloud services and tasks public cloud would entail. The foremost objective of this research is to find analogous exemplars in resource requests. These tasks, therefore pooled, alongside accessible cloud-based virtual servers, based on these physiognomies using brand-new clustering model called Perspicacious Fuzzy C-Means clustering. Additionally, clustering process in this research enhanced by presumptuous optimization of grasshopper behavior in direction of identifying cluster hearts as an alternative of opting outset center hubs haphazardly.

REFERENCES

[1]. S. Singh and I. Chana, "Consistency verification and quality assurance(CVQA) traceability framework for SaaS", 2013 3rd IEEE, International Advance Computing Conference (IACC), Ghaziabad, pp.1-6, 2013.
[2]. N. Kim, J. Cho and E. Seo, "Energy-credit scheduler: An energy aware virtual machine scheduler for cloud systems", Future Generation Computer Systems, vol. 32, pp. 128-137, 2014.
[3]. F. Chen, J. Grundy, J.-G. Schneider, Y. Yang and Q. He, "Automated analysis of performance and energy consumption for cloud applications", In Proceedings of 5th ACM/SPEC international conference on Performance engineering, ACM, pp. 39-50, 2014. https://doi.org/10.1145/2568088.2568093.
[4]. Li X. and Zheng M., "An Energy-Saving Load Balancing Method in Cloud Data Centers", In: Frontier and Future Development of Information Technology in Medicine and Education. Lecture Notes in Electrical Engineering, vol. 269. Springer, Dordrecht. https://doi.org/10.1007/978-94-007-7618-0_35.
[5]. Tilak, S., and Patil, D., "A survey of various scheduling algorithms in cloud environment", International Journal of Engineering Inventions, vol. 1, no. 2, 36-39, 2012.
[6]. M. Rantonen, Tapio Frantti, and Kauko Leivisk, "Fuzzy expert system for load balancing in symmetric multiprocessor systems ", Expert Systems with Applications Journal, Vol. 37, Issue 12, pp. 8711-8720, December 2010.
[7]. K. Pradeep and T. P. Jacob, "Comparative analysis of scheduling and load balancing algorithms in cloud environment", In: Proc. Of International Conf. on Control, Instrumentation, Communication and Computational Technologies, pp. 526-531, 2016.
[8]. G. Kiruthiga and S. Mary Vennila, "Robust Resource Scheduling With Optimized Load Balancing Using Grasshopper Behavior Empowered Intuitionistic Fuzzy Clustering in Cloud Paradigm", International Journal of Computer Networks and Applications (IJCNA), Volume 7, Issue 5, September – October (2020), ISSN: 2395-0455, Pgs.137-145, DOI: 10.22247/ijcna/2020/203851.
[9]. R. Raju, J. Amudhavel, M. Pavithra, S. Anuja and B. Abinaya, "A heuristic fault tolerant MapReduce framework for minimizing makespan in Hybrid Cloud Environment," International Conference on Green Computing Communication and Electrical Engineering (ICGCCEE), Coimbatore, pp. 1-4, 2014. doi: 10.1109/ICGCCEE.2014.6922462.

[10]. Ge, Y., & Wei, G., "GA-based task scheduler for cloud computing systems", International Conference on Web Information Systems and Mining, WISM 2010), Sanya, vol. 2, pp. 181-186, IEEE, 2010. doi: 10.1109/WISM.2010.87.
[11]. K. Naveen Durai, R. Subha and Anandakumar Haldorai , "Hybrid Invasive Weed Improved Grasshopper Optimization Algorithm for Cloud Load Balancing", Intelligent Automation & Soft Computing, DOI:10.32604/iasc.2022.026020,January 2020.
[12]. Jang, S. H., Kim, T. Y., Kim, J. K., and Lee, J. S. "The study of genetic algorithm-based task scheduling for cloud computing", International Journal of Control and Automation, vol. 5, no. 4, pp.157-162, 2012.
[13]. Guo Q, "Task scheduling based on ant colony optimization in cloud environment", In AIP Conference Proceedings, vol. 1834, no. 1, p.040039, 2017.
[14]. Zuo, L., Shu, L., Dong, S., Zhu, C., and Hara, T.," A multi-objective optimization scheduling method based on ant colony algorithm in cloud computing", pp. 2687-2699, 2015. doi:10.1109/ACCESS.2015.2508940.
[15]. Srinivasa Rao Gampaa , Kiran Jasthia , Preetham Golib , D. Dasc , R.C. Bansald, "Grasshopper optimization algorithm based two stage fuzzy multiobjective approach for optimum sizing and placement of distributed generations, shunt capacitors and electric vehicle charging stations", Journal of Energy Storage, https://doi.org/10.1016/j.est.2019.101117, November 2019.
[16]. H. Liu, A. Abraham, A.E. Hassanien, "Scheduling Jobs on computational grids using fuzzy particle swarm optimization algorithm", Future Generation Computer Systems, 2009.
[17]. Ch.Srinivasa Rao, B. Raveendra Babu, "DE Based Job Scheduling in Grid Environments", Journal of Computer Networks, vol. 1, no. 2, pp.28-31, 2013.
[18]. Juan, W., Fei, L., and Aidong, C., "An Improved PSO based Task Scheduling Algorithm for Cloud Storage System", Advances in Information Sciences and Service Sciences, vol. 4, no. 18, pp. 465-471, 2012.
[19]. Krishnasamy K.," Task Scheduling Algorithm Based on Hybrid Particle Swarm Optimization In Cloud Computing Environment", Journal of Theoretical and Applied Information Technology, vol. 55, no.1 , pp. 33-38, 2013.
[20]. Alkayal, E. S., Jennings, N. R., and Abulkhair, M. F.,"Efficient task scheduling multi-objective particle swarm optimization in cloud computing", In 2016 IEEE 41st Conference on Local Computer Networks Workshops (LCN Workshops). pp. 17-24, IEEE, 2016.doi:10.1109/LCN.2016.024
[21]. Rao, R. V., Savsani, V. J., Vakharia, D. P, "Teaching–learning-based optimization: novel method for constrained mechanical design optimization problems", Computer-Aided Design, vol. 43, no. 3, pp. 303-315, 2011.
[22]. Dipesh Pradhan, Feroz Zahid, "Data Center Clustering for Geographically Distributed Cloud Deployments, Primate Life Histories, Sex Roles, and Adaptability", pp. 1030-1040, 2018. doi: 10.1007/978-3-030-15035-8_101.
[23]. Amer Al-Rahayfeh , Saleh Atiewi , Abdullah Abuhussein, MuderAlmiani,"Novel Approach to Task Scheduling and LoadBalancing Using Dominant Sequence Clustering and Mean Shift Clustering Algorithms", Future Internet, vol. 11,no. 109 , pp 1-15, 2019.
[24]. Malinen M.I., FräntiP. , "Balanced K-Means for Clustering. In: FräntiP., Brown G., Loog M., Escolano F., Pelillo M. (eds) Structural,Syntactic, and Statistical Pattern Recognition", Lecture Notes in Computer Science, vol 8621. Springer, Berlin, Heidelberg, 2014.
[25]. Geetha Megharaj, Dr. Mohan G. Kabadi, Rajani, Deepa M, "FCM-LB:Fuzzy C Means Cluster Based Load Balancing in Cloud", International Journal of Innovative Research in Science, Engineering and Technology, vol. 7, Special Issue 6, 2018.
[26]. Atanassov K, "Intuitionistic fuzzy logics as tools for evaluation of data mining processes", Knowl-Based Syst, vol. 80, pp. 122–130, 2015. doi: 10.1016/j.knosys.2015.01.015.
[27]. Zeshui Xu, and Junjie Wu,"Intuitionistic fuzzy C-means clustering algorithms", Journal of Systems Engineering and Electronics, vol. 21, no. 4, pp.580–590, 2010. doi: 10.3969/j.issn.1004-4132.2010.04.009.
[28]. Shahrzad Saremi, Seyedali Mirjalili, Andrew Lewis, "Grasshopper Optimization Algorithm: Theory and application", Advances in Engineering Software, vol. 105, pp. 30-47, 2017.

