

# Load Balancing in Cloud Environment Exploiting Hybridization of Chicken Swarm and Enhanced Raven Roosting Optimization Algorithm

**Ashok Kumar C**

*PSNA College of Engineering and Technology,  
Kothandaraman Nagar, Dindigul, India  
ashokkumar.c127@gmail.com*

**Vimala R**

*PSNA College of Engineering and Technology,  
Kothandaraman Nagar, Dindigul, India*

**Abstract:** All over the world, cloud computing offers a shared resources pool for several users. The cloud model encompasses the virtual machines and PMs from the users in order to processes the tasks in an equivalent way. In particular circumstances, the demand for users might be maximum that causes the processing units overloading and this circumstance influences the evaluation of the cloud setup. To balance the load of the cloud environment, numerous works have been proposed for the load balancing approach; however, they need to decrease the quantity of task migration. This model develops the balancing of load approach adopting crucial the optimization technique and the multi-objective design. This work develops chicken swarm and enhanced raven roosting optimization, for balancing the load (CS-ERRO) algorithm via the hybridization of the Chicken Swarm Algorithm (CSA), Enhanced Raven Roosting Optimization Algorithm (ERROA). Additionally, this work exploits the multi-objective model on the basis of the probabilities selection, the frequency scaling due to the ability of the machine and the data length of the task. Finally, the performance of the proposed CS-ERRO algorithm is evaluated in different cloud cases, and from the consequences, it is obvious that the proposed algorithm attained reduced load while comparing with the conventional algorithms.

**Keywords:** Load Balancing; Cloud Environment; Virtual Machines; Multi-Objective Model; Hybrid Optimization Algorithm

## Nomenclature

Abbreviations	Descriptions
DA	Dragonfly Algorithm
DWOLB	Dynamic Well-Organized Load Balancing
GSO	Glowworm Swarm Optimization
QoE	Quality of Experience
FF	Firefly
TSP	Task Selection Probability
ABC	Artificial Bee Colony
SLA	Service Level Agreement
GA	Genetic Algorithms
RRO	Raven Roosting Optimization
HG-GSA	Hybrid Genetic-Gravitational Search Algorithm
VSP	VM Selection Probability
IPSO	Improved Particle Swarm Optimization
VMs	Virtual Machines
QoS	Quality of Service
PM	Physical Machine
PSO	Particle Swarm Optimization
CDN	Content Delivery Network

## 1. Introduction

Generally, the cloud environments are performed on the basis of the virtualization, grid computing, and distributed computing. This environment encompasses clear benefits like media storage in worldwide in order to use the platforms. Additionally, at the end-user circumstances, the least demand in hardware is

needed, which is considered as the cloud problem. One of the most important cloud problems is load balancing, accessibility, scalability, superior performance, and cost. In this environment, load balancing was considered as confronts [1].

On cloud computing [19], the load balancing methods are considered as the general framework-based procedure whereas, in resources of the multiple data centers, it distributes the produced workloads. Moreover, aforesaid methods convey the minimum response time benefits. Nevertheless, for resources replication prices are also considered as an extra price [18]. In the hardware resources for load balancing, domain name service load balancers are set up in software and hardware components. While the load balancing in cloud-based methods installs software techniques otherwise protocols deal with the multiple load services. In addition, to implicit, the load balancing in cloud-based methods permits the consumers to employ the geodetically else global services in a distributed manner on account of the geodetically distributed servers [15]. The advantages attained for load balancing in the data center are abundant as managing the maximum unforeseen traffic usually indicated as Cyber Spikes.

The employ of cloud computing [16] [17] is growing in a day by day routine life basis, and by cloud servers loads encountered and it increases considerably. The demand for resources possesses consequently increases to a great extent. For these applications, there is a great count of users; and the attractiveness of cloud technologies represents that there is constantly a great count of consumers. In the cloud, this causes the greatest extent possible of productivity that consecutively initiates a minimization in the response time [3]. Starvation of resources must be minimized, with the intention to sustain an efficient service, as this can cause probable loads. Thus, it is essential on the server to decrease the load as a result that each and every user is given with equivalent presentation. Moreover, DWOLB is considered as a novel method for enhancing performance during a minimization in the employ of resources and for guarantees [2].

In a smaller quantity of computing resources, virtualization presents a capable technique of strengthening numerous online services in a data center. This application permits a solitary server needs to distribute between numerous isolated platforms performance named Virtual Machines. Additionally, on-demand is permitted by virtualization otherwise computing effectiveness a immediately-in-instance provisioning of resource technique in that computing resources like memory, disk space and CPU are done obtainable to technologies merely as required and not assigned statically because of the demand of the peak workload [20]. Using dynamically VMs maintaining, stabilizing the workload, and revolving on merely required servers, operators of the data center have the ability to sustain the required QoS whilst attaining superior utilization of server and the effectiveness of energy [11]. Nevertheless, as system computing turns out to be superior and additional compound, the alteration task several control considerations, like memory share, disk space, and CPU, is attractive exhausted, utilization of time, costly and approximately unfeasible for the operators of data center to perform automatically [13] [14]. Hence, it is extremely attractive for those operations to be mainly autonomic or self-managing; merely necessitating by operators the maximum-level instruction is attained [12].

The main contribution of this work is to propose the enhanced load balancing in cloud computing. Here, a multi-objective method is introduced which deals with the parameters of the frequency scaling and it is incorporated with the machine capability, and VM load, QoS parameters and the cost of task migration, for the choice of the best VM. Here, the hybridization of the Chicken Swarm Algorithm (CSA), Enhanced Raven Roosting Optimization Algorithm (ERROA) is designed in order to attain the enhanced balancing of the load for the cloud environment. The proposed CS-ERRO algorithm exploits the fitness criterion on the basis of the multi-objective method for the VM selection.

## 2. Literature Review

In 2017, Niladri Dey and T. Gunasekhar [1], presented a broad survey on conventional VM selection and migration procedures to recognize the definite technology-tilting abilities of these schemes by means of the bottlenecks and benefits. Additionally, by means of generous of the conventional metrics for balancing the load, it was significant to provide the promote enhancement schemes that can be done probably with comprehensive considerate of the parallel research results. Hereafter, also, this paper provides the work with a rule for enhancements and for additional researches. Consequently, this work besides proposes the explanation of analytical techniques and empirical consequence evaluation of the decisive decision on enhancement factors and usability of the exodus techniques for several reasons.

In 2019, Yunmeng Dong et al [2], utilized that the distribution form of collective cloud design, based on the distribution scheme HEELS on account of the evaluation of the heuristic task clustering approach and GSO approach. Its most important suggestion comprises of two kinds. Initially, the outsized resources task in the present task set was changed out using the analysis of the clustering, and the task offloading application was used to upload the consequence for the center of cloud computing in order to

compute and deployment. Subsequently, in the edge computing center, the optimized GSO technique was used, and the suggestion of SCA was incorporated into step size optimization with the intention of the GSO optimization technique.

In 2018, Utku Balkan et al [3], developed a method for calculating QoE for an extensible cloud-based online video program. The design offers an expertise principle concerning restricted cloud resources and memory, transcoding, transmits computational capacity, in addition to throughput ability and lastly latency capability for the service of cloud to QoE. System effectiveness, and scalability, was optimized by estimate an adequate amount of containers, and VMs to assure the requests by the users still on durations of the peak demand with the least amount of VMs. In cooperation of vertical and horizontal scaling schemes (and VM migration) were modeled to conspiracy reliability and availability of intermediary and edge CDN cache nodes.

In 2018, Nikolaos Leontiou et al [4], developed a hierarchical control model, in a data center that aspires to the objectives of the negotiation antagonistic. The local control level overcomes concurrent issues in the allocation of resources and VM admission control whilst the higher level tackles simultaneously balance the load for the inward requests and VM position into a physical servers cluster.

In 2018, V. Priya et al [5] developed an incorporated balancing of the load and scheduling of resource approach for competent cloud service provisioning. The technique models Fuzzy-based multidimensional scheduling of resources designed to attain scheduling of resource effectiveness in a cloud environment. Rising consumption of VMs during effectual and fair balancing of the load was subsequently attained via enthusiastically choosing a demand from a class exploiting Load Optimization of the Multidimensional Queuing technique. A balancing of load approach was subsequently applied to evade overutilization and underutilization of resources, enhancing latency time.

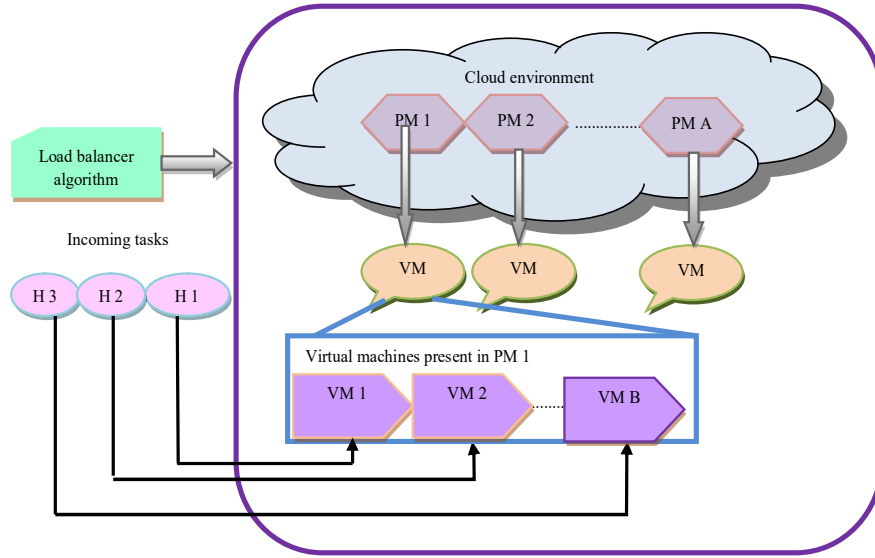
In 2019, Mahya Mohammadi Golchi et al [6], developed a hybrid of FF and IPSO techniques with the intention to achieve the improved average load in order to create and enhancing the significant metrics like effectual resource consumption and the response time of tasks correspondingly. Additionally, this work was developed a few indicators to estimate the developed hybrid algorithm performance.

In 2018, Divya Chaudhary and Bijendra Kumar [7], developed a novel load scheduling approach, called HG-GSA in order to minimize the computation of total cost. The overall computational cost comprises execution cost and transport. It functions in a hybrid crossover method on the basis of the GSA with the purpose of search the best position of the particle in the search space. The optimal location of the particle was exploited to compute the force.

In 2017, M. Vanitha and P. Marikkannu [8], worked on the cloud service provider takes on to bring each and every subscribers' necessities as stated by the SLA. These resources ought to be well-confined and numerous subscribers exploited. For these resources and services, there was a persistent large level of challenge, and it was thus essential for load balancing on a variety of servers; this was performed to evade overcrowding in-network and to decrease utilization for the resources else capital. To balance the load exploits techniques in the VM like GA with the intention of dealing with the load which was presented in the network. A new method for balancing the load technique was developed, concerning an efficient employ of resources that was called as the DWOLB technique.

### 3. System Model of Load Balancing

The cloud network process design is described in this section. Here, it is focused on the balancing of the load problems. Fig. 1 shows the cloud network process design and the criteria impacting the network load. Moreover, the cloud is constructed using various physical machines. To the end-users, each machine runs on multiple virtual machines. Over multiple computing resources, balancing of the load in the cloud is the procedure of assigning workloads. The overloaded and underloaded nodes are detected by load balancer approach and subsequently, loads are balanced between them. Load balancing in the cloud minimizes costs, which is related to the management of the document models and raises the availability of the resources.



**Fig. 1.** Diagrammatic representation of the load balancing

Let cloud network comprises 'C' PMs, set as  $\{C_1, C_2, C_3, \dots, C_c, \dots, C_C\}$ . Here, the physical machines encompass numerous VMs for dealing with each task from the users. Let, the PM comprise 'D' VMs, from the users, for evaluating the tasks. The VMs in attendance with the  $c^{th}$  PM is indicated as  $\{D_1, D_2, \dots, D_d, \dots, D_D\}$ . In the PM, the user requests are assigned to every virtual machine. In the cloud network, virtual machine available contains diverse properties. The capability/practice of the virtual machine based upon the aspects, like count of MIPS, count of processors, migration of task cost, total bandwidth, system memory, and scaling factor of frequency. In the cloud network,  $d^{th}$  VM includes the following factors,

$$D_d = \{V_d, U_d, F_d, K_d, L_d, M_d\} \quad (1)$$

In eq. (1),  $V_d$  states the overall processor included in the data processing in the virtual machine,  $U_d$  states the overall amount of MIPS in virtual machine that permits instructions processing, and  $L_d$  states the memory of the VM and  $K_d$  states the migration cost; and the  $F_d$  states the frequency scaling,  $M_d$  states the bandwidth of the  $d^{th}$  virtual machine in the physical machine.

In the cloud, capability for each VM available is described using the restrictions stated in the eq. (1). For the common appliances, the count of MIPS and processors, in the virtual machine differs among one and ten. The bandwidth and the migration cost of task values for the virtual machine diverge from zero to one. The virtual machine memory encompasses the values from zero to ten.

In the cloud network, the main difficulty is the load balancing between every virtual source accessible. From the user, the balancing of the load technique ensures the inward task and allocates the task to the accessible VM. At the cloud network, the tasks inward for the processing encompass numerous properties such as priority level, communication cost, data length, and execution time. Let  $T_i$  represents the  $i^{th}$  inward task from the user. From the user, the task  $T_i$  is assigned to the VM on the basis of the subsequent qualities,

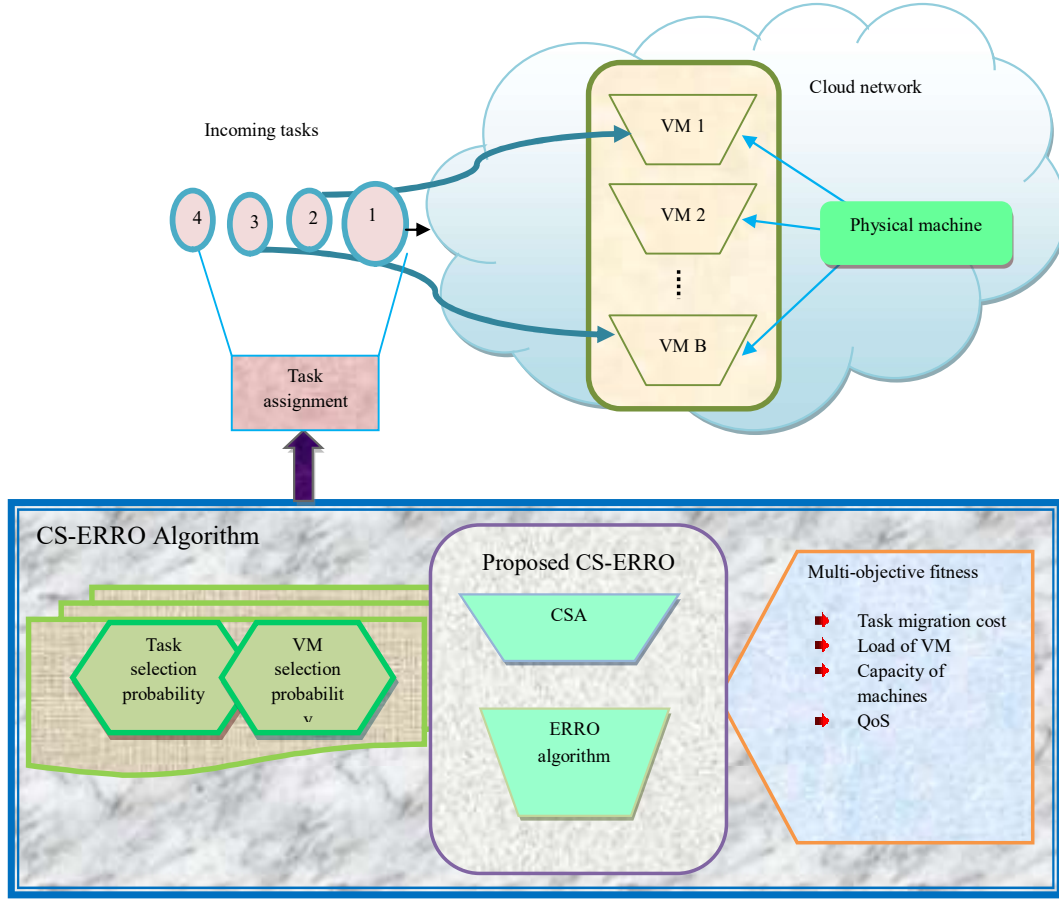
$$T_i = \{W_i, X_i, Y_i, Z_i\} \quad (2)$$

In eq. (2),  $X_i$  states the priority level and  $W_i$  states the total execution time of the  $i^{th}$  task,  $Z_i$  states the data length of the task and  $Y_i$  states the total communication cost. Due to the task properties and VM, the task to the VM is allocated by the balancing of the load; as a result, it evades the delay in the cloud operation.

#### 4. Load Balancing in Cloud Environment using Proposed Methodology

In the cloud environment, the hybridization of the CSA-ERRO algorithm is developed for the effectual balancing of the load. Fig. 2 shows the utilization of the proposed CSA-ERRO algorithm of the cloud network for load balancing. The proposed CSA-ERRO algorithm is the hybridization of the CSA, ERRO approach. Hence, in the cloud environment, CSA-ERRO ensures the position of the load for every virtual machine available and task reallocation if the virtual machine load goes beyond a predefined value of the

threshold. The CSA-ERRO states a TSP and the VSP on the basis of the properties of each task and VM. Additionally, the proposed CSA-ERRO algorithm indicates fitness on the basis of the numerous parameters purposely, a virtual machine load of the capability of the virtual machine, QoS and task migration cost. The proposed CSA-ERRO algorithm allocates the appropriate tasks to the best VM on the basis of the fitness criterion.



**Fig. 2.** Systematic illustration of the developed methodology for the load balancing

The load balancing model exploiting the developed CSA-ERRO algorithm is described in this section. In the developed algorithm, the load of each VM is stabilized with two chosen probabilities in the cloud such as TSP and VSP.

To the VM the obligation of the appropriate tasks is performed based upon the handling of the task that has the capability of all virtual machines. In PM, if the virtual machine has the ability to hold merely two tasks simultaneously, the load balancer will allocate two tasks to the virtual machine and redirects the inward task to the subsequently virtual machine. Therefore, for the competent balancing of the load, the capability of every virtual machine available in the network encompasses is established. Using eq. (3), the capability of the  $d^{\text{th}}$  VM in the  $c^{\text{th}}$  PM is computed.

$$h(D_d) = \frac{1}{3} \left[ \frac{V_d * U_d * L_d}{1000} + F_d + M_d \right] \quad (3)$$

In eq. (3),  $h(D_d)$  states the capability of the  $d^{\text{th}}$  VM. The ability of the VM chiefly based upon diverse properties of the VM. The load of the VM acquires diverse whilst the VM process a task. Eq. (4) indicates the load of the  $d^{\text{th}}$  VM which executes the  $i^{\text{th}}$  incoming task.

$$J(D_d) = \frac{1}{|W_i|} \sum_{i=1}^I \left( W_i * S_i^d * \frac{Z_i}{M_d} \right) \quad (4)$$

In eq. (4),  $S_i^d$  states if the  $i^{\text{th}}$  task is related in  $d^{\text{th}}$  virtual machine or not,  $J(D_d)$  indicates the load of the  $d^{\text{th}}$  virtual machine in the cloud  $v$  represents the lesser bound of the virtual machine and  $u$  indicates

the upper bound. The, u value indicated as 11 and v value indicated as 20. As per eq. (5), the value of  $S_i^d$  is selected.

$$S_i^d = \begin{cases} 1; & \text{For task } i \text{ related to VM } d \\ 0; & \text{else} \end{cases} \quad (5)$$

In the cloud network, the physical machine load available is an essential aspect of the allusion in the load balancer of technique. The system describes the developed balancer of load technique subjected to the physical machine load goes beyond by the user a predefined value of the threshold is chosen. Therefore, in load balancing, the load of the physical machine acts as the main function. In the cloud, the physical machine encompasses a varying number of the virtual machine, and so, the physical machine load based upon every virtual machine load in the individual physical machine. Eq. (6) states the physical machine load concerning the virtual machine load.

$$J(C_c) = \frac{1}{|D|} \sum_{d=1}^{|D|} J(D_d) \quad (6)$$

Eq. (6),  $|D|$  shows the overall count of VMs present in  $c^{\text{th}}$  PM.

#### 4.1 Selection Probabilities

**VSP:** The proposed CSA-ERRO algorithm chooses the virtual machine for the load balancing on the basis in the subsequent circumstances,

- ❖ The virtual machine should encompass a high capability.
- ❖ The virtual machine should encompass the least load.

Hence, the VSP expression in the proposed CSA-ERRO algorithm is expressed as eq. (7).

$$VSP = \frac{1}{2} [h(D_d) + (1 - J(D_d))] \quad (7)$$

In eq. (7),  $h(D_d)$  and  $J(D_d)$  indicates the capability and the load of the  $d^{\text{th}}$  virtual machine, correspondingly.

**TSP:** The proposed balancing of the load techniques chooses the virtual machine task due to the properties, like communication cost, task priority, execution time, and the task data length. By the average of each parameter describing a task, TSP of the proposed CSA-ERRO algorithm is decided. Eq. (8) indicates the TSP of the proposed CSA-ERRO algorithm.

$$TSP(T_i) = \{W_i + X_i + Y_i + Z_i\} * \frac{1}{4} \quad (8)$$

In eq. (8),  $TSP(T_i)$  indicates the selection probability of the task  $T_i$ .

#### 4.2 Proposed Methodology for Multi-Objective Fitness

The proposed CSA-ERRO algorithm exploits the multi-objective fitness models with the parameters, like QoS, the virtual machine capability, a VM load, and cost of the migration for transporting single tasks from the single virtual machine to other. The fitness of the proposed CSA-ERRO algorithm is explained as the maximization function, and it is stated in the eq. (9).

$$\text{Fitness } G = \max \left[ \sum_{i=1}^I (1 - J(D_i^d)) + h(D_d) + (1 - K_i^d) + Q_i^d \right] \quad (9)$$

In eq. (9),  $h(D_i^d)$  indicates the capacity of the  $d^{\text{th}}$  VM,  $J(D_i^d)$  represents the load of the VM, when executing the  $i^{\text{th}}$  task,  $K_i^d$  and  $Q_i^d$  indicates the migration cost for the task  $i$  and the parameter of QoS. In the proposed fitness model, the QoS parameter exploited is based upon resource exploitation and the capability of the VM. Eq. (10) is used to calculate the QoS of the VM.

$$Q_i^d = \left( 1 - \frac{H_i^d}{h(D_d)} \right) \quad (10)$$

In eq. (10),  $H_i^d$  indicates the resource used by the  $d^{\text{th}}$  VM. The resource utilization based upon the diverse parameters manipulates the performance of the task and the VM. In eq. (11) indicates the resource used in the VM.

$$H_i^d = \frac{1}{10} \left[ W_i * S_i^d * \frac{1}{3} \left( \frac{V_d * U_d * L_d}{100} + F_d + \frac{Z_i}{M_d} \right) \right] \quad (11)$$

### 4.3 Proposed Methodology for VM Selection

In this work, the proposed CSA-ERRO algorithm is presented in the cloud for the balancing of the load scheme. CSA-ERRO adapts the RRO with the employ of the CS algorithm. The CS algorithm discovers the best value by the behavior of the chicken swarm. The algorithm needs merely fewer parameters for the optimization and therefore, encompasses a simpler implementation. The load balancing technique is started while the PM load goes beyond a value of the predetermined threshold and it is stated using a value 0.8.

## 5. Proposed Methodology

### 5.1 Chicken Swarm Algorithm

The roosters with enhanced fitness functions encompass precedence for food admittance than the others with poorer fitness functions [9]. For ease, this scenario is experimented using circumstances that the roosters with enhanced fitness functions have the ability to investigate food in an ampler collection of seats than that of the roosters by means of poorer fitness functions. Additionally, it is devised using eq. (12).

$$y_{j,k}^{t+1} = y_{j,k}^t * (1 + rn(0, \omega^2)) \quad (12)$$

$$\omega^2 = \begin{cases} 1, & \text{if } f_j \leq f_i \\ \exp\left(\frac{f_i - f_j}{|f_j| + \eta}\right), & \text{otherwise} \end{cases} \quad i \in [1, N], i \neq j \quad (13)$$

In eq. (12),  $rn(0, \omega^2)$  represents a Gaussian distribution with standard deviation  $\omega^2$  and mean 0.  $\eta$ , that is exploited to evade 'zero-division-error', which represents the least constant in the processor.  $i$ , indicates an index of rooster's, which is arbitrarily chosen from the group of roosters,  $f$  indicates the value of the fitness for the equivalent  $y$ . To search for food, the hens can pursue their group-mate roosters. In addition, they could arbitrarily take the high-quality food which is reserved by other chickens, although by the other chickens they will be reserved. The additional leading hens will possess benefit in a rival for food than the additional obedient ones. These phenomena have the ability to devise as below equation:

$$y_{j,k}^{t+1} = y_{j,k}^t + R_1 * rn(y_{r_1,k}^t - y_{j,k}^t) + R_2 * rn(y_{r_2,k}^t - y_{j,k}^t) \quad (14)$$

$$R_1 = \exp((f_j - f_{r_1}) / (abs(f_j) + \eta)) \quad (15)$$

$$R_2 = \exp((f_{r_2} - f_j)) \quad (16)$$

In eq. (14),  $rn$  indicates a uniform arbitrary number above  $[0, 1]$ .  $r_1 \in [1, \dots, N]$  indicate an index of the rooster, specifically the  $i^{th}$  group-mate of hen's, whereas  $r_2 \in [1, \dots, N]$ , is an index of the chicken that is arbitrarily selected from the swarm  $r_1 \neq r_2$ .

Noticeably,  $f_j > f_{r_1}, f_j > f_{r_2}$ , thus  $R_2 < 1 < R_1$ . Consider  $R_1 = 0$ , subsequently, the  $i^{th}$  hen could forage for food presently goes after beside the remaining chickens. The greater the divergence of fitness values of the two chickens', the lesser  $R_2$  and the greater the breach among the two chickens' locations. Therefore hens could not simply take the food to establish by remaining chickens. The cause of the principle outline of  $R_1$  varies from  $R_2$  is that there subsists rivalry in a group. For ease, the fitness function values of the chickens compared with the rooster fitness function have experimented with the rivalry among chickens in a collection. Presume  $R_2 = 0$ , subsequently,  $i^{th}$  hen could look for food in their possess region. For the exact group, the fitness value of the rooster's is inimitable. Hence the lesser the  $i^{th}$  hen's fitness value, earlier  $R_1$  estimates to one and the lesser breach among the locations of  $i^{th}$  hen along with rooster group-mate. Therefore, additional leading hens could be additional probable than the further acquiescent ones to consume food.

For food, to search for food, the chicks follow its mother and it is devised in eq. (17).

$$y_{j,k}^{t+1} = y_{j,k}^t + F_L * (y_{p,k}^t - y_{j,k}^t) \quad (17)$$

In eq. (17),  $y_{p,k}^t$  shows the position of the  $j^{\text{th}}$  chick's mother ( $p \in [1, N]$ ).  $F_L$  ( $F_L \in [0, 2]$ ) is a parameter, it indicates which the chick might follow its mother to scavenge for food. Let the individual differences,  $F_L$  of every chick, would arbitrarily select among 0 and 2.

## 5.2 Raven Roosting Optimization (RRO)

In recent times, the bio-inspired meta-heuristic algorithm, RRO is proposed [10]. This algorithm concentrates on the raven's social behavior and the information flow among members of the population with the intention of discovering food. Typically, ravens exist together in groups of 200 to ten thousand in autumn and winter (the non-breeding seasons). These collections are named as social sleep or "roost" and formed by considering sources of food on trees. Still, the causes are obscurity for social sleep. Nevertheless, there is a hypothesis for the improvement for these social groups such as a) in the roost, birds that live are less horizontal to assault by other animals. b) The roost can be represented as an in sequence center, a position which provides since the aim of substitute in sequence on food sources among members of the group. In this algorithm, the primary step is arbitrary chosen of a roost in the search space. In the search space, every population of  $N$  members is positioned arbitrarily. For each bird the fitness value is computed for the location, and the bird with the optimal solution is represented as the "leader." Few raven populations leave the roost and pursue the leader. Within a hypersphere, the admirer bird's search comprises the radius about the position formerly establish by the leader at that to scavenge. Members of the remaining population take off toward their individual optimal position and carry on to scavenge. Ravens, which are flying are separated into  $N$  steps. Every step length is selected arbitrarily, and while flying every bird's location is updated using eq. (18).

$$y_{j,t} = y_{j,t-1} + d_{j,t} \quad (18)$$

In eq. (18)  $y_{j,t}$  indicates the new location of the  $j^{\text{th}}$  raven,  $y_{j,t-1}$  represents the current location of  $j^{\text{th}}$  raven and  $d_{j,t}$  is an arbitrary step size.

The bird perceives in each step, the quality of its discontinuing point and its surroundings in the range of radius  $RD_{\text{pct}}$ . Within this hypersphere, it samples  $N_{\text{pct}}$  perceptions arbitrarily. If an enhanced position is established than that of the optimal locations bird, subsequently there is a percentage of a possibility  $\text{prob}_{\text{stop}}$  the raven stops its flying in that moment and forages at the recently establish position or else, it carries on to travel to its objective.

## 5.3 Hybrid CS and Enhanced RRO Algorithm

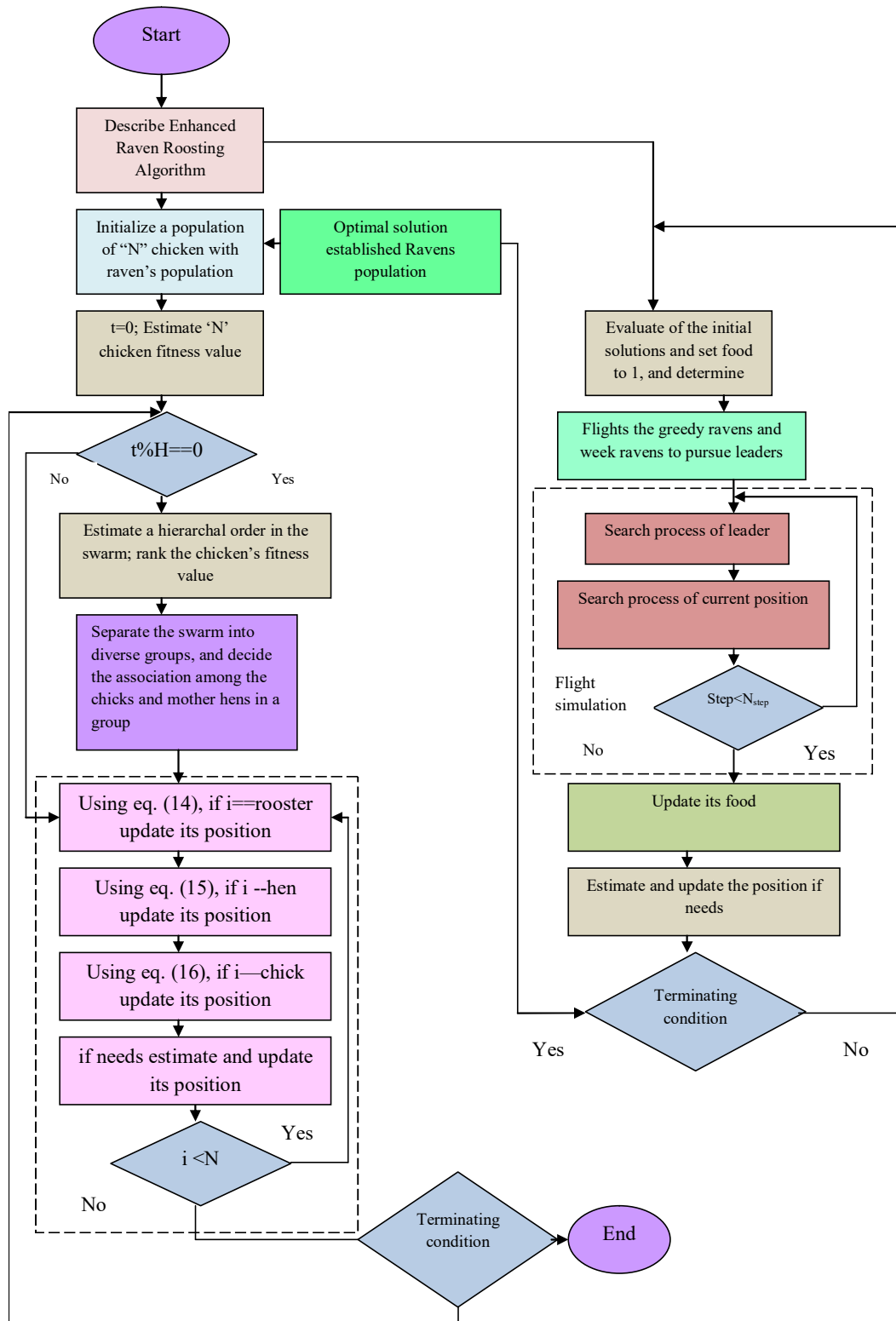
The roost is contemplated the center of information, a position where food sources information is transferred among associates. Here, the major focal point was the construction of the RRO technique. The component for the population of a raven which left the roost to discover food and pursues the leader is categorized in two collections, in relation to greedy ravens and weak ravens. In a day, the ravens which cannot discover food sources for themselves are decided as weak ravens. This group member observes the ravens that resume the roost containing establishes the affluent sources of food and pursue them to discover the food. The subsequent group comprises of gluttonous ravens, which pursue the leader to obtain additional food sources. How to partition this part of the population into two collections and to decide the recognition of the ravens (greedy or weak) all based upon the raven's fitness values themselves. The percentage of raven's weak population with the most horrible fitness values could perform in limitation, the others could be greedy. In the proposed algorithm, rather than the possibility parameter  $\text{prob}_{\text{stop}}$ ,  $\text{food}_{\text{st}}$  parameter is exploited as the control parameter decided using the number of food residual for the raven, so that the parameter value was minimized at each iteration.

The  $\text{food}_{\text{st}}$  parameter was updated using

$$\text{food}_{\text{st}+1,k} = \text{food}_{\text{max}} \times \left( \frac{\text{max}_{\text{it}} - j}{\text{max}_{\text{it}}} \right) \quad (19)$$

Where  $\text{food}_{\text{max}}$  indicates the utmost number of food for the raven and it is assigned as one. Additionally,  $\text{max}_{\text{it}}$  indicates the utmost number of iterations and  $j$  is the current iteration. Fig 3 demonstrates the flowchart of the proposed algorithm.





**Fig. 3.** Flow chart of the proposed Hybrid CS and enhanced RRO algorithm

## 6. Results and Discussions

### 6.1 Simulation Setup

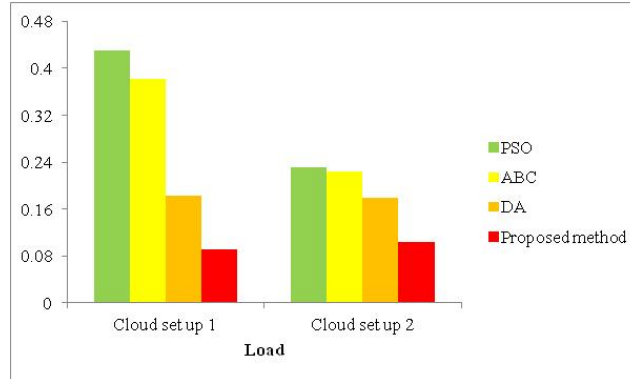
In this section, the outcomes of the proposed algorithm were explained for several cloud setups. In this work, numerous simulation setups were exploited to estimate the performance of the proposed technique.

The outcomes of the proposed technique were compared with the other conventional techniques, like ABC, DA, and PSO. The outcomes attained by the proposed technique were evaluated with the measures, like load and the number of reallocation of tasks. Moreover, the experimentation outcomes of the proposed technique for cloud setup 1 and set up 2 were explained for altering tasks.

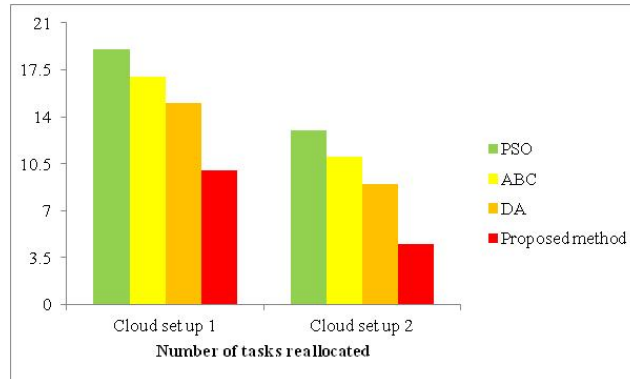
## 6.2 Comparative Evaluation

Fig. 4 demonstrates performance evaluation of the proposed technique with the conventional methods like DA, ABC, and PSO with respect to the Load for both the cloud set up 1 and 2.

Fig. 5 demonstrates the performance evaluation of the proposed technique over the conventional techniques like DA, ABC, and PSO with respect to the number of reallocation tasks for both cloud set up 1 and 2.



**Fig. 4.** Comparative evaluation of the proposed technique concerning the Load



**Fig. 5.** Comparative evaluation of the proposed technique concerning the number of tasks reallocated

## 7. Conclusion

In this work, for load balancing, a novel optimization algorithm CS-ERRO was presented. The proposed CS-ERRO, by hybridizing CS, and enhanced the RRO algorithm, to discover the optimal VM for migration. The proposed CS-ERRO technique exploits the multi-objective design due to the factors, like machine ability, cost of task migration, a VM load, and the QoS parameters. Additionally, multi-objective design exploits the parameters of the frequency scaling in order to refine the selection criterion. The simulation of the proposed method was evaluated for several cloud setups and a diverse number of tasks. The experimentation outcomes of the proposed CS-ERRO method were evaluated on account of values of the load and the allocation of tasks, and outcomes were evaluated over the conventional algorithms, like ABC, PSO, and DA. The proposed method had attained overall superior performance with the least load values.

## Compliance with Ethical Standards

**Conflicts of interest:** Authors declared that they have no conflict of interest.

**Human participants:** The conducted research follows the ethical standards and the authors ensured that they have not conducted any studies with human participants or animals.

## References

- [1] N. S. Dey and T. Gunasekhar, "A Comprehensive Survey of Load Balancing Strategies Using Hadoop Queue Scheduling and Virtual Machine Migration," *IEEE Access*, vol. 7, pp. 92259-92284, 2019.
- [2] Y. Dong, G. Xu, Y. Ding, X. Meng and J. Zhao, "A 'Joint-Me' Task Deployment Strategy for Load Balancing in Edge Computing," *IEEE Access*, vol. 7, pp. 99658-99669, 2019.
- [3] U. Bulkan, T. Dagiuklas, M. Iqbal, K. M. S. Huq, A. Al-Dulaimi and J. Rodriguez, "On the Load Balancing of Edge Computing Resources for On-Line Video Delivery," *IEEE Access*, vol. 6, pp. 73916-73927, 2018.
- [4] Nikolaos Leontiou, Dimitrios Dechouniotis, Spyros Denazis, Symeon Papavassiliou, "A hierarchical control framework of load balancing and resource allocation of cloud computing services", *Computers & Electrical Engineering*, Volume 67, April 2018, Pages 235-251.
- [5] V. Priya, C. Sathiya Kumar, Ramani Kannan, "Resource scheduling algorithm with load balancing for cloud service provisioning", *Applied Soft Computing*, Volume 76, March 2019, Pages 416-424.
- [6] Mahya Mohammadi Golchi, Shideh Saraeian, Mehrnoosh Heydari, "A hybrid of firefly and improved particle swarm optimization algorithms for load balancing in cloud environments: Performance evaluation" *Computer Networks*, Volume 162, 24 October 2019.
- [7] Divya Chaudhary, Bijendra Kumar, "Cost optimized Hybrid Genetic-Gravitational Search Algorithm for load scheduling in Cloud Computing", *Applied Soft Computing*, Volume 83, October 2019.
- [8] M. Vanitha, P. Marikkannu, "Effective resource utilization in cloud environment through a dynamic well-organized load balancing algorithm for virtual machines", *Computers & Electrical Engineering*, Volume 57, January 2017, Pages 199-208.
- [9] Wu D, Kong F, Gao W, Ji Z, "Improved chicken swarm optimization", *IEEE international conference on cyber technology in automation, control, and intelligent systems (CYBER)*, page no: 681-686, 2015.
- [10] Torabi S, Safi-Esfahani F, "Improved Raven Roosting Optimization algorithm (IRRO)", *Swarm Evolut Comput*, vol. 40, page no:144-154, 2018.
- [11] Singh, A., Juneja, D. and Malhotra, M., "Autonomous agent based load balancing algorithm in cloud computing," *Procedia Computer Science*, Volume 45, page no.832-841, 2015.
- [12] B. P. Rimal, E. Choi, and I. Lumb, "A Taxonomy and Survey of Cloud Computing Systems," In *Proceedings of Fifth International Joint Conference on INC, IMS and IDC*, Seoul, page no. 44-51, 2009.
- [13] Razzaghzadeha, S., HabibizadNavinb, A., MasoudRahmania, A., and Hosseinzadeh, M., "Probabilistic modeling to achieve load balancing in expert clouds," *Ad Hoc Network*, Volume. 59, page no.12-23, 2017.
- [14] G. Xu, Y. Ding, J. Zhao, L. Hu, and X. Fu, "A Novel Artificial Bee Colony Approach of Live Virtual Machine Migration Policy Using Bayes Theorem," *The Scientific World Journal*, 2013.
- [15] J. Shen, T. Zhou, D. He, Y. Zhang, X. Sun and Y. Xiang, "Block Design-Based Key Agreement for Group Data Sharing in Cloud Computing," *IEEE Transactions on Dependable and Secure Computing*, Volume. 16, Issue. 6, page no. 996-1010, Nov.-Dec. 2019.
- [16] Z. Yin, F. R. Yu, S. Bu and Z. Han, "Joint Cloud and Wireless Networks Operations in Mobile Cloud Computing Environments With Telecom Operator Cloud," *IEEE Transactions on Wireless Communications*, Volume. 14, Issue. 7, page no. 4020-4033, July 2015.
- [17] J. N. Khasnabish, M. F. Mithani and S. Rao, "Tier-Centric Resource Allocation in Multi-Tier Cloud Systems," *IEEE Transactions on Cloud Computing*, Volume. 5, Issue. 3, page no 576-589, July-Sept. 2017.
- [18] N. Grozev and R. Buyya, "Performance Modelling and Simulation of Three-Tier Applications in Cloud and Multi-Cloud Environments," *The Computer Journal*, Volume. 58, Issue. 1, page no. 1-22, Jan. 2015.
- [19] H. Rong, H. Wang, J. Liu and M. Xian, "Privacy-Preserving k-Nearest Neighbor Computation in Multiple Cloud Environments," *IEEE Access*, Volume. 4, page no. 9589-9603, 2016.
- [20] K. K. Nguyen and M. Cheriet, "Environment-Aware Virtual Slice Provisioning in Green Cloud Environment," *IEEE Transactions on Services Computing*, Volume. 8, no. 3, page no. 507-519, 1 May-June 2015.
- [21] Vidyadhari Ch, Sandhya N, Premchand P, "A Semantic Word Processing Using Enhanced Cat Swarm Optimization Algorithm for Automatic Text Clustering", *Multimedia Research*, Volume 2, Issue 4, October 2019.