Load balancing using a Hybrid Hydrozoan and Sea Turtle Foraging Optimization Algorithm in FOG Computing

D.Karthika Renuka
{dkr.cse@pshtech.ac.in}

Associate Professor, PSG College of Technology, Coimbatore, Tamil Nadu, India

Abstract. Internet of Things (IoT) is network of huge and intricate devices, wherein fog computing systems is significantwith the intention of handling the data flow of such huge and intricate network. Customarily, in fog computing environments, load balancing delinquent ariseswhen a large count of new IoT user requests are linked with specific fog nodes. So, a well-organized load balancing tactic is needed in fog computing. Therefore, in this manuscript, a Hybrid Hydrozoan and Sea Turtle Foraging optimization Algorithmsbased improved energy efficient Resource AllocationMethod to load balance in fog computing (IDRAM-LB-FC-Hyb-HySTFOA)is effectively proposed for reducing task waiting time, Load Balancing Rate, Scheduling Time, Delay and Energy Consumption. The evaluation metrics, like Response Time, Load Balancing Rate, Scheduling Time, Delay,and Energy Consumption are analyzed. Then the simulation performance of the Improved energy efficient resource allocation method for load balancing in Fog computing using Hybrid Hydrozoan and Sea Turtle Foraging optimization Algorithms(IDRAM-LB-FC-Hyb-HySTFOA) provide 32.82% and 25.32% low delay, 38.22% and 25.46% low energy consumption compared with the existing methods, like dynamic resource allocation method based load balancing using genetic algorithm in fog computing environment (DRAM-LB-FC-GA) and Load balancing in the fog nodes using particle swarm optimization-based enhanced dynamic resource allocation method (EDRAM-LB-FC-PSOA).

Keywords: Internet of Things, fog computing systems, improved energy efficient Resource Allocation Method, Hybrid Hydrozoan and Sea Turtle Foraging optimization Algorithms, load balancing, Delay, Energy Consumption.

1 Introduction

Generally, the cloud computing system has a congestion issue due to multiple data approaches from numerous sources, which origins high latency to instantaneous approachable devices [1-2]. To overcome these issues, fog computing system delivers responses, because it is systematized closer the end users edge [3]. Fog computing is a decentralized computing system, which is in between the cloud computing system and IoT devices. In fog computing, the information is attained and scrutinized at the border of the system [4-5]. Normally, in fog computing environments, load balancing problem is caused by huge count of new internet of things user requests are associated with particular fog nodes [6-7].

ICCAP 2021, December 07-08, Chennai, India

DOI 10.4108/eai.7-12-2021.2314533

So, awell-organized load balancing tactic is needed in fog computing, which automatically expands the QoS aspects[8-9]. This is the dissemination of tasks procedure amidnumerous fog nodes along thesustenance of well-organized load balancing tactic [10-11]. The well-organized load balancing tactic needed the following requirements such as minimum waiting time of task, minimum usage of resources, high throughput, No impasses, sophisticated scalability, fault tolerant, low network delay. For minimum waiting time of task, well-organized load balancing tactic handles theprecedencecredentials of everyinternet of things device named tasks trying to attach the Fog node depending on the precedence with usage of resources [12, 13].

Generally, resource usage is a significant task of Fog node for transferring the unexploited resources of newly tasks [14].If well-organized load balancing tactic handles the minimum waiting time of task and minimum usage of resources perfectly, then the fog computing provides high throughput [15-16]. Similarly, during the load balancing process, it is compulsory to evade impasses during resource usage [17]. The volume of Fog node can be increased while future tasks increase [18, 19]. Generally, Fog nodes need processing even its component's catastrophe condition. Similarly, it deals 4G/5G network topology, also reports network latency during load balancing [20]. The benefit of fog computing with well-organized load balancing tactic has less waiting time, propinquity, real time interaction and multiple occupancy. But it has some issues such as energy consumption, load balancing rate and delay.

In this manuscript, IDRAM-LB-FC-Hyb-HySTFOA is effectively proposed to overcome these issues, also reducing task waiting time, Load Balancing Rate, Scheduling Time, Delay, and Energy Consumption. The major contribution of this manuscript is,

- In this manuscript, Hybrid Hydrozoan and Sea Turtle Foraging optimization Algorithms based Improved energy efficient Resource Allocation Method is effectively proposed for load balancing in fog computing.
- Initially, the manipulator assigns number of tasks to the Responsibility Administrator.
- At the same time, Resource Info Benefactor catalogues possessions from the centers of Cloud Data.
- The data regarding with tasks with resources are acquiesced for Resource Scheduler.
- Resource scheduler organizes the accessible resources at descendent direction according to their resource usage.
- Then the resource scheduler provides information regarding the tasks, then resources given into resource engine.
- After getting the information, it allocates tasks to the resources in accordance with well-organized list.
- Throughout task implementation, the data about the resources status is correspondingly directed to the Resource Load Administrator and Resource Power Administrator.
- The resource power Administrator maintains the resource on / off power status depending on resource load status.
- Afterwards the execution of efficacious task, the resource engine compiles the output,and then sends the outcome to the manipulator.
- To optimize the IDRAM-LB-FC-Hyb-HySTFOA is implemented.
- This Hybrid Hydrozoan with Sea Turtle Foraging optimization Algorithms affords early convergence and attains the optimized fitness solution by minimizing Energy Consumption $(Energy_{consumption})$, Delay $(delay)$, responsetime (RT), Schedule Time (ST) ***and***Load BalanceRate (L).

- Then the proposed approach is simulated by iFogSim toolkit.
- The evaluation metrices, viz Response Time, Load Balancing Rate, Scheduling Time, Delay, Energy Consumption with count of tasks are analyzed.
- Then the simulation performance of IDRAM-LB-FC-Hyb-HySTFOA is analyzed and it was compared with the existing methods, like DRAM-LB-FC-GA [21] and EDRAM-LB-FC-PSOA [22].

The remaining manuscript is structured as: Section 2 presents the Literature survey. Section 3 illustrates about proposed Load balancing using a Hybrid Hydrozoan and Sea Turtle Foraging Optimization Algorithm in FOG Computing. Section 4 demonstrates the resultswith discussion. Section 5 concludes the manuscript.

2. Literature survey

Several researches works were presented in the literature based on resource allocation procedure to load balance in Fog environment, a certain works are reviewed here,

In 2020, *Talaat, et.al.,* [21] have presented a load balancing with optimization strategy (LBOS) utilizing reinforcement learning at the environment of fog computing. LBOS notices network traffic constantly, and the information regarding every server load were collected, the receiving requests were handled, dispenses them amid the accessible servers based on the dynamic resource allocation approach. Therefore, during peak time, it improves the performance uniformly. Consequently, the presented method was effective in real-time methodfor fog computing.

In 2021, *Baburao, et.al.,* [22] have presented a load balancing on the fog nodes utilizing particle swarm optimization-depend enhanced dynamic resource allocation method (EDRAM). To deal load balance competently, a particle swarm optimization-depend EDRAM was presented,it diminishes task waiting time, latency, consumption of network bandwidth, increases the Quality of Experience. The EDRAM supports to resource usage by eradicating the longer waiting time on task allocation.

In 2020,*Kaur, et.al.,* [23] have presented an energy-aware load balancing in fog cloud computing. Where,the presented method was used to scientific workflows at fog-cloud computing environment. Moreover, a load-balancing approach was suggested for fog environment. The simulation was done by iFogSim. Load balancing at fog layer supports accurate consumption of resources, which automatically decreases the latency and improves the quality of service.

In 2020, *Singh, et.al.,* [24] have suggested a leveraging energy-efficient load balancing approaches in fog computing. Where, load balancing have deliberated, alsoits comparative analysis was made. In fog computing environments, Round Robin load balancing was simply load balancing strategy to be applied. The difficult of source IP hash load balance approach in every modification, it redirects the information to anyone with various server, sodesirableon fog networks.

3. Proposed Method

This section describes about the proposed IDRAM-LB-FC-Hyb-HySTFOA. Generally, load balancing tactics is an operative technique for apportioning the resources to the manipulator on depending on energy consume. The work flow of proposed method is depicted in Figure 1, which consist of IoT devices, Fog computing infrastructure and cloud data center.

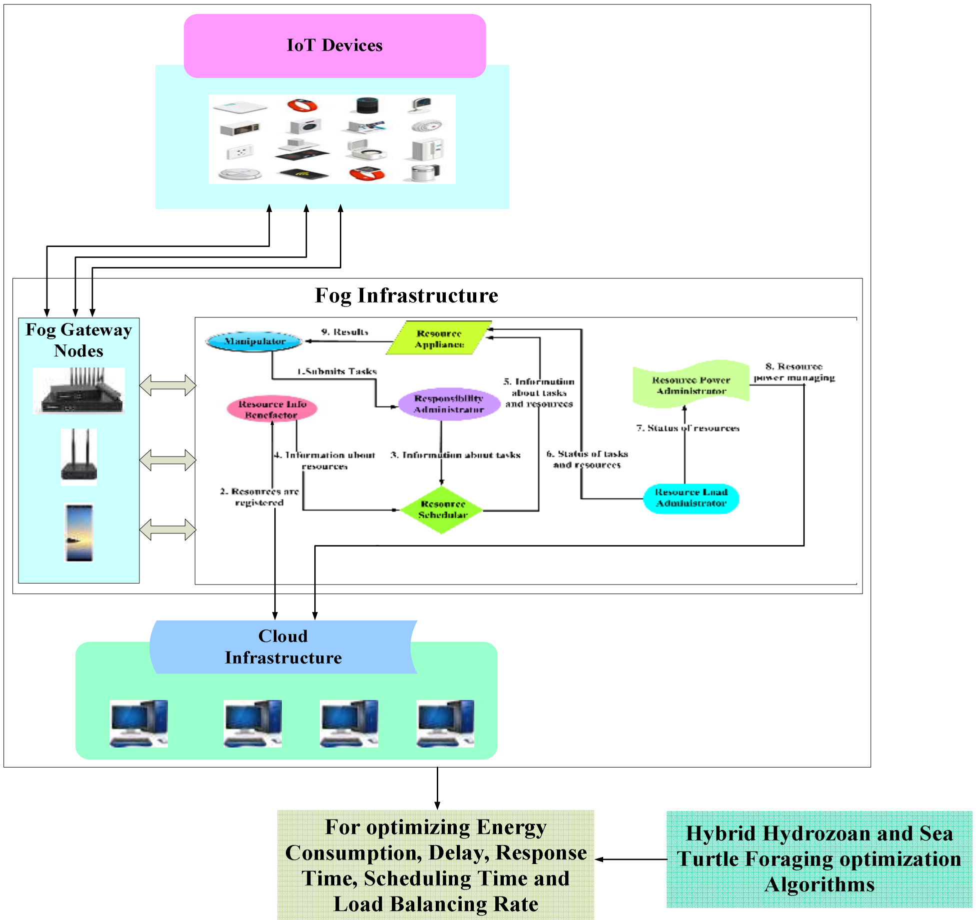


Figure 1: Block diagram of proposed system architecture

3.1. IoT devices

In IoT devices, it empowers the sensors, apparatus, and possessions for exchanging information along Fog computing scheme through the Fog Gateways internet.

3.2 Fog computing infrastructures

InFog computing infrastructures, it consists of Responsibility Administrator, Resource Info Benefactor, Resource Scheduler, Resource Load Administrator, Resource Power Administrator and resource engine. In the beginning, the manipulator will consent more number of tasks to the Responsibility Administrator. At the same time, one or more manipulators can submit tasks to the Responsibility Administrator. Here, to every task, the energy consume is predefined with the help of instruction that it encompasses. Simultaneously, Resource Info Benefactor catalogues the resources from Cloud Data Centers, which not only register the resources but also provide the available resources information.

Here, for each accessible resource, energy consume is predefined with the help of instruction restricted at that particular task. The Resource Scheduler acquires information from Responsibility Administrator and resource from Resource Info Benefactor. Here, task is arranged by the value of energy consumption in ascending order. Similarly, resources are arranged by the value of energy consumption in descending order. Then, Resource Scheduler conveys the tasks with resource data for Resource Engine.

The resource engine allocates the task according to the sorted list of the resource engine and then it starts execution. The status of tasks with resources is shared with Resource Load Administrator by the resource engine. After victorious task completion, the Resource Engine also precedes the outcomes to the manipulator. While the execution of task the main function of Resource Load Administrator is inspect the resource status. It transfers to the resource power Administrator, after inspecting the status. The resource power Administrator maintains the resource on / off power status depending on resource load status. Afterwards the execution of efficacious task, the resource engine compiles the output, then forward the output to the manipulator. By this, the proposed IDRAM-LB-FC-Hyb-HySTFOA approach provides better results.

3.3Cloud infrastructures

Mostly cloud data centre acts as manager amongst the manipulator request and the applications of server for manipulator request. If cloud server did not properly execute to the manipulator request, it sends the messages to total cloud data centre. Like this, the manipulator requested task is executed in the proposed system architecture.

To optimize the improved energy efficient Resource Allocation Scheme for load balancing in fog computing, Hybrid Hydrozoan and Sea Turtle Foraging optimization Algorithms is implemented. This Hybrid Hydrozoan and Sea Turtle Foraging optimization approach affords initial integration, then attains the optimum fitness solution by minimizing Energy Consumption $(Energy_{consumption})$, Delay $(delay)$, response time (RT), Scheduling Time (ST) ***and*** Load Balancing Rate (L). The energy consumption can be calculated with the help of equation (1)

$$Energy_{consumption} = \sum_{l=1}^{n} E_{transmission(l)} + E_{execution(l)} + E_{sensing(l)}$$

(1)

Here, theoverall energy consumption is the combination of transmitting, implementation, sensing everyunique task. Then, Delay represents the time duration for implementing the entire task allocated to the fog node. It is calculated with the help of equation (2)

$$delay = CT - SST$$

(2)

Where, CT indicates current time, SST represents the starting time of the simulation. Then, Response time represents time taken by user request till the response arrival on requesting interface. This is the sum of request time (TR) as well as behavior of processing time (TBP), it is determined with the help of equation (3)

$$RT = TR + TBP$$

(3)

Then Scheduling time is a primary factoron allocating resources. It computes starting times ofevery task (STE) as well as simulation ending time (SET). It is scaled by equation (4)

$$ST = STE + SET$$

(4)

Then Load balancing rate represents the load distribution amid the fog nodes to lessen the problem of congestion. It is assessed by every fog node current workload, then it is determined with the help of equation (5)

$$L = \frac{N_{TS}}{N_{FNS}}$$

(5)

Where N_{TS} implies number of tasks and N_{FNS} implies number of fog nodes. The proposed algorithm is the joint execution of both the Hydrozoan and sea turtles and hence it is called HA-STFA approach. The hydrozoans reproduction procedures and sea turtles foraging behaviorare considered for resolving the issues of global optimization. Here,at the Hydrozoan approach,the sea turtle foraging algorithms is implanted. The major objective is to given theoptimal balance amid the exploration and exploitation capabilities. Hydrozoan algorithm depends upon clonal selection, cross over, mutation operators. These three operators are to be efficient for exploring the problem space. Owing to its own gain, theHybrid Hydrozoan with Sea Turtle Foraging optimization Algorithmsis selected. It takes less iteration time than other tuning models,viz grid search, random search fordetermining the optimumfilter parameters. Hybrid Hydrozoan and Sea Turtle Foraging optimization Algorithmsprovides new early convergence and achieves the optimized fitness solution. `

3.3.1 Step by step process of Hybrid Hydrozoan and Sea Turtle Foraging optimization Algorithmsfor optimizing $Energy_{consumption}$, $delay$, RT, ST *and* L

Here, the stepwise procedure for getting the optimumEnergy Consumption $(Energy_{consumption})$, Delay $(delay)$, response time (RT), Scheduling Time (ST) and Load Balancing Rate (L) of IDRAM-LB-FC-Hyb-HySTFOA is discussed and the related flow chart are represented in Figure 2.First, Hybrid Hydrozoan with Sea Turtle Foraging optimization Algorithmsgenerates the uniformly distributed initial population of Hydrozoan and Sea Turtle. After the initialization process, generates randomly the parameters and calculate the fitness function. By utilizing thereproduction manners of hydrozoanstogether withforaging behavior of sea turtles, it optimizes the Energy Consumption $(Energy_{consumption})$, Delay $(delay)$, response time (RT), Scheduling Time (ST) and Load Balancing Rate (L) of improved energy efficient Resource Allocation Scheme to load balance in fog computing. The optimal solution is updated via the Hybrid Hydrozoan and Sea Turtle Foraging optimization Algorithms. The above process repeats until acquiring the feasible optimal solution. The detailed step procedure is illustrated as follows,

Step 1: **Initialization**

Initialize the initial population of Hydrozoan and Sea Turtle

Step 2: **Random generation**

After the procedure of initialization, the input parameters have generated randomly. Here, the best fitness values of each Hydrozoan and Sea Turtleare selected depending on explicit hyper-parameter situation.

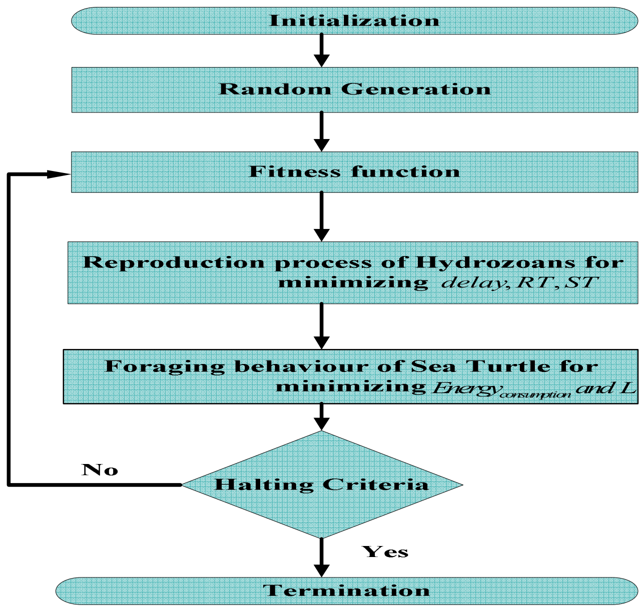


Figure 2:Flow chart for Hybrid Hydrozoan and Sea Turtle Foraging optimization

Algorithms for optimizing $Energy_{consumption}$, $delay$, RT, ST **and** L

Step 3: **Fitness Function**

The arbitrary quantity of resolutions is engendered after the initialized values. Then the fitness function is scaledusing the givenequation (6)

$$fitness\ \ function = Minimize\ [Energy_{consumption}, delay,\ RT,\ ST\ and\ L)$$

(6)

Step 4: **Reproduction processes of Hydrozoan for minimizing** $delay,\ RT,\ ST$

In this step, for each hydrozoan l, the quantity of buds can be divided and the Bud_l can be calculated with the help of equation (7-8)

$$DM_l = F_l(s) - Median$$

(7)

$$Bud_l = \begin{cases} 0; & if\ DM_l < 0 \\ 1; & if\ DM_l = \min^{+} \\ 3; & if\ DM_l = \max^{+} \\ 2; & otherwise \end{cases}$$

(8)

From the medusa position updation, the delay can be minimized utilizing equation (9)

$$Delay = M_i(t+1) = M_i(t) + \eta VC_i(t) + C_{ij}(t)[K_J - M_i(t)] \quad (9)$$

From equation (9) $C_{ij}(t)$ represents the odor strength of food source K_J, $VC_i(t)$ represents the velocity of the each medusa, $M_i(t)$ and represents the newly created medusa, η represents constant. The count of genes that every parent contributes to the offspring can minimizes the response time (RT), Scheduling Time (ST) and it is defined by equation (10-11)

$$RT = \left[\frac{x-y}{x}\right]\left[\frac{d}{2}\right] + \left[\frac{d}{2}\right] \quad (10)$$

$$ST = d - RT \quad (11)$$

Where, (RT) and (ST) specifies count of genes that the higher and lower fitness parent contribute to the offspring respectively, x implies fitness of strong parent, y implies fitness of weak parent, d implies dimension of the search space.

Step 5: **Foraging behaviour of Sea Turtle for minimizing** $Energy_{consumption}$ *and* L

In this step, from the Foraging behaviour of Sea Turtle can minimize the $Energy_{consumption}$ *and* L is discussed. From the sea turtle velocity updation, the energy consumption is minimized utilizing equation (12)

$$Energy_{consumption} = V_i(t+1) = V_i(t) + V_i(t) + \left[\frac{f(T_i(t)) - f(T_i(t-1))}{f(T_i(t-1))}\right][T_i(t) - T_i(t-1)] \quad (12)$$

Where, $T_i(t)$ represents the position of turtle i at time t, $f(T_i(t))$ represents the fitness of turtle i at time t. From the sea turtle velocity updation, the load balancing rate is minimized through equation (13)

$$T_i(t+1) = T_i(t) + \eta V_i(t+1) + C_{ij}(t)[K_j - T_i(t)] \quad (13)$$

Step 7: Termination

In this step, the optimal Energy Consumption $(Energy_{consumption})$, Delay $(delay)$, response time (RT), Scheduling Time (ST) and Load Balancing Rate (L) values of improved energy efficient Resource Allocation Scheme iteratively repeat step 3 till met the halting criteria to load balance in fog computing**.**

The runtime overhead of proposed Improved energy efficient Resource Allocation System to load balance in fog computing denotes O(z). Hence,the proposed IDRAM-LB-FC-Hyb-HySTFOA is arrangedevery assets through its utilization, because effective use of resources. The similar overhead is unimportant in proposed IDRAM-LB-FC-Hyb-HySTFOA. The obtainable resources and the algorithm workactively, as the proposed method firstly categories. When a newly resource comes to set of organized assets it is located in the suitable position animatedly.

4. Result with discussion

Here, simulation result of Hybrid Hydrozoan and Sea Turtle Foraging optimization Algorithmsbased Improved energy efficient Resource Allocation Scheme to load balanceat fog computing is discussed. The simulations are performed PC using 2.50 GHz CPU, Intel Core i5, 8GB RAM, Windows 7. The proposed model is simulated iniFogSim toolkit. Here, the evaluation metrics, like Response Time, Load Balancing Rate, Scheduling Time, Delay, Energy Consumption depending on count of tasks are analyzed. Then the simulation performance of IDRAM-LB-FC-Hyb-HySTFOA is analyzed and it was compared with the existing methods, like DRAM-LB-FC-GA [21] and EDRAM-LB-FC-PSOA [22]. Table 1 displays the simulation parameter.

Table 1: Simulation parameter

	Simulation tool	iFogSim	
System configuration	IDE	Net beans 8.0	
	Language	Java	
	Topology type	Fully connected	
		Memory	9 Megabyte
		RAM capacity	15
		Bandwidth	1500KBs
	Number of users	100	
	Count of fog nodes	64 (8*8)	
	Fog node	Storage cost	0.0012
		Resource cost	3.1
		Storage capacity	1.1Gigabyte
		Bandwidth	9990KBS
	Mobile devices	Delay	125ms (among the proxy server-cloud)
	Mobile devices	Delay	1.5ms (amid the mobiles and the parent fog device)

4.1 Simulation phase: performance comparison of various methods

Figure 3-4 shows the Simulation result for Improved energy efficient Resource Allocation Scheme to load balanceat fog computing with various methods. The various evaluation metrics like Response Time, Load Balancing Rate, Scheduling Time, Delay, Energy Consumption based on number of tasks are analyzed in this segment. Here, the performance of proposed IDRAM-LB-FC-Hyb-HySTFOA method is compared with the existing methods like DRAM-LB-FC-GA method and EDRAM-LB-FC-PSOA method by varying number of tasks.

Figure 3(a) depicts the response time performance analysis. At number of task 10, the proposed IDRAM-LB-FC-Hyb-HySTFOA method attains 15.2483% and 6.06208% lower response time compared with the existing method such as DRAM-LB-FC-GA and EDRAM-LB-FC-PSOA respectively. At number of task 30, the proposed IDRAM-LB-FC-Hyb-HySTFOA method attains 17.3306% and 7.22201% lower response time compared with the existing method such as DRAM-LB-FC-GA and EDRAM-LB-FC-PSOA respectively. At number of task 50, the proposed IDRAM-LB-FC-Hyb-HySTFOA method attains 14.5476% and 7.59792% lower response time compared with the existing method such as DRAM-LB-FC-GA and EDRAM-LB-FC-PSOA respectively.

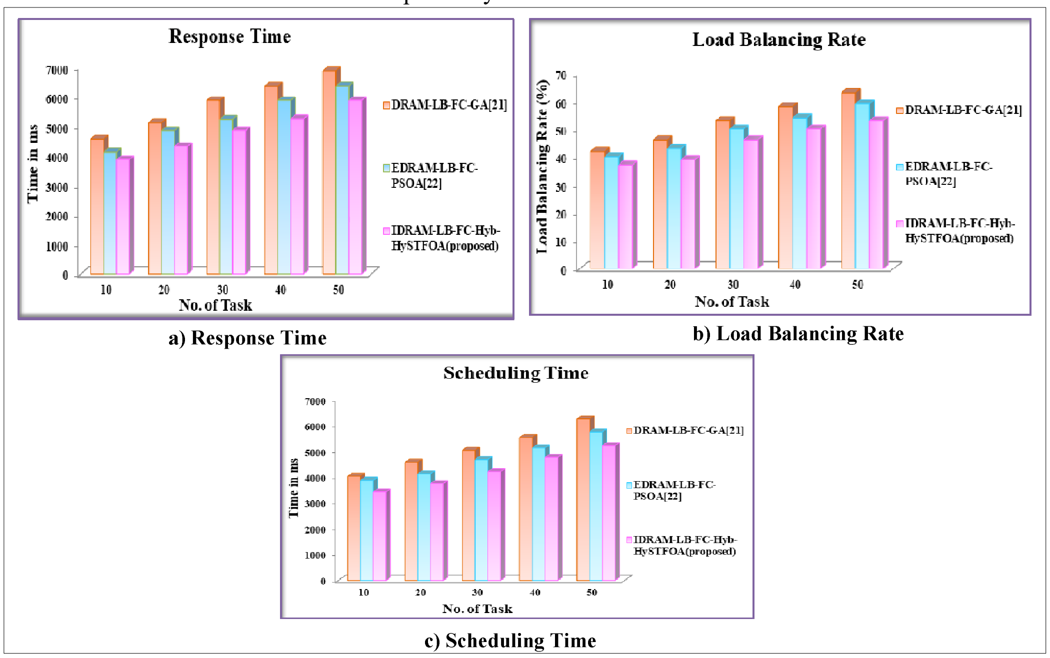
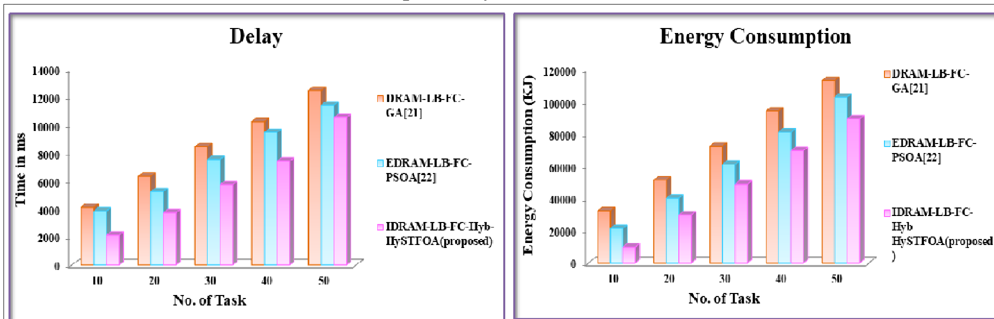


Figure 3: Performance Analysis of Response Time, Load Balancing Rate and Scheduling Time

Figure 3(b) shows the load balancing rate performance analysis. At number of task 10, the proposed IDRAM-LB-FC-Hyb-HySTFOA method attains 11.9048% and 7.5% lower load balancing rate compared with the existing method such as DRAM-LB-FC-GA and EDRAM-LB-FC-PSOA respectively. At number of task 30, the proposed IDRAM-LB-FC-Hyb-HySTFOA method attains 13.2075% and 8% lower load balancing rate compared with the existing method such as DRAM-LB-FC-GA and EDRAM-LB-FC-PSOA respectively. At number of task 50, the proposed IDRAM-LB-FC-Hyb-HySTFOA method attains 15.873% and 10.1695% lower load balancing rate compared with the existing method such as DRAM-LB-FC-GA and EDRAM-LB-FC-PSOA respectively.

Figure 3(c)represents the scheduling time performance analysis. At number of task 10, the proposed IDRAM-LB-FC-Hyb-HySTFOA method attains 14.9068% and 11.6129% lower scheduling time compared with the existing method such as DRAM-LB-FC-GA and EDRAM-LB-FC-PSOA respectively. At number of task 30, the proposed IDRAM-LB-FC-Hyb-HySTFOA method attains 16.256% and 9.841% lower scheduling time compared with the existing method such as DRAM-LB-FC-GA and EDRAM-LB-FC-PSOA respectively. At number of task 50, the proposed IDRAM-LB-FC-Hyb-HySTFOA method attains 16.4557% and 9.2428% lower scheduling time compared with the existing method such as DRAM-LB-FC-GA and EDRAM-LB-FC-PSOA respectively.



a) Delay b) Energy consumption

Figure 4: Performance Analysis of Delay and Energy Consumption

Figure 4(a) displays the performance analysis of delay. At number of task 10, the proposed IDRAM-LB-FC-Hyb-HySTFOA method attains 48.0844% and 44.7341% lower delay compared with the existing method such as DRAM-LB-FC-GA and EDRAM-LB-FC-PSOA respectively. At number of task 30, the proposed IDRAM-LB-FC-Hyb-HySTFOA method attains 32.0672% and 23.9401% lower delay compared with the existing method such as DRAM-LB-FC-GA and EDRAM-LB-FC-PSOA respectively. At number of task 50, the proposed IDRAM-LB-FC-Hyb-HySTFOA method attains 15.4278% and 7.66874% lower delay compared with the existing method such as DRAM-LB-FC-GA and EDRAM-LB-FC-PSOA respectively.

Figure 4(b) shows the performance analysis of energy consumption. At number of task 10, the proposed IDRAM-LB-FC-Hyb-HySTFOA method attains 69.6059% and 54.0543% lower energy consumption compared with the existing method such as DRAM-LB-FC-GA and EDRAM-LB-FC-PSOA respectively. At number of task 30, the proposed IDRAM-LB-FC-Hyb-HySTFOA method attains 32.6742% and 20.3977% lower energy consumption compared with the existing method such as DRAM-LB-FC-GA and EDRAM-LB-FC-PSOA respectively. At number of tasks 50, the proposed IDRAM-LB-FC-Hyb-HySTFOA method attains 20.9906% and 13.0247% lower energy consumption compared with the existing method such as DRAM-LB-FC-GA and EDRAM-LB-FC-PSOA respectively.

5. Conclusion

In this,Hybrid Hydrozoan and Sea Turtle Foraging optimization Algorithmsbased Improved energy efficient Resource Allocation Scheme to load balance is successfully implemented to deal every fog computing system, also verdicts the appropriate fog node for task obligations. The experimental results show that the proposed model performance is very effectual in terms of evaluation metrics,viz Response Time, Load Balancing Rate, Scheduling Time, Delay, Energy Consumption. Here the performance of proposed IDRAM-LB-FC-Hyb-HySTFOA method provide 16.04% and 8.507% low response time, 13.99% and 8.47% low load balancing rate and 15.93% and 9.42% low scheduling time compared with the existing methods like DRAM-LB-FC-GA method and EDRAM-LB-FC-PSOA method.In future work, data security-basedfognode computing using the Improved energy efficient Resource Allocation Scheme to load balance approach for the application ofReal time is considered for reducing latency.

References

[1] Abdali, T.A.N., Hassan, R., Aman, A.H.M. and Nguyen, Q.N., 2021. Fog Computing Advancement: Concept, Architecture, Applications, Advantages, and Open Issues. IEEE Access, 9, pp.75961-75980.

[2] Tortonesi, M., Govoni, M., Morelli, A., Riberto, G., Stefanelli, C. and Suri, N., 2019. Taming the IoT data deluge: An innovative information-centric service model for fog computing applications. Future Generation Computer Systems, 93, pp.888-902.

[3] Singh, S.P., Nayyar, A., Kumar, R. and Sharma, A., 2019. Fog computing: from architecture to edge computing and big data processing. The Journal of Supercomputing, 75(4), pp.2070-2105.

[4] Moura, J. and Hutchison, D., 2020. Fog computing systems: State of the art, research issues and future trends, with a focus on resilience. Journal of Network and Computer Applications, p.102784.

[5] Abdel-Basset, M., Mohamed, R., Elhoseny, M., Bashir, A.K., Jolfaei, A. and Kumar, N., 2020. Energy-aware marine predators algorithm for task scheduling in IoT-based fog computing applications. IEEE Transactions on Industrial Informatics, 17(7), pp.5068-5076.

[6] Sriraghavendra, M., Chawla, P., Wu, H., Gill, S.S. and Buyya, R., 2022. DoSP: A Deadline-Aware Dynamic Service Placement Algorithm for Workflow-Oriented IoT Applications in Fog-Cloud Computing Environments. In Energy Conservation Solutions for Fog-Edge Computing Paradigms (pp. 21-47). Springer, Singapore.

[7] Verba, N., Chao, K.M., Lewandowski, J., Shah, N., James, A. and Tian, F., 2019. Modeling industry 4.0 based fog computing environments for application analysis and deployment. Future Generation Computer Systems, 91, pp.48-60.

[8] Talaat, F.M., Saraya, M.S., Saleh, A.I., Ali, H.A. and Ali, S.H., 2020. A load balancing and optimization strategy (LBOS) using reinforcement learning in fog computing environment. Journal of Ambient Intelligence and Humanized Computing, pp.1-16.

[9] Singh, S.P., Kumar, R., Sharma, A. and Nayyar, A., 2020. Leveraging energy-efficient load balancing algorithms in fog computing. Concurrency and Computation: Practice and Experience, p.e5913.

[10] Sulimani, H., Alghamdi, W.Y., Jan, T., Bharathy, G. and Prasad, M., 2021. Sustainability of Load Balancing Techniques in Fog Computing Environment. Procedia Computer Science, 191, pp.93-101.

[11] Rehman, A.U., Ahmad, Z., Jehangiri, A.I., Ala'Anzy, M.A., Othman, M., Umar, A.I. and Ahmad, J., 2020. Dynamic energy efficient resource allocation strategy for load balancing in fog environment. IEEE Access, 8, pp.199829-199839.

[12] Adil, M., Song, H., Ali, J., Jan, M.A., Attique, M., Abbas, S. and Farouk, A., 2021. EnhancedAODV: A Robust Three Phase Priority-based Traffic Load Balancing Scheme for Internet of Things. IEEE Internet of Things Journal.

[13] Khalil, U., Ahmad, A., Abdel-Aty, A.H., Elhoseny, M., El-Soud, M.W.A. and Zeshan, F., 2021. Identification of trusted IoT devices for secure delegation. Computers & Electrical Engineering, 90, p.106988.

[14] Li, X., Liu, Y., Ji, H., Zhang, H. and Leung, V.C., 2019. Optimizing resources allocation for fog computing-based internet of things networks. IEEE Access, 7, pp.64907-64922.

[15] Shafiq, D.A., Jhanjhi, N.Z., Abdullah, A. and Alzain, M.A., 2021. A Load Balancing Algorithm for the Data Centres to Optimize Cloud Computing Applications. IEEE Access, 9, pp.41731-41744.

[16] Sohani, M. and Jain, S.C., 2021. A Predictive Priority-Based Dynamic Resource Provisioning Scheme With Load Balancing in Heterogeneous Cloud Computing. IEEE Access, 9, pp.62653-62664.

[17] Afzal, S. and Kavitha, G., 2019. Load balancing in cloud computing–A hierarchical taxonomical classification. Journal of Cloud Computing, 8(1), pp.1-24.
[18] Baek, J.Y., Kaddoum, G., Garg, S., Kaur, K. and Gravel, V., 2019, April. Managing fog networks using reinforcement learning based load balancing algorithm. In 2019 IEEE Wireless Communications and Networking Conference (WCNC) (pp. 1-7). IEEE.
[19] Hejja, K., Berri, S. and Labiod, H., 2021. Network slicing with load-balancing for task offloading in vehicular edge computing. Vehicular Communications, p.100419.
[20] [20] Mumtaz, T., Muhammad, S., Aslam, M.I. and Mohammad, N., 2020. Dual connectivity-based mobility management and data split mechanism in 4G/5G cellular networks. Ieee Access, 8, pp.86495-86509.
[21] Talaat, F.M., Saraya, M.S., Saleh, A.I., Ali, H.A. and Ali, S.H., 2020. A load balancing and optimization strategy (LBOS) using reinforcement learning in fog computing environment. Journal of Ambient Intelligence and Humanized Computing, pp.1-16.
[22] Baburao, D., Pavankumar, T. and Prabhu, C.S.R., 2021. Load balancing in the fog nodes using particle swarm optimization-based enhanced dynamic resource allocation method. Applied Nanoscience, pp.1-10.
[23] D. S. Vijayan, A. Leema Rose, S. Arvindan, J. Revathy, C. Amuthadevi, "Automation systems in smart buildings: a review", Journal of Ambient Intelligence and Humanized Computing https://doi.org/10.1007/s12652-020-02666-9
[24] Singh, S.P., Kumar, R., Sharma, A. and Nayyar, A., 2020. Leveraging energy-efficient load balancing algorithms in fog computing. Concurrency and Computation: Practice and Experience, p.e5913.