Received: August 17, 2023. Revised: October 7, 2023.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Movement Mode Harmony Search Based Multi-objective Firefly Algorithm Feature Selection for Detecting the Security Threats in Virtual Machine

Adalagere Nemirajaiah Naveen Kumar[1] **Madigahalli Mallegowda[2]**
Atur Venkateshmurthy Krishna Mohan[3]* **Kannughatta Narasimhamurthy Shreenath[3]**
Channa Krishna Raju[4]

[1]*Department of Computer Science and Engineering, Sri Siddhartha Academy of Higher Education, Tumakuru, India*
[2]*Department of Computer Science and Engineering, M S Ramaiah Institute of Technology, Bengaluru, India*
[3]*Department of Computer Science and Engineering, Siddaganga Institute of Technology, Tumakuru, India*
[4]*Department of Computer Science and Engineering, Sri Siddhartha Institute of Technology, Tumakuru, India*
* Corresponding author's Email: Krishnamohan_av@sit.ac.in

Abstract: The energy-efficient and secure allocation of virtual machines (VMs) plays an important role at the data center. As cloud computing continues to expand rapidly and the number of cloud users increases day by day, the issue of high energy consumption in complex cloud data centers has become a significant concern. To address this challenge, the consolidation of virtual machines (VMs) emerges as a crucial strategy for optimizing cloud resources efficiently. In this study, a novel security evaluation method is proposed to assist the model available for the virtualized system. A multi-objective model-based firefly algorithm (FA) and harmony search (HS) algorithm are used for the system configuration in VM migration is proposed to measure the security threats such as denial of service (DoS), distributed denial of service (DDoS) and Man-in-the-middle attack. The proposed method also decreases the power consumption, network usage and resource wastage in virtual machines. The proposed algorithm achieves better results compared to other existing methods by utilizing the number of virtual machine blocks with cost migration. The experimental results shows that the proposed FA+HS delivers the performance metrics such as makespan, execution cost and resource utilization and achieved at the 1000VMS of 950, 0.001 and 62 respectively, which ensures the better results compared with the existing methods such as whale optimization genetic algorithm (WOGA), multi-objective whale optimization algorithm-based differential evolution (M-WODE) and joint task scheduling and virtual machine placement (JTSVMP).

Keywords: Cloud resources, Denial of service, Distributed denial of service, Firefly algorithm, Harmony search algorithm, Security, Virtual machine.

1. Introduction

In recent years, to counter the security threats in cloud computing, major cloud providers have adopted various security protection mechanisms based on the characteristics of cloud platforms [1]. Cloud computing is a model for delivering on-demand computational resources and services named storage of data as well as computing power over the internet [2]. The consolidation of virtual machines plays an important role in efficiently using cloud resources [3]. The VM provides on-demand resources to the users and offers a higher quality of service (QoS) [4]. In a cloud environment, the VMs are created and then mapped to the host or physical machines, when computing requests are made. The problem of seeking a suitable host is called the VM placement problem [5]. The providers of the cloud are required to schedule user's task into VM as well as strictly locate the VMs to physical hosts. Each cloud user requests a set of VMs to run its application on a cloud data center, where VMs usually communicate with each other to process the application [6]. In recent years, to counter the security threat in cloud computing environments, major cloud providers have adopted various conventional security protection mechanisms based on the characteristics of their cloud platforms

The technology of virtualization permits cloud providers to run number of VMs on a physical machine [7]. An efficient VM placement allows the cloud providers to the cloud users to increase the benefit over reducing the energy consumption, physical machine resource wastage, and increasing the QoS [8]. The scheduling process of the VM is supported by to demand of the physical machines in a specific data center, and the necessity to be fulfilled with the demanded resources [9]. The VM algorithm provides a better scheduling strategy for the cloud data center and executes the plan in an efficient manner and time [10]. The VMs are moved to prevent the attacks is the standard technique for co-resident VMs attacks or available physical hosts [11]. The evaluation techniques are essential to evaluate how much the security to be quantified and it evaluates the throughout availability in the VM migration scheduling [12, 13]. The metaheuristics algorithms of FA and HS algorithms are utilized in various applications such as medical applications, wireless communications and so on. The meta-heuristic algorithms and its improvement must have a perfect balance between exploring and exploiting operations to do both global and local searches well. As a result, many researchers proposed a hybrid algorithm that combines two or more algorithms to improve algorithm search and find optimal path planning with a speed up convergence rate. [14-16]. The proposed method’s goal is detecting the security threats in the virtual machines by using the feature selection-based multi-objective method of firefly and harmony search optimization algorithm. The advantages include enhanced feature selection, improved security level, efficiency and versatility. These advantages collectively contribute to the development of accurate and efficient classification. The major contributions of this proposed study are explained as follows:

- The architecture of the virtual machine (VM) migration-based system model in cloud security is proposed.
- A novel method of multiple-objective optimization algorithms such as the firefly algorithm and harmony search (HS) Algorithm is proposed to obtain optimal VM solutions for cloud users and providers.
- The VM method is evaluated using various performance metrics such as utilization of CPU, makespan, migration cost as well and response time.

The rest of the paper is arranged as follows: section 2 discussed the recent research on task assignment problems. Section 3 provides the proposed work of this paper. The results and analysis are discussed in section 4 and the conclusion of this paper is presented in section 5.

2. Literature survey

Saxena [17] implemented a new method of secure and multiple objective virtual machine placement (SM-VMP) with an effective virtual machine (VM) migration. The VMP is carried out by applying the proposed WOGA, inspired by whale evolutionary optimization and non-dominated sorting based genetic algorithms. This method ensured an energy-efficient distribution of physical resources among VMs that emphasized timely and secure user application by reducing inter-communication delay. The advantage of this method was minimizing the security attack cost of inter-communication, consumption of power, enhancement in resource utilization, and execution time. But this method does not provide the better optimal solution for the multi-objective problem.

Duan [18] implemented a design of multiple-level security threats and examined the protection of security under some attack situations for Ethereum. This method utilized the ten security attacks at various Ethereum levels, which mainly consist of three layers such as application, network as well and smart contract. This method has prohibited attacks by enhancing Ethereum contract quality. The advantage of this method was enhanced security level, versatility, and detection tools efficiency but this method consumed a lot of time and. The protection method of the replay attack extended the signing and a verification of signature, which required to perform difficult calculations, thus it consumed a much time.

Rana [19] implemented a method of M-WODE for solving the problem of VM scheduling. The WOA had gained a notable reputation for solving optimization problems. The unique bubble-net hunting behaviour and fast convergence of the algorithm led to the development of a M-WODE technique to solve the VM scheduling problem. The strategy of the differential evolution (DE) was utilized to change incidentally caused solutions formed using WOA to establish distinction as well as M-WODE local search strength. The advantage of this method was provided better performance and better solutions. However, this method does not measure the impact of some adaptive control parameters on the M-WODE method to evaluate the performance.

Alboaneen [20] implemented a novel metaheuristic method to enhance the JTSVMP in the center of cloud data. The problem of JTSVMP consists of two parts such as VM placement as well as task scheduling. These problems can be considered to be a single problem to be solved by utilizing a metaheuristics optimization algorithm. The optimization algorithms such as glow-worm swarm optimization (GSO), moth-flame glowworm swarm optimization (MFGSO) and genetic algorithm (GA). The advantage of this method was reducing the cost execution and increasing the usage of Physical Host (PH) resources. But this method consumed a much time to organize the task scheduling and VM as well as it had the problem of flexibility.

Feizollahibarough and Ashtiani [21] implemented a plan for security-aware virtual machine placement to minimize the co-location problems for susceptible virtual machines. The four attributes produced to minimize the problems consist of virtual machine vulnerability, importance level, cumulative vulnerability as well as physical machineability. The proposition of a computational model that can take into account four different attributes to enable security awareness for VM placement strategy. The advantage of this method was minimizing the provided cloud data center security problem compared to other methods. But this method has a lack of attention to resource allocation as well as do not support to the real-world applications.

Gharehpasha [22] implemented a new method using the hybrid discrete multiple-object whale optimization algorithm, a multiple verse optimizer using chaotic functions for enhancing placement in cloud data centers. An initial objective of this method was to reduce the consumption of power and then break the wastage of resources and manage based on virtual machine placement. The second objective is to cut the resource wastage and managing resources using the optimal placement of virtual machines over physical machines in cloud data centers. The advantage of this method was provided better performance results. However, this method only focussed on minimizing the consumption of power of the data centre but it does not concentrate on the detecting the security problem.

Alsharif and Rawat [23] implemented a cloud-supported machine learning (ML) with a security solution for resource-limited IoT devices, which enhances effective machine learning using intrusion detection system (IDS). The framework of cloud mobility as a service (MAAS) was implemented to offload heavy-weight ML actions to resources of cloud computing as well as to arrange internet of things (IoT) devices. The advantage of this method achieves better performance as well as better accuracy in network traffic. However, the IDS operation had been limited to the feature extraction from network traffic and anomaly detection.

The limitations found from the related work are insecurity in web services, lack of effective encryption and authentication, required a large amount of communication costs, lack of attention to resource allocation, computational time as well as required number of ways to solve metaheuristics algorithms. To overcome this, the multiple objective method of firefly algorithm (FA) and harmony search (HS) Algorithm is proposed for security threats in virtual machines.

3. Proposed Methodology

In this section, the architecture of the Virtual Machine migration-based system model is shown in Fig. 1. A group of architectures is studied in the proposed system, that ranges from one to a number of physical machine pools. This studied architecture has three main characteristics; 1. A physical machine pool's distinctive hypervisor with hypervisor is the insider attacker's objective. 2. The migration of the Virtual Machine is desirable between the physical machine pool. 3. As a minimum physical machine is desirable in each pool migration receiving. This studied architecture is shown in Fig. 1.

3.1 Threat model

The insider attackers aim to target the hypervisor, which acts as a middleware between the Virtual Machine (VM) and the Physical Host (PH), to accomplish their attack objectives. It an essential to investigate the attacker or threaten the tenant Virtual Machines once the control of the hypervisor is suspected. The resignation is not done by the attacker till the success of the attack is assumed in the proposed system. There are multiple phases in insider attacks such as the exploration that demonstrates through the attacker's requirement to recognize the hypervisor host's variant. And the attack demonstrates once the malicious activity is accomplished along the attacker, that is against the attacker of the host. Consecutively, the success chance of an attack is larger once the provided the longer duration similar to the physical host.

Received: August 17, 2023. Revised: October 7, 2023.

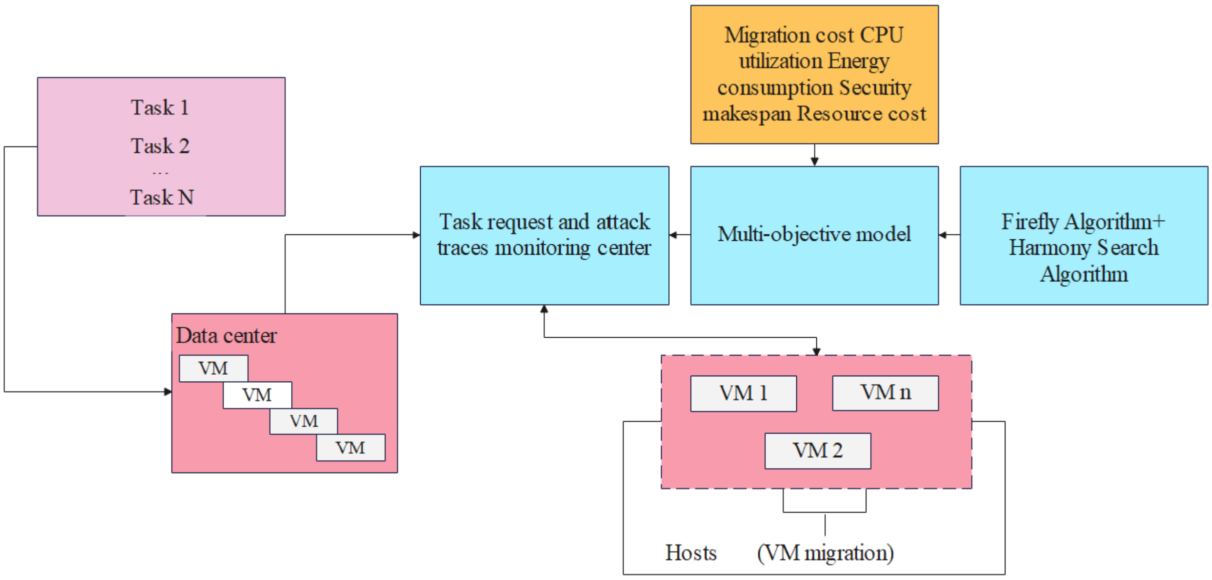

Figure. 1 The architecture of the virtual machine migration-based system model

Additionally, can be continued the error and try method, and the failed attempts are to be ignored; that is the knowledge accumulated by the attacker when the attacker reconnoiters the hypervisor of the host.

3.2 Cloud platform

Examine the cloud platform, where the resource pools are collected from the server S_i and resources set is S_i where, $i = 1, 2, \ldots, n$. The virtual machine is accepted on the server at each node and is described as $S_i = VM_{i,1}, \ldots, VM_{i,k}$. The $VM_{i,k}$ states the virtual machine over S_i, each server consists of huge resources such as network, CPU, and memory. The host resolution is important for the appropriation of the virtual machine. This requires some conditions and this is essential to manage some objectives of virtual machines over the cloud. The virtual machine set supports in planning of the objective as mentioned along virtual machine allocation of resources as well as requirements of the objective. Each objective is described using specific parameters such as consumption of energy, utilization of CPU, security, cost of migration and resource, and makespan are represented as key terms for successful migration of the Virtual Machine and which is described in the problem of optimization.

3.3 Multiple objective constraints

The prime objective depends on minimization and it is demonstrated as shown in Eq. (1) as;

$$\text{Objective} = \min\{\text{Energy}, U_{CPU}, security, migration\ cost, resource\ cost, makespan\} \quad (1)$$

1. The utilization of the CPU is the time consumed by the CPU to compute specific tasks. When there is a rise in resource demand in the cloud, Virtual machines are mapped to the obtainable server. It is examined as essential for CPU using the eliminating delay objective and which is expressed in Eq. (2);

$$U_{CPU}(task\ allocation) = \sum count_{CPU}(T_1 + \cdots + T_n) \quad (2)$$

2. Consumption of energy shows a direct relationship with the utilization of the physical machine. It is represented as the total number of consumptions of energy for a specific time and which is expressed in Eq. (3) as;

$$energy\ consumption = 0.1 \log(count_{CPU}) + Energy_i + 0.1 \quad (3)$$

3. In the constitution of security, the virtual machine migration is accomplished securely without all threats intervention such as attacks of denial of service, MITM, or both. This is estimated using the difference between the physical machine as well as the security of the task and the estimation is completed using the risk probability.

Received: August 17, 2023. Revised: October 7, 2023.

4. The Makespan measure is a hypothesis in scheduling the completion time of a certain task, which is measured as the maximal wall time to the overall task blocks and it is shown in Eq. (4) as;

$$Makespan\ (tasks) = max \sum_{i=1}^{N} W_{time} \quad (4)$$

5. The cost of migration is evaluated using a scheduling algorithm and it is the recurrence of relocation execution as well as it is connected with the capacity. It is called the sum of cost sustained through migration from one task to another task with a virtual machine and is demonstrated in Eq. (5) as;

$$Migration\ cost = \sum_{T_i=1}^{n} migration\ cost\ (T_{i=1,2,..,n}) \quad (5)$$

6. In resource cost, the process of relocation depends on:

- Content of memory
- Update of memory beyond all Virtual Machine
- The ability of the data transfer
- Remaining concern between source and server.

It is represented as the sum of resource cost sustained with particular resources to accomplish related tasks and which is expressed in Eq. (6) as;

$$Resource\ cost\ (tasks) = \sum resource_{cost}(T_1 + \cdots + T_2) \quad (6)$$

3.4 Firefly algorithm and harmony search algorithm

In this section, a new method is proposed with the firefly algorithm (FA) and harmony search (HS) algorithm to evaluate security as well as the process of the virtual machine migration.

3.4.1. Firefly algorithm (FA)

The firefly algorithm (FA) is a population-based intelligence algorithm. This algorithm is based on the flashing ability for the algorithm development. The FA is the natural fireflies' features as well as efficient for accomplishing activities like communicating as well as foraging using various fireflies to identify them. The generating mechanism of the new solution of FA consists of two components such as attractiveness as well as luminance, these two components are continuously adapted for optimization. The attractiveness is utilized to supervise the moves of individual distance and luminance is utilized to demonstrate the firefly quality, next, the supervision of the moves of individual distance. Every firefly endures to fly against the brightest individual in the population in their search area. The enhanced solution of the local or global is identified over the iteration of location. The fireflies make random movements when there are no brighter individuals in the range of search.

The firefly’s mathematical method is fully based on the following regulations:

1. The fireflies are attracted to each other.
2. The attraction between the fireflies is inversely proportional to the distance between them.
3. The attraction level is described by the brightness level. The low-brightness fireflies are to be attracted to the high-brightness fireflies. The fireflies move randomly, because of their equal brightness level, so it fully depends on the objective function.

The relative luminance and attractiveness of fireflies’ formula can be expressed in Eqs. (7) and (8) as;

$$I_i = I_0 * e^{-\gamma r_{i,j}} \quad (7)$$

$$\beta(r_{ij}) = \beta_0 * e^{-\gamma\, r_{ij}^2} \quad (8)$$

Where, I_0 represents maximum luminance; γ represents the coefficient of absorption that the light strength; $r_{i,j}$ represents spatial interval; β_0 represents the maximum degree of attraction.

3.4.2. Harmony search (HS) algorithm

The harmony search (HS) is a metaheuristic algorithm, that is introduced to improve the performance utilized better harmonies by musicians. In this method, the performances of the music are enhanced using increasing the harmony effects to achieve approval from audiences. For the optimistic approach of the natural music process, the HS is operated via two different choices the possibility of memory usage and the generation of random notes. The working basis of HS optimization is identical to the song composition with musicians utilizing a number of notes. In reality, the HS algorithm is motivated using a procedure of generating a new song. In this procedure, the composer generates a new song using the musical notes as well as pieces, the best notes as well as pieces are taken to the limited memory of the composer.

 DOI: 10.22266/ijies2024.0229.16

Received: August 17, 2023. Revised: October 7, 2023.

Table 1. Comparison of utilization of CPU with number of VM blocks

Virtual Machine Blocks	**ABC**	**GWO**	**ACO**	**FA**	**HS**	**FA+HS**
10	935	3700	1300	1100	900	890
20	1700	3200	1500	1400	1200	1300
30	1600	3300	1100	900	1100	1000
40	1600	1800	1600	1000	1250	1150
50	980	2400	1311	1300	1500	1400

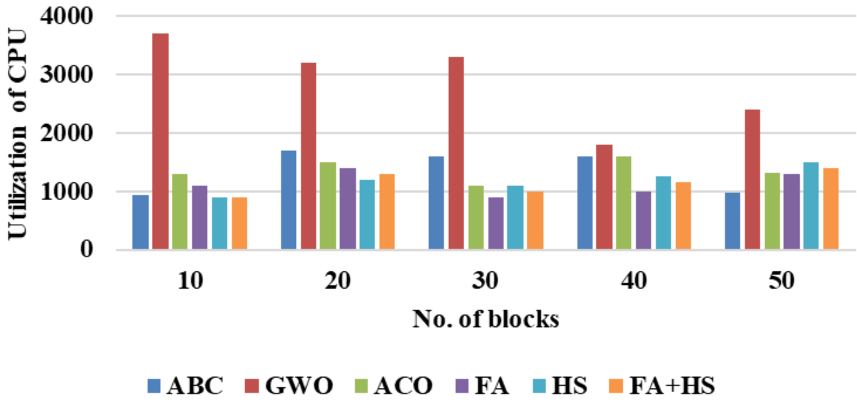

Figure. 2 Comparison of utilization of CPU with number of VM blocks

The matrix of harmony memory (HM), which stores the problem values in this algorithm, is designed to imitate the situation that plays the notes from the musician's mind in the process of music-making. The matrix of HM is represented in Eq. (9) as;

$$HM = \begin{bmatrix} x_1^1, x_2^1, \dots \dots \dots, x_n^1 \\ x_2^2, x_2^2, \dots \dots \dots, x_n^2 \\ x_1^{HMS}, x_2^{HMS}, \dots \dots \dots, x_n^{HMS} \end{bmatrix} \quad (9)$$

Where, HMS represents HM matrix size.

The general HS algorithm contains four steps as;

Step 1: The harmony search memory (HM) should be initialized. The amount of randomly produced solutions is produced by problem constraints.

Step 2: Design a novel solution from the harmony search memory.

Step 3: The novel solution from the previous step is estimated and next, updating the HM. If the novel solution is better than the worst solution, that is kept in HM, the two solutions get swapped.

Step 4: The computation as well as the process of this algorithm iterated. In that, the step 2 and 3 loops are repeated till achieving the stopping conditions like maximum iteration.

In the spirit of evolutionary computing, the optimization algorithm explores the objective space for the optimal regions and then evaluates the found solutions with respect to the fitness conditions. The FA+HS program halts the candidate solutions have achieved the best fitness function during the iterations. To obtain the multi-objective optimization model, the fitness function can be formulated in given Eq. (10) as;

$$Fitness\ f(x) = min\ f(y) + min\ f(z) \quad (10)$$

Where, $f(y)$ – makespan; $f(z)$ – cost function. The fitness function is designed to evaluate the quality of feasible solutions to minimize the makespan and cost function. The obtained results are further processed to evaluate the performance of the proposed method.

4. Results

The divine virtual machine migration over malevolent threats is provided for organizing the efficiency of the model and simulation is done by utilizing the environment of MATLAB 2020a. The divine model performance is examined using number of metrics like CPU utilization, makespan, migration cost as response time. The proposed FA with HS algorithm is compared with different existing methods like artificial bee colony (ABC), grey wolf optimization (GWO) as well as ant colony optimization (ACO). The estimation results may change from one Virtual machine to another.

Table 1 represents the comparison of the utilization of CPU with the 10-50 virtual machine blocks. The VM blocks 10, 20, 30, 40 and 50 are utilized with the ABC, GWO as well as ACO. The proposed method achieves 890 in 10VM, 1300 in 20VM, 1000 in 30VM, 1150 in 40 and 1400 in 50VM. The CPU utilization of the proposed method achieves the 1400 VM compared to the other methods.

Received: August 17, 2023. Revised: October 7, 2023.

Table 2. Comparison of makespan with the number of VM blocks

Virtual Machine Blocks	ABC	GWO	ACO	FA	HS	FA+HS
10	285	525	370	290	260	200
20	330	390	333	245	340	320
30	360	600	310	320	287	290
40	420	540	280	276	310	360
50	320	610	385	290	320	310

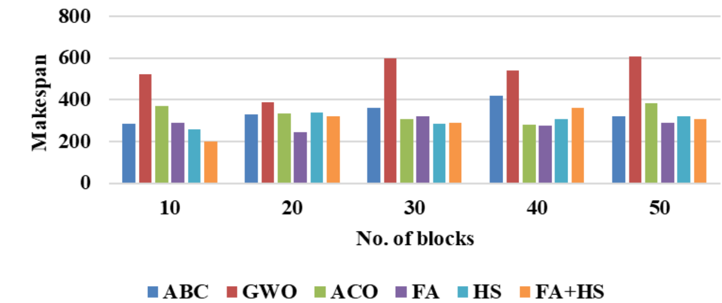

Figure. 3 Comparison of makespan with number of VM blocks

Table 3. Comparison of cost of migration with number of VM blocks

Virtual Machine Blocks	ABC	GWO	ACO	FA	HS	FA+HS
10	290	230	260	285	298	175
20	270	250	250	245	230	200
30	220	370	250	275	240	240
40	230	245	270	220	250	160
50	220	275	210	260	200	190

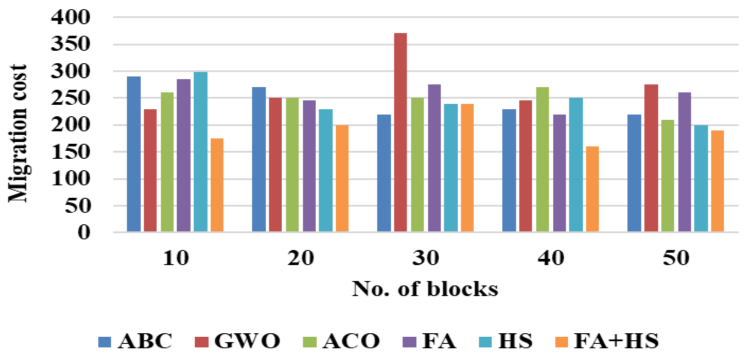

Figure. 4 Comparison of cost of migration with number of VM blocks

The graphical representation of CPU utilization is shown in Fig. 2.

Table 2 represents the comparison of the makespan with the 10-50 Virtual Machine blocks. The VM blocks 10, 20, 30, 40, and 50 are utilized with the ABC, GWO as well as ACO. The proposed method achieves 300 in 10VM, 370 in 20VM, 290 in 30VM, 360 in 40 and 400 in 50VM. The CPU utilization of the proposed method achieves the 400 VM compared to the other methods. The graphical representation of CPU utilization is shown in Fig. 3.

Table 3 represents the comparison of the cost of migration with the 10-50 Virtual Machine blocks. The VM blocks 10, 20, 30, 40 and 50 are utilized with the ABC, GWO as well as ACO. The proposed method achieves 95 in 10VM, 200 in 20VM, 250 in 30VM, 160 in 40 and 150 in 50VM. The CPU utilization of the proposed method achieves 150 VM compared to the other methods. The graphical representation of CPU utilization is shown in Fig. 4.

Table 4 represents the comparison of response time with the 10-50 Virtual Machine blocks. The VM blocks 10, 20, 30, 40 and 50 are utilized with the ABC, GWO as well as ACO. The proposed method achieves 22.5 in 10VM, 29.5 in 20VM, 25.5 in 30VM, 24.0 in 40 and 26.2 in 50VM. The CPU utilization of the proposed method achieves the 1400 VM compared to the other methods. The graphical representation of CPU utilization is shown in Fig. 5.

Table 4. Comparison of response time with number of VM blocks

Virtual Machine Blocks	ABC	GWO	ACO	FA	HS	FA+HS
10	52.2	56.9	31.7	32.8	29.6	22.5
20	30.6	43.7	34.1	31.2	30.7	29.5
30	29.8	55.2	29.4	32.8	37.9	25.5
40	28.9	34.6	49.3	25.5	24.7	24.0
50	34.5	41.7	45.5	26.5	27.8	26.2

Table 5. Comparative analysis of proposed method with existing methods

Author	Method	Number of VMs	Number of tasks	Makespan	Execution cost	Resource Utilization
Saxena [17]	WOGA	200	N/A	N/A	N/A	63
		400				63
		600				63.5
		800				64
		1000				64
Rana [19]	M-WODE	30	200	150	N/A	N/A
			400	400		
			600	550		
			800	850		
			1000	1100		
Alboaneen [20]	JTSVMP	N/A	100	300	0.042	N/A
			200	750	0.030	
			300	800	0.002	
			400	900	0.003	
			500	1000	0.002	
Proposed FA+HS	FA+HS	200	200	250	0.020	59
		400	400	600	0.003	59.5
		600	600	750	0.001	61
		800	800	800	0.030	61.7
		1000	1000	950	0.001	62

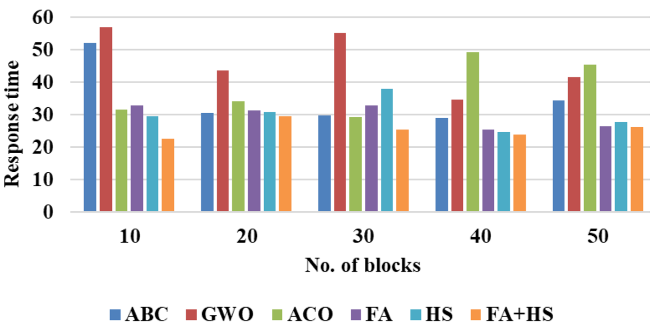

Figure. 5 Comparison of response time with number of VM blocks

4.1 Comparative analysis

This section shows the comparative analysis of the proposed combination of the firefly algorithm (FA) with the harmony search (HS) algorithm in terms of method, number of tasks, and makespan are shown in Table 5. The number of tasks as well as the makespan of the proposed method is more efficient than the existing methods. Table 5 shows the comparative analysis of the proposed method with recent research methods.

4.2 Discussion

This section provides the discussion about proposed method and its result comparisons. The Firefly algorithm has the limitation such as high probability of being trapped in local optima, because they are local search algorithms. To solve this problem, the harmony search algorithm has the advantage is that utilizes the real random numbers and it based on the global communication among the swarming particles. To utilize this advantage of HS, this research proposed a novel method by combining the FA and HS to selecting the features efficiently in the threats. In FA-HS, the whole HM is divided into number of sub- HMS, each of which can evolve independently. However, these sub-HMs will form the HM again after searching for the optimal solutions in their own regions. The meta-heuristic algorithms and its improvement must have a perfect balance between exploring and exploiting operations to do both global and local searches well. As a result, many researchers proposed a hybrid algorithm that combines two or more algorithms to improve

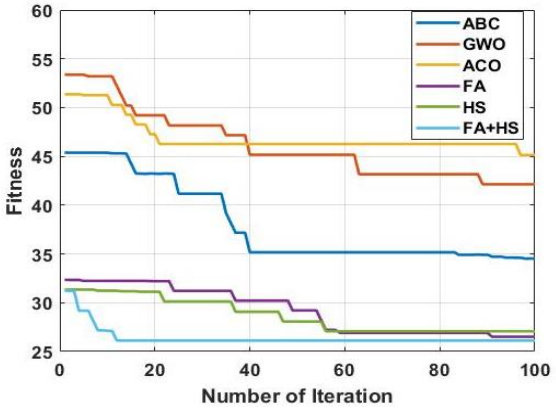

Figure. 6 Graphical representation of fitness function of the proposed method

algorithm search and find optimal path planning with a speed up convergence rate [14, 15, 16]. Fig. 6 shows the graphical representation of fitness function of the proposed method with the existing methods. The proposed method achieved the best fitness values at the maximum iteration. The proposed method effectively minimized the fitness value compared to the existing methods such as ABC, GWO, ACO, FA and HS. The WOGA [17] method had utilized the 200, 400, 600, 800 and 1000 VMs and number of tasks and it achieves the resource utilization of 63, 63, 63.5, 64 and 64 respectively. The M-WODE [19] method had generated the number of tasks of 200, 400, 600, 800 and 1000 and it achieves the makespan of 150, 400, 550, 850 and 1100 on the 30 VMs respectively. The JTSVMP [20] method (IMFGSO) has generated the tasks range from 100 to 500 and it executes the makespan of 450,900, 1500, 1700 and 2100 for all tasks on the VM. Further, it minimizes the cost of 0.042, 0.030, 0.002, 0.003 and 0.002 respectively. The proposed FA+HS overcomes these existing model limitations and generates the 200, 400, 600, 800 and 1000 VMs and number of tasks and it executes the makespan of 250, 600, 750, 800 and 950 for all tasks on the VM. Further it minimizes the execution cost of 0.020, 0.003, 0.001, 0.030 and 0.001 and resource utilization of 59, 59.5, 61, 61.7 and 62 on all VMs respectively. The proposed method achieved the minimum makespan, execution cost and resource utilization on all VMs results by utilizing the hybrid optimization algorithms.

5. Conclusion

The virtual machine (VM) allocation on physical machines in cloud data centers is a significant challenge in the cloud. In this paper, a new evaluation method to benefit the availability of the model for the virtualized system is proposed. A new method of the multiple-objective model such as firefly algorithm (FA) and harmony search (HS) algorithm for the configuration of the system in migration of VM is proposed to evaluate the security threats like MITM, DoS as well as DDoS. The combination of FA and HS algorithms has provided a better result in optimizing the VM problem achievement. The proposed method aims to decrease power consumption, wastage of resources, and consumption of energy in virtual machine migration. The proposed method utilizes the up to 1000 VMs and number of tasks up to 1000 to evaluate the performance. The experimental results shows that the proposed FA+HS delivers the performance metrics such as makespan, execution cost and resource utilization and achieved the at the 1000VMs 950, 0.001 and 62 respectively, which ensures the better results compared with the existing methods such as WOGA, M-WODE and JTSVMP. The future work is to identify the external threats for improving the performance model.

Conflicts of interest

The authors declare no conflict of interest.

Author contributions

For this research work all authors' have equally contributed in Conceptualization, methodology, validation, resources, writing—original draft preparation, writing—review and editing.

References

[1] N. Chandrakala and V. Enireddy, “A Heuristic Model for Securing Virtual Machine Migration During Attack Traces in Cloud Environment”, *Indian Journal of Computer Science and Engineering*, Vol. 13, No. 4, pp. 1264-1276, 2022.

[2] Z. Zhang, Z. Yang, X. Du, W. Li, X. Chen, and L. Sun, “Tenant-Led Ciphertext Information Flow Control for Cloud Virtual Machines”, *IEEE Access*, Vol. 9, pp. 15156-15169, 2021.

[3] B. Pourghebleh, A. A. Anvigh, A. R. Ramtin, and B. Mohammadi, “The importance of nature-inspired meta-heuristic algorithms for solving virtual machine consolidation problem in cloud environments”, *Cluster Computing*, Vol. 24, No. 3, pp. 2673-2696, 2021.

[4] E. G. Radhika and G. S. Sadasivam, “Budget optimized dynamic virtual machine provisioning in hybrid cloud using fuzzy analytic hierarchy process”, *Expert Systems with Applications*, Vol. 183, p. 115398, 2021.

[5] G. J. Mirobi and L. Arockiam, “DAVmS: distance aware virtual machine scheduling

approach for reducing the response time in cloud computing", *The Journal of Supercomputing*, Vol. 77, No. 7, pp. 6664-6675, 2021.

[6] S. Rajasoundaran, A. V. Prabu, S. Routray, S. V. N. S. Kumar, P. P. Malla, S. Maloji, A. Mukherjee, and U. Ghosh, "Machine learning based deep job exploration and secure transactions in virtual private cloud systems", *Computers & Security*, Vol. 109, p. 102379, Oct. 2021.

[7] S. Omer, S. Azizi, M. Shojafar, and R. Tafazolli, "A priority, power and traffic-aware virtual machine placement of IoT applications in cloud data centers", *Journal of Systems Architecture*, Vol. 115, p. 101996, 2021.

[8] T. Ganesan, P. Vasant, and I. Litvinchev, "Chaotic simulator for bilevel optimization of virtual machine placements in cloud computing", *Journal of the Operations Research Society of China*, Vol. 10, No. 4, pp. 703-723, 2022.

[9] L. Caviglione, M. Gaggero, M. Paolucci, and R. Ronco, "Deep reinforcement learning for multi-objective placement of virtual machines in cloud datacenters", *Soft Computing*, Vol. 25, No. 19, pp. 12569-12588, 2021.

[10] J. Doyle, M. Golec, and S. S. Gill, "Blockchainbus: A lightweight framework for secure virtual machine migration in cloud federations using blockchain", *Security and Privacy*, Vol. 5, No. 2, p. e197, 2022.

[11] S. Gharehpasha and M. Masdari, "A discrete chaotic multi-objective SCA-ALO optimization algorithm for an optimal virtual machine placement in cloud data center", *Journal of Ambient Intelligence and Humanized Computing*, Vol. 12, No. 10, pp. 9323-9339, 2021.

[12] Shalu, and D. Singh, "Artificial neural network-based virtual machine allocation in cloud computing", *Journal of Discrete Mathematical Sciences and Cryptography*, Vol. 24, No. 6, pp. 1739-1750, 2021.

[13] A. A. R. Melvin, G. J. W. Kathrine, S. S. Ilango, S. Vimal, S. Rho, N. N. Xiong, and Y. Nam, "Dynamic malware attack dataset leveraging virtual machine monitor audit data for the detection of intrusions in cloud", *Transactions on Emerging Telecommunications Technologies*, Vol. 33, No. 4, p. e4287, 2022.

[14] K. R. Gajulapalli, and M. S. Gnanadhas, "Analysis of PAPR and BER Reduction in MIMO-OFDM using Hybrid Moth Flame-Improved Firefly Algorithm", *International Journal of Intelligent Engineering & Systems*, Vol. 15, No. 4, pp. 97-105, 2022, doi: 10.22266/ijies2022.0831.10.

[15] M. N. F. Fajila, and Y. Yusof, "Hybrid Gene Selection with Mutable Firefly Algorithm for Feature Selection in Cancer Classification", *International Journal of Intelligent Engineering & Systems*, Vol. 15, No. 3, pp. 24-35, 2022, doi: 10.22266/ijies2022.0630.03.

[16] C. Gandikoti, S. K. Jha, and B. M. Jha, "Integration of Moth Flame Optimization with Improved Firefly Algorithm in Islanded Microgrid Using Renewable Sources", *International Journal of Intelligent Engineering & Systems*, Vol. 15, No. 5, pp. 1-11, 2022, doi: 10.22266/ijies2022.1031.01.

[17] D. Saxena, I. Gupta, J. Kumar, A. K. Singh, and X. Wen, "A Secure and Multiobjective Virtual Machine Placement Framework for Cloud Data Center", *IEEE Systems Journal*, Vol. 16, No. 2, pp. 3163-3174, 2022.

[18] L. Duan, Y. Sun, K. Zhang, and Y. Ding, "Multiple-layer security threats on the ethereum blockchain and their countermeasures", *Security and Communication Networks*, Vol. 2022, p. 5307697, 2022.

[19] N. Rana, M. S. A. A. Latiff, S. M. Abdulhamid, and S. Misra, "A hybrid whale optimization algorithm with differential evolution optimization for multi-objective virtual machine scheduling in cloud computing", *Engineering Optimization*, Vol. 54, No. 12, pp. 1999-2016, 2022.

[20] D. Alboaneen, H. Tianfield, Y. Zhang, and B. Pranggono, "A metaheuristic method for joint task scheduling and virtual machine placement in cloud data centers", *Future Generation Computer Systems*, Vol. 115, pp. 201-212, 2021.

[21] S. Feizollahibarough and M. Ashtiani, "A security-aware virtual machine placement in the cloud using hesitant fuzzy decision-making processes", *The Journal of Supercomputing*, Vol. 77, No. 6, pp. 5606-5636, 2021.

[22] S. Gharehpasha, M. Masdari, and A. Jafarian, "Virtual machine placement in cloud data centers using a hybrid multi-verse optimization algorithm", *Artificial Intelligence Review*, Vol. 54, No. 3, pp. 2221-2257, 2021.

[23] M. Alsharif and D. B. Rawat, "Study of Machine Learning for Cloud Assisted IoT Security as a Service", *Sensors*, Vol. 21, No. 4, p. 1034, 2021.