Materials Today: Proceedings xxx (xxxx) xxx

Contents lists available at ScienceDirect

Materials Today: Proceedings

journal homepage: www.elsevier.com/locate/matpr

A improved genetic algorithm based proficient resource management and scheduling

D. Sundaranarayana [a], A. Suresh [a], G. Saravanan [b]

[a] Department of Computer Science and Engineering, Veltech Rangaranjan Dr. Sagunthala R&D Institute of Science and Technology, India
[b] Department of Computer Science and Engineering, Erode Sengunthar Engineering College, India

ARTICLE INFO

Article history:
Received 1 November 2020
Received in revised form 11 November 2020
Accepted 13 November 2020
Available online xxxx

Keywords:
Grid Computing
Genetic Algorithms (GA)
Virtual machine (VM)
Grid Booster algorithm

ABSTRACT

Grid networking is one of the computing methods used to improve the performance, quality of service and provide the better utilization. It enables the application data in an open environment. Grid scheduling is a process of selecting which job is in queue and it is maintained which is to be next. The main goal of the scheduling is to minimize the make-span by finding the solution. Grid is basically a collection of distributed computing resources within one or multiple locations. The computer resources are collected to act as a unified processing resource. The interconnecting process of these resources are made into a unified pool involves the coordination of the usage policies, queuing issues, job scheduling, grid security and user authentication. The grid networking used here to interconnect and utilize the available processor, storage, or memory subcomponents of distributed computing systems to solve larger problems more efficiently. This Improved GA algorithm represents a balanced scheduling approach of VM resources in Grid Networking. This helps to provide high make-span of resource scheduling in comparing with Genetic Algorithm.

Selection and peer-review under responsibility of the scientific committee of the Emerging Trends in Materials Science, Technology and Engineering.

1. Introduction

The advantages of Grid Networking were cost investment funds, improved business readiness by diminishing the time, and upgraded joint effort and sharing of assets among occupations. These days a few productive booking calculations were dissected dependent on the Grid systems administration to assign framework assets with a focus on work planning [1]. When contrasted and regular PCs the variety will be high. All in all the matrix registering is an extraordinary kind of equal figuring that depends on complete PCs associated with an organization that might be a private, public, or the Internet by an ordinary organization. In this paper, the matrix put together climate is engaged with respect to the booking approaches for VM (virtual Machine) resources of an entire structure. Once in a while, the system is failing to consider the structure assortment and some genuine data lead which brings about a lopsidedness load. In customary techniques, the framework manages just the future burden expectation components. In light of the factor Virtual Machine (VM) asset assignment is made.

The organization interface creating the production equipment which is utilized to looked at the lower proficiency of planning and improving a few custom supercomputers. The fundamental constraint is that the different processors and nearby stockpiling territories don't have fast associations. This cycle is appropriate just for the various equal calculations can occur autonomously and without the need to convey transitional outcomes between processors [2]. The versatility of geologically appropriated networks is commonly great, because of the low requirement for availability between hubs comparative with the limit of the public Internet. A portion of the distinctions emerges in programming and sending. Sometimes the cost shrewd issue emerges and it is hard to get to the composed projects that run in the climate of a supercomputer. The working framework has required a program for custom utilization, or requires the program to address simultaneousness issues. In the event that an issue can be parallelized, at that point, a "flimsy" layer of the "framework" foundation can permit ordinary and independent projects. It gives the diverse aspect of a similar issue, to run on various machines. This assists with composing and investigates on a solitary ordinary machine and furthermore kills the intricacies because of numerous cases of a similar program running in a similar shared memory. Typically

E-mail addresses: drdsundar@veltech.edu.in (D. Sundaranarayana), drasuresh@-veltech.edu.in (A. Suresh)

https://doi.org/10.1016/j.matpr.2020.11.481
2214-7853/
Selection and peer-review under responsibility of the scientific committee of the Emerging Trends in Materials Science, Technology and Engineering.

the lattice processing depends on the mix of traditional working framework level virtual machine (VM) and center product to oversee VMs in a disseminated climate. Regarding security, customization and asset control were made as for the quantitative contentions. The customary multiplexing is utilized to trust and make responsibility based substance. To every client, the Virtual machines give a committed crude machine. The reflection is extremely ground-breaking in matrix registering because the clients become decoupled from the framework programming of the hidden asset and different clients sharing the asset. Because of security reasons the dependent client or application may bargain with their own working framework inside a virtual machine. To meet incredible adaptability in supporting the various working frameworks traditional virtual machine assumes a significant job.

The requirement for network figuring with an old-style VM is to give another reflection layer, with low overhead that offers usefulness that enormously disentangles tending to huge numbers of the issues of matrix processing.

2. Virtualization

Regularly virtualization is a demonstration of making a virtual instead of real. It has the capacity to run different working frameworks on a solitary framework. It is additionally a type of something, including virtual PC hardware stages [3]; working structures, storing devices, and PC network resources. It is the cycle where one PC has the demeanour of different frameworks. Virtualization is required in light of the fact that so as to improve the IT throughput and expenses by utilizing assets as a primary subject and their virtual assets were assigned. The accompanying figure shows the disconnected runtime climate, for example, visitor OS and applications. As examined above it can run numerous virtual machine frameworks (VMs) on a solitary physical framework (Fig. 1).

An advantage of Virtualization is cost decrease because of the Sharing of assets. Virtual machines are effectively segregated from one another as though they are truly isolated. It epitomizes a total figuring climate. Its fundamental preferred position is immovability, which can be relocated between various hosts.

Security Challenges in Virtualization, for example, the Trusted Computing Base (TCB) of a virtual machine is excessively huge. TCB is one of the mixes of programming and equipment that are basic to its security. Thus on the off chance that TCB is little, at that point, it is greater security [4]. A portion of the Virtualization Security Requirements are given underneath: customer utilizes the administration of distributed computing to manufacture the far off virtual machine with the accompanying perspectives, for example, secure organization interface, secure auxiliary stockpiling and secure run-time climate, for example, construct, spare, reestablish and pulverize. Another prerequisite is the secure run-time climate is the most crucial.

In the VM [5], as a methodology for authentically isolating the system resources gave by brought together worker PCs between different applications. Virtualization is a compelling method to lessen IT costs while boosting effectiveness and readiness. It isn't made only for enormous endeavors, yet for little and moderate size organizations as well. VMware virtualization can run various working frameworks and applications on a solitary server. The primary target of Grid Networking is to allow customers to take profit from these progressions, with no top to bottom information about every last one of them. The network intends to lessen the expense and to help the clients focusing on their center business. Virtualization is the standard enabling development for Grid Networking. Virtualization gives the physical structure, which is the most inflexible portion, and makes it easy to use and supervise as a delicate segment. Henceforth by doing this the virtualization gives the availability expected to quicken the undertakings and decreases cost by growing system use. Autonomic enrolling motorizes the cycle through which the customer can mastermind resources on-demand. On the off chance that the client inclusion is limited, at that point the computerization speed increments and diminishing the chance of human blunders. Virtualization makes a layer among Hardware and working framework. In conventional, centralized server began supporting numerous clients utilizing virtual machines. The virtualization shapes the establishment of Grid innovation. Utilizing virtualization, clients can get to workers or capacity without knowing explicit worker or capacity subtleties.

3. Different types of virtualization techniques

Virtualization procedures are utilized to produce various disengaged allotments on a solitary physical work and a few strategies fluctuate in the Virtualization arrangements techniques and deliberation. By and large, in gear virtualization, the host machine is the real machine on which the virtualization occurs, and the guest machine is the virtual machine [6]. The words host and guest are acquainted with perceiving the item that abrupt spikes popular for the physical machine from the item those unexpected spikes sought after for the virtual machine. The item or firmware that makes a virtual machine on the host gear is known as a Virtual Machine Manager the product that sudden spikes in demand for the virtual machine. The product or firmware that makes a virtual machine on the host equipment is known as a Virtual Machine Manager.

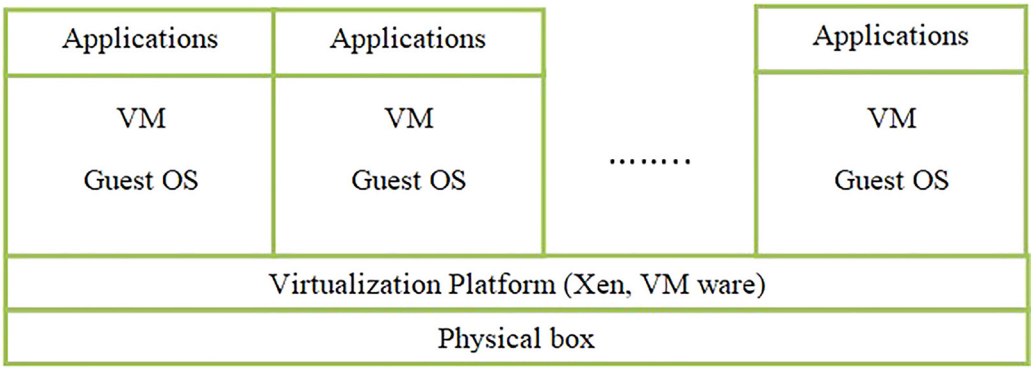

Fig. 1. Virtualization Architecture.

3.1. Full virtualization

It completely virtualizes the fundamental physical worker to help applications and programming to work likewise. The full virtualization strategy empowers the worker to run unaltered. The upsides of this technique are contributing successfully to manage down the working costs occupied with fixing and upgrading more established frameworks. It additionally lessens the force by diminishing the physical space.

3.2. Half virtualization

It incorporates address space virtualization, each virtual machine recreates the occurrences which fundamental equipment climate especially addresses space, and virtual machine comprises of a free location space. It requires address migration and introduced in most handy instances of virtualization.

3.3. Para virtualization

It is one of the adjusted renditions of working frameworks. Its fundamental favourable position is to permit many working frameworks on a solitary worker. Here in hardware virtualization, the host machine is the veritable machine on which the virtualization occurs, and the guest machine is the virtual machine [7]. The words host and guest are acclimated with perceiving the item that unexpected spikes sought after for the physical machine from the item those abrupt spikes popular for the virtual machine. The item or firmware that makes a virtual machine on the host gear is known as a Virtual Machine Manager.

3.4. Hardware virtualization

It is one of the ways to deal with improve the overall capability of virtualization. To offer assistance for virtualization in hardware CPU is incorporated and other gear fragments used to improve the show of a guest atmosphere. Gear virtualization fuses autonomic figuring, considering the evident activity the total atmosphere is regulated without any other person. The central objective of virtualization is to join legitimate tasks while improving adaptability and for the most part hardware resource utilization. A couple of working systems can be run in equivalent on alone central taking care of unit (CPU) with virtualization. The parallelism is used to diminish the overhead costs and shifts from playing out various undertakings, which remembers running a couple of ventures for a comparable Operating system. By using virtualization, an undertaking can all the more promptly supervise updates and snappy changes to the working system. Finally, virtualization altogether improves the capability and openness of resources and applications in an affiliation. Hardware virtualization contrasts from gear duplicating. In gear virtualization, a hypervisor mimics a particular piece of PC hardware yet in gear replicating, a touch of hardware copies another.

3.5. Work region virtualization

Work region virtualization is one of the thoughts of confining the shrewd work region from the physical machine. An instance of work region virtualization is a virtual work zone establishment (VDI), it is considered as another kind of gear virtualization. As opposed to helping out a host PC, it interfaces clearly through a comfort, mouse, and screen; the customer speaks with the host PC using another work station or mobile phone by techniques for an association relationship, for instance, a LAN, Wireless LAN, or even the Internet. If the affiliations continue to virtualized and join their worker ranch atmosphere, by then client plans in like manner need to grow to achieve the consistency, movement, and nature of organization passed on by their consolidated system.

3.6. Meeting virtualization

Meeting virtualization is a pattern of allowing various customers to interface and log use it at the same time into a typical stunning PC over the association. In the work territory, nuances appeared and records will be taken care of in an individual envelope. In the multisite arrangement, the gathering virtualization can be developed using a singular PC with various screens' consoles affiliations.

4. Improved genetic algorithm

The hereditary calculation with versatile boundaries is a promising variation of hereditary calculations. It is an arbitrary looking through a technique that has better improvement capacity. It additionally gives inner verifiable parallelism [8]. The level of the arrangement is significantly controlled by the Probabilities of Crossover (PC) and Probabilities of Mutation (PM). It additionally gets the upgraded looking through space and changes the looking through bearing naturally through the enhancement strategy for likelihood. With different preferences for improved hereditary calculation, this examination presents a fair booking system of VM assets in Grid Networking. By breaking down and thinking about the present statutes and recorded information, this technique will figure ahead of time its impact, in general, the framework. The change network portrayal is utilized in the improvement of a versatile hereditary calculation so the determination cycle doesn't need any outer info boundary. The assembly speed likewise relies upon the PC and PM. Contingent on the wellness esteem the change of pc and pm was made. A portion of the mathematical instances of backpack issues shows that this new versatile hereditary calculation, called MOGA (Mutation Only Genetic Algorithm) is better than numerous conventional strategies. The thought behind MOGA is that change likelihood is an element of time, wellness positioning of the chromosome, and locus. The reliance is naturally controlled by the insights of the locus and the wellness positioning of the chromosome at the given time. In the customary hereditary calculation, all the change probabilities are fixed; it is a unique instance of MOGA. In this paper, we expand the essential thought behind MOGA to incorporate the hybrid cycle (Fig. 2).

4.1. System architecture

In this part, a fundamental issue in Virtual Machine (VM) resources the board was watched out for that how to reasonably profile physical resource utilization of individual Virtual Machines. The crucial focus is to remove the use of physical resources by a VM across time, where the resources make sure to use for CPU cycles and use in-memory size. It is basic to use the Virtual Machine considering the way that the resource information is flighty about the essentialness of an autonomic resource the heads that depend on the model [9]. The resource the board is completely established on resource arranging across virtual machines. The designing of the Grid is referred to with respect to "layers", which give a specific limit. With everything taken into account, the higher layers are focussed on the customer and the lower layers are focussed on PCs and associations. At last, the base layer is the association that ensures the accessibility of the resources in the Grid. On its head, the resource layer lies and contained the certifiable resources that are significant for the Grid, for instance, PCs, storing structures, electronic data records, and even sensors, for instance, telescopes or instruments, which can be related directly to the

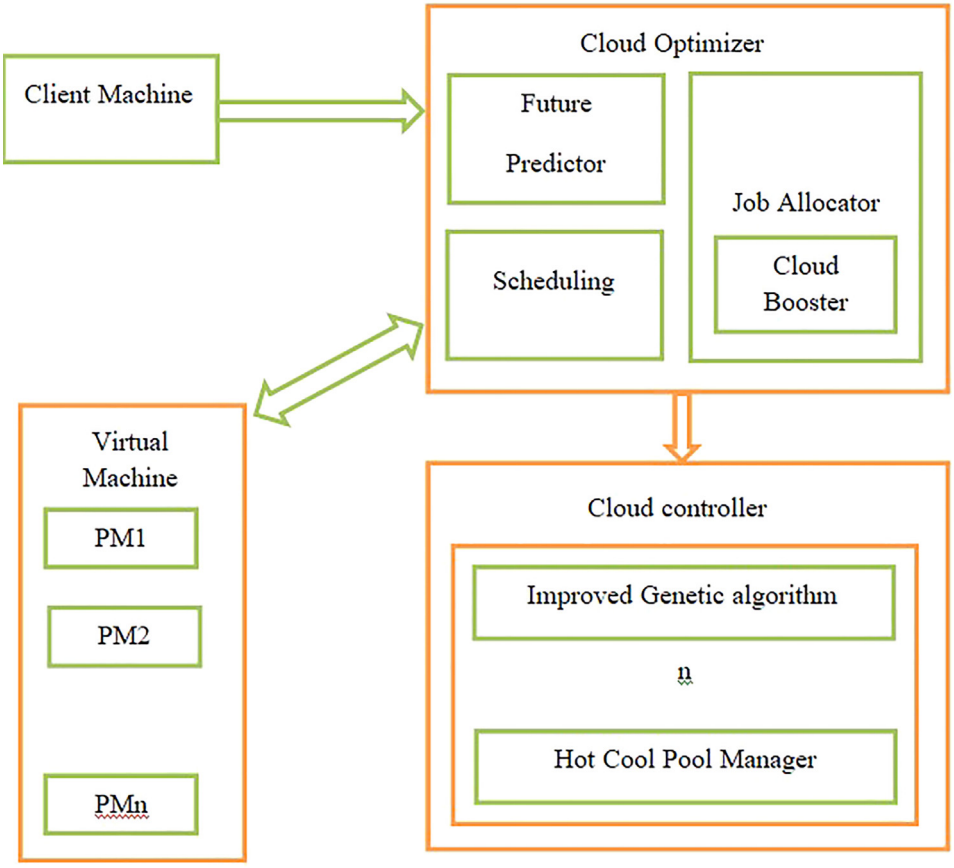

Fig. 2. System Architecture.

association. The figure shows the design to the arrangement, make, and realize the proposed estimation for VM task to reduce the pain point centers in the cloud. After this cycle needs to design and execute the directing of the over-troubling cycle to Migrate VM from over-trouble PM to vanquish over-troubling. On the client-side, the customer can introduce a task to the Cloud Controller, which courses the occupation among each and every equivalent center point. The occupation allocator is at risk for all the positions dispensing to the equivalent centers. While distributing the occupation it must consider the results from the future marker after the task of errands to cloud centers. After all the cycle the introduction of the occupation will be done uninhibitedly in cloud center points. At the last stage, the result transported off the cloud controller. In this Architecture, a future marker needs to foresee the future resource needs of VMs. For a change of VM is somewhat unpredictable considering the way that the application-level estimations by parsing logs of imminent requesting present inside the VM. As opposed to making a desire, the past external acts of VM is to be made. Considering the result, the given positions will be passed on among the fogs.

This proposed approach makes load changing decision reliant on center point stacks that change on each machine that flows the pile on the center points at run time. Current equivalent enlisting gear demands logically explicit respect for the nuances of booking and weight changing. It is furthermore a better path than manage dealing with the issue of Virtual Machine resource movement in a Grid Networking atmosphere. While existing Virtual Machine resources are disseminated to each physical center point and will pick the sending that will have an insignificant weight on the system. This paper depicts in detail the inherited computation based VM resource migration philosophy that centers just around system load changing. The estimation measures early, that it will have on the system after the new VM resource is passed on in the structure, by utilizing recorded data and the current status of the structure. It by then gets the plan, which will have an insignificant effect on the system. To ensures better burden changing and diminishes the amount of dynamic VM movements.

4.2. Job allocation

Client propose job requests to the Grid Controller. It will allocate the job among virtual machines. Then the Job will be executed on the different Physical Machines and it is shown in Fig. 3.

The designation depends on the outcomes from the future indicator and after the allotment, the positions made autonomously by Physical Machines. At last, the outcomes will be shipped off back to

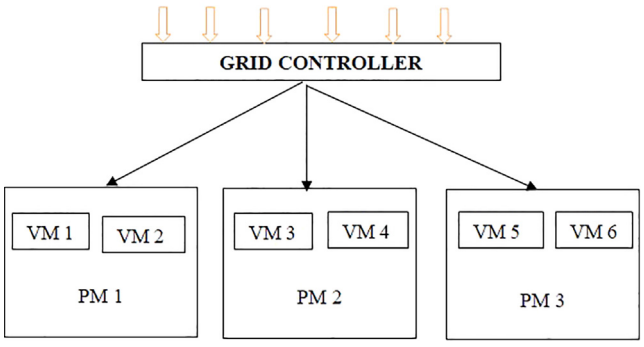

Fig. 3. Job allocation.

the matrix regulator if there is a fruitful finishing at work. The future indicator will foresee the future asset needs of VMs dependent on past insights. See Fig. 3. The arrangement is to glimpse inside a VM for application-level insights that are parsing logs of forthcoming solicitations. Doing so requires a change of the Virtual Machine which may not generally be conceivable as opposed to making our expectations dependent on the past outer practices of VMs. It quantifies the heap on consistently and predicts the heap in the following moment. The given positions were appropriated to the Grid assets dependent on the expected result.

5. Design goals

The design goals are follows,

Throughput: To minimize the total work done in a given period of time.
Turnaround: Minimize total time required for a single complex task, such as a batch job.
Response Time: It is defined as the total time required for a single task such as a command should be minimized.
Resource Utility: Minimize the usage of other resources such as processor time and storage.
Fairness: Service providing based on the task with priority this may be established by external policies.
Adaptability: Based on the resource allocation the process must dynamically and efficiently adapt to changes in the demand from tasks.
Consistency: To minimize the resource use and to ensure consistent service over a set of tasks.
Scalability: The number of resources in the Grid and the number of jobs that the Grid hosts must be scalable. This means that the resources consumed per machine in order to achieve a given performance objective must increase sub linearly with both. Based on the number of machines and the number of jobs the performance objective must increase sub linearly in resources consumed per machine.

For an efficient incoming set of tasks using huge data's, the system need to fits well in a grid. In this research the work starts one virtual machine (VM) in view of the intricacy of subtasks; further VMs are distributed and de-designated. Due to the changes it presents an inert line between the Job Manager and the Task Manager (VM). In the event that VM gets inactive, at that point it frames a line. In the event that the employment moves toward the Job chief, it brings the primary VM in the line and designates the occupation to it. The general throughput of the information handling framework is improved and reduces the communication overhead. Adaptable asset provisioning and movement of machine state gives an improved effectiveness of asset use.

The assignment relies upon the results from the future pointer and after the apportioning, the positions made self ruling by Physical Machines. Finally, the results will be sent off back to the grid controller if there is a productive completion at work. The future marker will predict the future resource needs of VMs subject to past experiences. See Fig. 4. The game plan is to look inside a VM for application-level bits of knowledge that are parsing logs of imminent requesting. Doing so requires a difference in the Virtual Machine which may not commonly be possible instead of making our desires reliant on the past external acts of VMs. It evaluates the stack reliably and predicts the load in the accompanying second. The given positions were appropriated to the Grid resources subject to the normal outcome.

For a proficient approaching arrangement of assignments utilizing enormous data, the framework needs to fits well in a network.

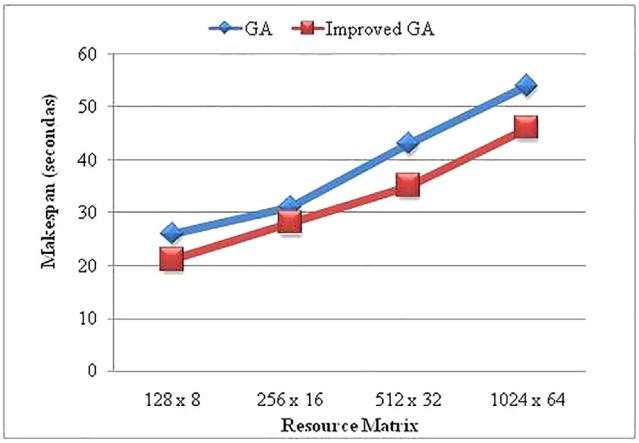

Fig. 4. Comparison for make-span.

In this examination the work begins with one virtual machine (VM) taking into account the multifaceted nature of subtasks; further VMs are disseminated and de-assigned. Because of the progressions, it presents a dormant line between the Job Manager and the Task Manager. If VM gets latent, by then it outlines a line. If the work pushes toward the Job boss, it brings the essential VM in the line and assigns the occupation to it. The overall throughput of the data taking care of the system is improved and lessens the correspondence overhead. Versatile resource provisioning and development of machine state gives improved viability of resource use.

5.1. Grid Booster technique

This calculation has two stages. The first stage is utilized for computing the weight of the hub in the equal handling framework. The second stage is utilized for assessing the position extent needed by each equal machine dependent on weight measurements.

Documentations:

WCPU – CPU Weight.
WMem – Memory Weight.
WLoad – Weight for current burden. Resource factors for calculating weight for a particular machine is tabulated as follows: (Table 1).

5.2. Node weight calculation procedure

i. Gather asset data from each equal hub.
ii. Compute Weight for every asset independently by barring negative components.
iii. In the wake of assessing WCPU, WMem, and WLoad, decide the heaviness of a hub by utilizing the equation.

$$\text{Weight of Machine} = \text{WCPU} + \text{WMem} - -\text{Wload}$$

Table 1
Resource Information.

Resource	Sample Value for a Machine	Weight Constant
CPU Speed (GHZ)	2.4	0.5
Main Memory (GB)	2	0.4
CPU Utilization	0.013	0.5
Memory Utilization	0.536	0.4

Table 2
Resource Information.

	P1	1				9		10
	P2	3		6				
	p3	4	5	7				
	p4	2				8		
Time	0	1	2	3	4	5	6	7

5.3. Job allocation method

i. Figure the necessary occupation Percentage for a machine utilizing the recipe.

For Node X, Wx = (Wx/∑Wi.).

Where ∑ Wi is the complete load of the apparent multitude of hubs.

ii. Separation the occupation dependent on the work extent and disperse the employment among equal hubs. Accordingly, Job (load) is put together by the client and it is conveyed among equal hubs in an equal handling framework, and yield is produced.

- Sort these successions on rank base, select first n groupings with min

$$\text{weight}.R = \sum_{i=1}^{N_j} \frac{\frac{W_i}{Max(W_i)}}{N_j} + \sum_{i=1}^{N_j} \frac{\frac{Max(P_i)-P_i}{Max(P_i)}}{N_{vm}} + \sum_{i=1}^{N_j} \frac{\frac{T_i}{Max(T_i)}}{N_{vm}} + \sum_{i=1}^{N_j} \frac{\frac{Min(C_i)-C_i}{Max(C_i)}}{N_j}$$

R – Rank
N_j – Number of jobs
W_i – Weighting time of *i*th Job
P_i – Communication Cost of *i*th Job
N_{vm} – Number of Jobs
T_i – Idle time *i*th VM
C_i – Migration cost of *i*th Job

Consider Following Allocation
Table 2

Holding up Time of Job 9: 6
Here most noteworthy holding up time 7 (work 10)
PW (Percentage of holding up time) = Sum (hanging tight an ideal opportunity for each work)/(most noteworthy holding up time * number of occupations)
Inactive season of p4 = 1, Highest inert time = 4
PI (Percentage of inactive time) = Sum (inert time for every hub)/(most noteworthy inert * number of hubs)
PP (level of correspondence cost) = Sum (correspondence cost of every hub)/(Maximum correspondence cost * number of occupations).
PP (level of replication cost) = Sum (replication cost of every hub)/(Maximum replication cost * number of occupations).
Rank = PC + PP + PW + PI

Here we select progression with the least position. Thusly, the AGA (Acclimatize Genetic Algorithm) makes a new progression. During this age, AGA limits rank. Here the position is the total of holding up time, correspondence cost, replication cost, and dormant time. So these ages limit the above components.

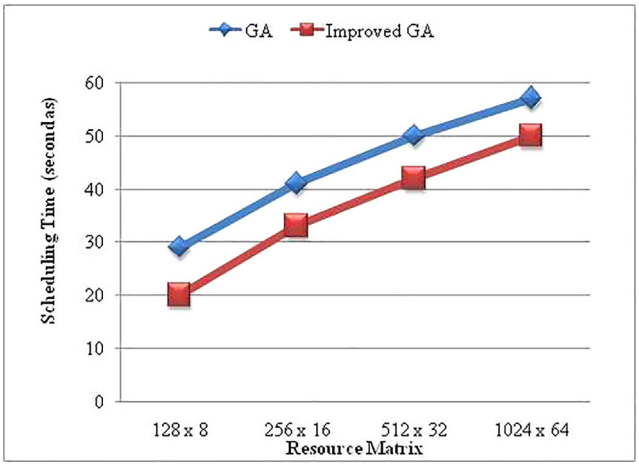

Fig. 5. Comparison for scheduling time.

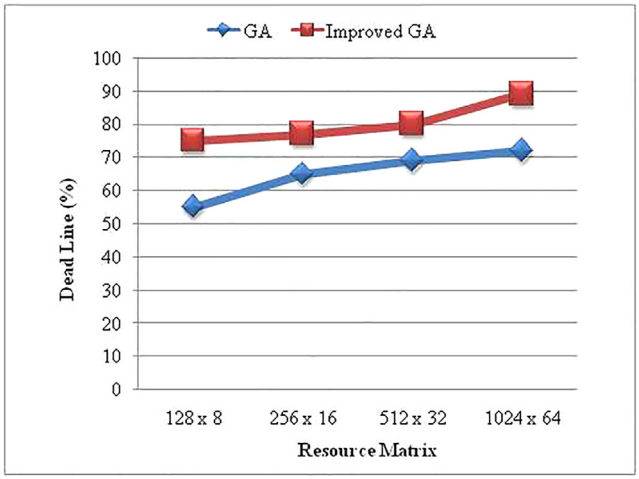

Fig. 6. Comparison for dead line.

Table 3
The comparison of scheduling algorithm using make-span, scheduling time, deadline and resource utilization.

Algorithms	Resource Matrix	Make-span (Seconds)	Scheduling Time (Seconds)	Deadline hit (%)	Resource utilization (%)
GA	128 × 8	27	29	56	23
Improved GA		22	21	76	32
GA	256 × 16	32	42	66	26
Improved GA		29	34	78	35
GA	512 × 32	44	51	69	27
Improved GA		36	43	81	37
GA	1024 × 64	55	58	73	29
Improved GA		47	51	89	39

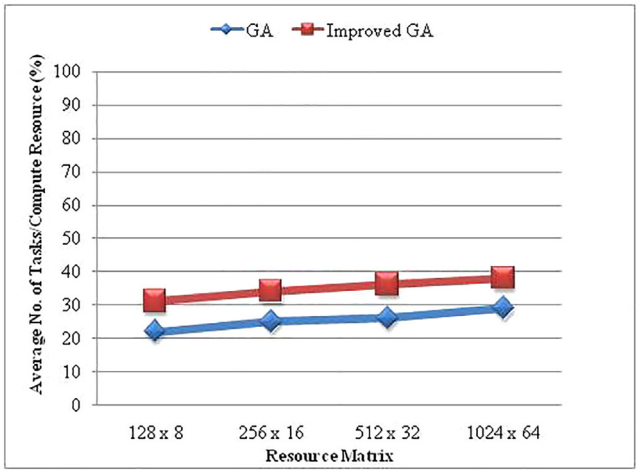

Fig. 7. Comparison for resource utilization.

6. Experimental results

An Improved calculation is a tally that converges, regardless, two particular calculations that manage a similar issue, either picking one or exchanging between them through the scope of the assessment. This is commonly done to join required highlights of each, with the target that the general figuring is superior to the individual parts. Here, for network systems association to be proficient, the individual authorities that make up the worker ranch structure should be utilized ideally. Truth be told, even an inert worker eats up about a tremendous fragment of its most over the top force. With the improvement of structure masterminding in the past generally scarcely any years, map decline has seen the goliath unforeseen development, particularly for gigantic expansion information real figuring. In this, the issue of occupation development for get-togethers of focus focuses that run the framework. Thusly, work for the use of structure relocation based reconfiguration of focuses, a structure that extends and improves network sum runtime system to enough arrangement with complex information appraisal assignments with advancement utilizing improved Genetic Algorithm. This piece presents the assessment of evaluation and test intends to test the proposed figuring of the association's environment. From the beginning, starters depended upon the going with limits; for example, make range, booking time, cutoff time hit, and asset use. Each errand in the work cycle has information and yield reports of moving sizes.

Table 3 shows the comparison of scheduling algorithm using parameters like make-span, scheduling time, deadline hit and resource utilization. The resource matrix used for different measure then calculates the values.

Fig. 4 shows the comparison of make-span for GA, Improved GA and hybrid scheduling. The proposed method has less make-span than other preceding techniques.

Fig. 5 shows the comparison of scheduling time for GA, Improved GA and Hybrid GA. The proposed method of hybrid scheduling has less scheduling time when compare with GA and Improved GA.

The comparison for deadline is shows in Fig. 6. The proposed hybrid scheduling has high value of deadline hit rate when compare with GA and Improved GA.

The comparison for resource utilization is shows in Fig. 7. The proposed method of hybrid scheduling has balanced the load for each resource when compare with GA and Improved GA. The proposed method performs better function and gives accurate result.

7. Conclusion

This paper presents the design of a improved scheduling algorithm for scheduling the workflow. The proposed method will process each request and allocate the resource and balance the workload efficiently. To prove the efficiency and reliability of the proposed method, an implementation and experiments have been carried out and compared against normal genetic algorithm. The results obtained by proposed optimizer are shown in experimental results section of this paper. The result table shows the resource matrices for levels starting from (128×8), (256×16), (512×32), (1024×64) and its mark-span. It shows better than genetic algorithms, as well as scheduling time and deadline hit also accurate and reduced. It also can observe the improved genetic algorithms take the highest resource utilization compared with GA.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] G. Banavar, J. Beck, E. Gluzberg, J. Munson, J. Sussman, D. Zukowski, Challenges: an application model for pervasive computing, in: MobiCom '00: Proceedings of the 6th annual international conference on Mobile computing and networking, New York, NY, USA, 2000, pp. 266–274, ACM Press.

[2] M. Migliardi, M. Maheswaran, B. Maniymaran, P. Card, F. Azzedin, Mobile interfaces too computational, data, and service grid systems, Mob. Comput. Commun. Rev. 6 (4) (2002) 71–73.

[3] D.S. Litke, T. Varvarigou, Mobile grid computing: Changes and challenges of resource management in a mobile grid environment, Proceedings of Practical Aspects of Knowledge Management (PAKM 2004), 2005.

[4] M.G. Ball, B. Qela, S. Wesolkowski, A review of the use of computational intelligence in the design of military surveillance networks, in: Recent Advances in Computational Intelligence in Defense and Security, Springer International Publishing, 2016, pp. 663–693.

[5] G. Han, A. Qian, J. Jiang, N. Sun, L. Liu, A grid-based joint routing and charging algorithm for industrial wireless rechargeable sensor networks, Comput. Netw. 101 (2016) 19–28.

[6] V. Di Martino, M. Mililotti, Sub optimal scheduling in a grid using genetic algorithms, Parallel Comput. 30 (5) (2004) 553–565.

[7] Y. Gao, H. Rong, J.Z. Huang, Adaptive grid job scheduling with genetic algorithms, Future Gener. Comput. Syst. 21 (1) (2005) 151–161.

[8] J. Kołodziej, S.U. Khan, L. Wang, A.Y. Zomaya, Energy efficient genetic-based schedulers in computational grids, Concurr. Comput.: Pract. Exp. 27 (4) (2015) 809–829.

[9] Shi, J.J. Dongarra, Scheduling workflow applications on processors with different capabilities, Future Gener. Comput. Syst. 22 (6) (2006) 665–675.