A Novel Dynamic Optimization Technique for Finding Optimal Trust Weights in Cloud

Aluri V.H. Sai Prasad [1]* and Ganapavarapu V.S. Rajkumar[2]

[1]Associate Professor, Department of Computer Science and Engineering, Neil Gogte Institute of Technology
Hyderabad, Telangana, 500088, India
[e-mail: sai.alurip@gmail.com]

[2] Professor, Department of Computer science and Engineering, GITAM Institute of Technology,
GITAM (Deemed to be University), Visakhapatnam, Andhra Pradesh, 530045, India
[e-mail: gganapav@gitam.edu]

*Corresponding author: Aluri V.H. Sai Prasad

Received October 14, 2020; revised November 23, 2021; accepted February 6, 2022; published June 30, 2022

Abstract

Cloud Computing permits users to access vast amounts of services of computing power in a virtualized environment. Providing secure services is essential. There are several problems to real-world optimization that are dynamic which means they tend to change over time. For these types of issues, the goal is not always to identify one optimum but to keep continuously adapting to the solution according to the change in the environment. The problem of scheduling in Cloud where new tasks keep coming over time is unique in terms of dynamic optimization problems. Until now, there has been a large majority of research made on the application of various Evolutionary Algorithms (EAs) to address the issues of dynamic optimization, with the focus on the maintenance of population diversity to ensure the flexibility for adapting to the changes in the environment. Generally, trust refers to the confidence or assurance in a set of entities that assure the security of data. In this work, a dynamic optimization technique is proposed to find an optimal trust weights in cloud during scheduling.

Keywords**:** Cloud Computing, Cluster based dynamic optimization techniques, Dynamic programming, Evolutionary Algorithms, Task scheduling, Meta heuristic algorithms and Trust.

http://doi.org/10.3837/tiis.2022.06.015 **ISSN : 1976-7277**

1. Introduction

A collection of several interconnected computers consisting of several united computing resources is called the Cloud. Recently, cloud computing has seen a lot of advancement that has helped in stimulate the arrangement of several data centres that are inter-connected and geographically dispersed to offer dependable services of the best quality [1]. Today, cloud computing has brought about conformity and a major change in IT as a business. The development of its application along with its promotion has opened several doors and has also confronted various challenges in advancing traditional IT services.

In recent times, cloud computing has risen to be another model based on the Internet to empower clients. It can organize access to several shared pools consisting of configurable assets and these are providing and further discharged immediately with minimal administration or the cooperation of the provider [2].

Any scheduling in the cloud will be the reason behind the choice of well-suited resources to execute tasks. It will implement by taking specific dynamic and static parameters or some restrictions into consideration. Any scheduling algorithm will be applying used for planning a task that has an evaluated gain or benefit for the execution of the task. But, the ability of computing within a distributed system can shift from different resources to the resource cost used.

Scheduling is the major challenge in cloud computing which can allocate tasks to available resources based on task qualities, metrics and requisites without affecting the services provided by the cloud. Allocation of resources is decided by scheduling in cloud computing such as CPU, memory, secondary storage space I/O, network bandwidth and so on between tasks and users. Scheduling strategy is adapted by a good scheduler based on changing environment over time, the nature and relative metrics of tasks involved in batch of tasks submitted for processing [3].

A Min-Min algorithm will begin with a set of unassigned tasks. The task identified first is the Minimum Completion Time (MCT). Once it completes, among all the minimum completion times, there is a minimum value that is selected, and this refers to the minimum time that is surrounded by the tasks on the resources. Once this is done, based on the minimum time, a task may be scheduled, which will be allocating on the corresponding machine. This process continues until all the tasks are assigning to their respective resources [4]. The task having a Maximum Execution Time (MET) is selecting for the execution. During the selection of the resources consuming the MCT, it will refer to the choice of a resource which is the slowest among all other resources available. Also, it can further achieve a makespan and this is the shortest one for tasks submitted on different resources other than concurrency [5].

Dynamic environment-based optimization can be very challenging but also an important one as several real-world problems keep changing with time. Swarm intelligence and evolutionary computation will be ideal tools for addressing optimization problems observed in dynamic environments because their inspiration comes from certain self-organized systems or biological evaluation. These have always been changing in terms of their environment. An evolutionary optimization found in dynamic environments, also known as Evolutionary Dynamic optimization (EDO), today attracts plenty of research in the last twenty years. It has become a very active area of research.

Dynamic programming has to be identifying as a design paradigm with the idea of dividing a problem into several sub-problems of a similar type. Today, it has been identifying as effective in several single-objective and multi-objective problems of optimization.

For this work, an optimal weight is identifying by a novel cluster based dynamic optimization techniques. The rest of the investigation is planning as follows: Section 2 discusses all related work in the literature and the different methods employed in Section 3. The outcomes are addressing in Section 4, and the conclusion is in Section 5.

2. Literature Survey

Jethava and Desai [6] had been dealing with various algorithms of scheduling, along with making a comparative study of the algorithm of task scheduling. Such algorithms will be used by providers of cloud service and different researchers for dynamic as well as static tasks of these applications. The focus of task scheduling is the efficient resource utilization and reduction of the time taken for task compilation. Several techniques are employed to solve the problem of task scheduling. The option is to propose flow for the optimization of multi-objective-based workflow which is dynamic using two different scheduling algorithms: The Black Hole and the Ant Colony Optimization algorithms.

Cloud Computing is its ability to dynamically schedule an application based on the best resources available in the load. A task schedule on the environment of the cloud was proposed by Ibrahim et al [7]. The proposed algorithm and its principle included allocating incoming tasks to the best possible resources while running specific tasks that are based on measuring the situation of every resource. It is implementing in connection with the actual level of availability, cost, processing power and finally, the actual number of tasks. For the proposed algorithm and its performance, a new comparative study is taking between this algorithm, the Round Ribbon (RR) algorithm, and the Minimum Completion Time (MCT) algorithm. Experimental results proved the proposed algorithm to have outperformed the MCT and the RR by reducing cost and make-span of the tasks.

The problem of dynamic energy management which was in pursuit of energy efficiently in the cloud data centres, was investigated. More specifically, the model of a dynamic energy management system for the data centres of the cloud also built. It consisted of the Task Scheduling Module, Load Balancing Module, and the DVS Management Module. By the Task Scheduling Module, the process of scheduling gets analysed by the Stochastic Petri Net, in which a method of task-oriented resource allocation (LET-ACO) has proposed as in Pang et al [8]. It can optimize the system running time and the consumption of energy by the tasks of scheduling. The studies of simulation have confirmed the proposed model's effectiveness and the results of the simulation observed. This proved that in comparison to the ACO, the Min-Min and the RR scheduling, the LET-AC saved about 28% and 31% and 40

Dashti and Rahmani [9] had proposed a new hieratical architecture that could satisfy the needs of the producers and consumers in terms of technology. From the perspective of the providers, any incompatibility observed between the physical machine specification and user requests in the cloud can result in issues such as trade-off and energy performance. To ensure the user tasks of Quality of service are maintained, and the efficiency of energy has obtained, a modified Particle Swarm Optimisation was proposed for reallocating the migrated virtual machines. There was also a dynamic consolidation of all under-loaded hosts providing power saving. The simulation results in the CloudSim demonstrated the condition of simulation that is close to its real environment and this method could save about 14% more energy. It also could reduce migrations and simulation time compared to other methods.

Cloud Computing has various issues such as the provision of hardware and software, data security, scheduling of virtual machines and their management. The primary task, however, is Load Balancing. Padmavathi and Basha [10] had proposed a new and innovative Elasticity Algorithm for performing load balancing using the ACO among the other systems that were currently present in data centres.

3. Methodology

The Details of trust computation and proposed Dynamic Herding Optimization has been presented in this section.

3.1 Trust Computation

An estimate of the trust value can be made by making use of several constraints over the applied security features. An estimation based on trust was formulated to evaluate trust value. There was also a list for this that was made and the trust value if affected based on user experience, their transactions were also observed for a certain period of time. There was yet another refined set of such parameters that were formulated for dynamically evaluating trust. Both the static and dynamic trusts determined the security of cloud services. Such a trust model will act in accordance with the security and its strength with a ranking service for the cloud applications kept in place [11]. Trusts are of two types, direct and indirect. The direct trust will be dependent on the user impression that is based on the quality of service of that of its CSP, and indirect trust, on the other hand, is the aggregated impression of previous consumers using services. This is also known as reputation trust. There was a degree level of such trust that had been characterized by the star rating models, which were signified as follows: very untrustworthy, untrustworthy, neutral, trustworthy and very trustworthy.

In case we make an assumption that the user u_a prefers to use the services of cloud from CSP X, there is a need to compute trust value for X and identify if X is to be trusted. If the Imp (a, b) denotes the impression of b on a, the trust value of that of X for u_a can be calculated as given below in (1):

$$\mathrm{Imp}(X,U_a) = C1 \sum_{u_i\,\text{€}U} \alpha_i \mathrm{Imp}(X, u_i) + C2\sum_{Y}\beta_i \mathrm{Imp}(X,\ Y) \qquad (1)$$

Where, Imp (X, u_i) refers to the impression value of u_i that is on the CSP X, Imp (X, Y) is the impression of all other CSPs that are on X, and lastly, C_1, C_2, α_i, β_i denote coefficients [12]. Among the current models, an evaluation was performed with one node and this was found to be inappropriate as in (2).

$$\text{Direct Trust (DT)} = w_1 * A + w_2 * DI + w_3 * C \qquad (2)$$

Where, w_1, w_2, w_3 are weighted coefficients. A, I and C indicate the values of trust parameters in the cloud which are its authenticity, confidentiality, and Data integrity as according to (3 and 4).

Assuming that $w_1+w_2+w_3 =1$.

$$\text{Indirect Trust (IDT)} = \text{Trusted Third Party (TTP)} \quad (3)$$

$$\text{Trust (Final)} = w4 * DT + w5 * IDT \quad (4)$$

Assuming that $w_4+w_5 = 1$ (the weighted coefficients also called positive weights).

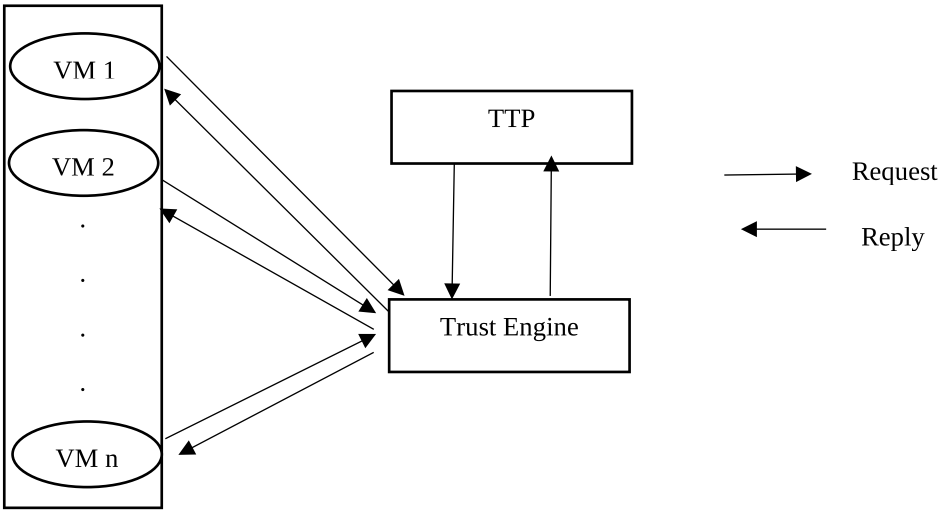

Fig. 1. Trust Evaluation Model

There is also another final trust value computed for every machine that is later incorporated with in Trust Min-Min and Trust Max-Min consisting of time of completion and time of execution to support integrity and security for implementing the new trust model.

(a) Initially, the service consumers are looking for a cloud service with the good feedback. (b) Secondly, the service users will use this cloud service and give their opinion or feedback. (c) Next, cloud providers or trusted third parties take the users feedback and assess their cloud services using some tools or models. (d) Finally, they preserve or save this feedback into their cloud provider database or/and into the trusted third-party database which will be available for other cloud service users.

3.2 Proposed Trust-based Particle Swarm optimization (PSO) scheduling

Kennedy and Eberhart (1995) proposed the PSO, which was a very significant invention based on the birds and their group behaviour. There are stochastic methods that are used by the PSO for the generation of a feasible population to accelerate the speed of convergence. Furthermore, individuals shifted to various regions, based on their fitness levels. The fitness of all these particles computed using the PSO, and this was iterative. Accordingly, the particles have by tracking their individual and optimal P_i along with their global minimum P_g [13, 14]. To schedule a job for the cloud, if there are N_p particles, the search space of the particle is $D=m \times n$, wherein the actual set of tasks are called the m and the actual resources are the n. The position of the i^{th} particle will be $X_i = (X_{i11} \ldots X_{imn})$ and its speed is

$V_i = (V_{i11} \ldots V_{imn})$. From among these, the $X_{iab} \in \{0,1\}$ means the i^{th} particle has been assigned the a^{th} job for the b^{th} resources. Thus, the i^{th} particle along with its optimal position is P_{iab} =

(P_{i11}…P_{imn}). The optimal global location is P_{gab}= (P_{g11}…P_{gmn}). The velocity of the updating formula is shown in equation (5):

$$V_{iab}^{new} = \omega V_{iab}+c_1r_1(P_{iab}-X_{iab}) + c_2r_2(P_{gab}-X_{iab}) \quad (5)$$

Wherein, ω is the final inertia weight, and r_1, r_2 are random numbers falling between (0,1), and finally, $c_{1,}$ c_2 are the learning factors.

3.3 Proposed Dynamic Herding Optimization

There is a herding algorithm based on simulating of the herding behaviour of different individuals using a Lagrangian model and the Genetic Operators [15]. Problems in scheduling and new tasks that arise over time can be a special case of dynamic optimization problems. The primary focus of the dynamic herd optimization was to achieve a secure level of scheduling that decreases cost and meets deadlines to choose trustworthy resources.

Herding is dependent on the individuals and their motion. There are two major goals to this: (i) increasing density and (ii) finding food. These are used as the objectives that result in the individual herding around the solution, which is global best. The individual will move to the best solution while looking for the highest density of food. That is, if the distance is closer to the higher food density, the lower is its objective function.

Herding has been inspired by the herding simulation of individuals in connection with environmental and biologic processes. x_i has a time-dependent position in the search space that is governed by three different actions which are: (i) the induced movement by other individuals, (ii) foraging activities and (iii) random diffusion as per equation (6):

$$x_i (t+ \Delta t) = x_i(t)+ \Delta t(N_i+F_i+D_i) \quad (6)$$

Where,

$$\Delta t=C_t\sum_{j=1}^{d}(UB_j -LB_{j)} \quad (7)$$

And N_i refers to the motion that is induced by that of the other individuals, and F_i refers to foraging motion. D_i denotes physical diffusion of that of the ith individual, d denotes the variables that need optimization. UB_j and LB_j refer to the upper bounds and the lower bounds of their jth variable. C_t is the constant number as in [0,2]. Individuals maintain higher density and move based on mutual effects. The induced motion in a direction that takes into consideration the individual i, and α_i, will be estimated from its local swarm density (a local effect), target swarm density (a target effect), and finally, repulsive swarm density (a repulsive effect). For individual i, the movement is as in equation (8) [16].

$$N_i^{i+1} = N^{max} * \alpha_i + \omega_n * N_i^t \quad (8)$$

Where,

$$\alpha_i = \alpha_i^{local} + \alpha_i^{target} \quad (9)$$

and N^{max} denote maximum induced speed, ω_n denote inertia weight for the induced motion as per [0,1], and N_i^{old} denotes the final motion induced.

Foraging motion will be formulated based on the location of food and the earlier experiences on the location of food that is expressed for their ith individual as per equation (10& 11):

$$F_i = V_f B_i + \omega_f F_i^{old} \tag{10}$$

Where,

$$B_i = B_i^{food} + B_i^{best} \tag{11}$$

V_f denotes the foraging speed, ω_f indicates inertia weight for the foraging motion as in [0,1], F_i^{old} denotes the final foraging motion, refers to the attractiveness of the food and B_i^{best} refers to the best effect and its fitness of the ith individual. Physical diffusion is a random process that is expressed as the maximum diffusion speed with a random vector as per equation (12):

$$D_i = D^{max} (1 - I / I_{max}) \delta \tag{12}$$

Wherein, D^{max} denotes the maximum speed of diffusion, I is the iteration and δ refer to the random vector in [−1,1].

The problems in scheduling have been shown in the form of a graph G = (N, E), in which the nodes N will indicate VMs and tasks and set of edges, E refers to the connection between that of the task and the VM. Every edge has a trust which is computed and this will need to be dynamically optimized based on the trust weights. The methods proposed will optimize the trust weights and utilization of resources dynamically. All the individuals will be placed at the beginning of the VMs randomly. At the time of iteration, the individuals will build certain solutions to the scheduling problems of the cloud. The iterations will be indexed by t, and $1< t<t_{max}$, wherein t_{max} will be the maximum iterations permitted.

4. Results and Discussion

Table 1 shows the parameters of Herding optimization algorithm. Below **Table 2 to 4** and **Fig. 1 to 3** shows the results for trust Min-Min, trust Max-Min, Average Schedule Length in second, Resource Utilization (%) and System Success Probability (%) respectively.

Table 1. Parameters of Proposed Algorithm

Parameter	**Value**
Number of Iterations	200
Inertia weights	0.9
Maximum Demands	0.005
Foraging Velocity	0.02

Table 2. Schedule Length (ms) for Proposed Herding Dynamic Optimized Scheduling

Number of Tasks	Trust Min-Min Algorithm	Trust Max-Min Algorithm	Proposed trust based PSO Scheduling Algorithm	Proposed Herding Dynamic Optimized Scheduling Algorithm
300	723	704	672	645
600	1596	1521	1462	1397
900	2339	2287	2177	2082
1200	3181	3035	2940	2846
1500	3964	3695	3522	3358

From the above **Table 2** it is observed that the proposed algorithms Trust based PSO Scheduling, Herding Dynamic Optimized Scheduling are significantly better with compared to existing Trust Min-Min and Trust Max-Min algorithms [5] from the perspective of schedule length. Hence, proposed algorithms yield optimal performance.

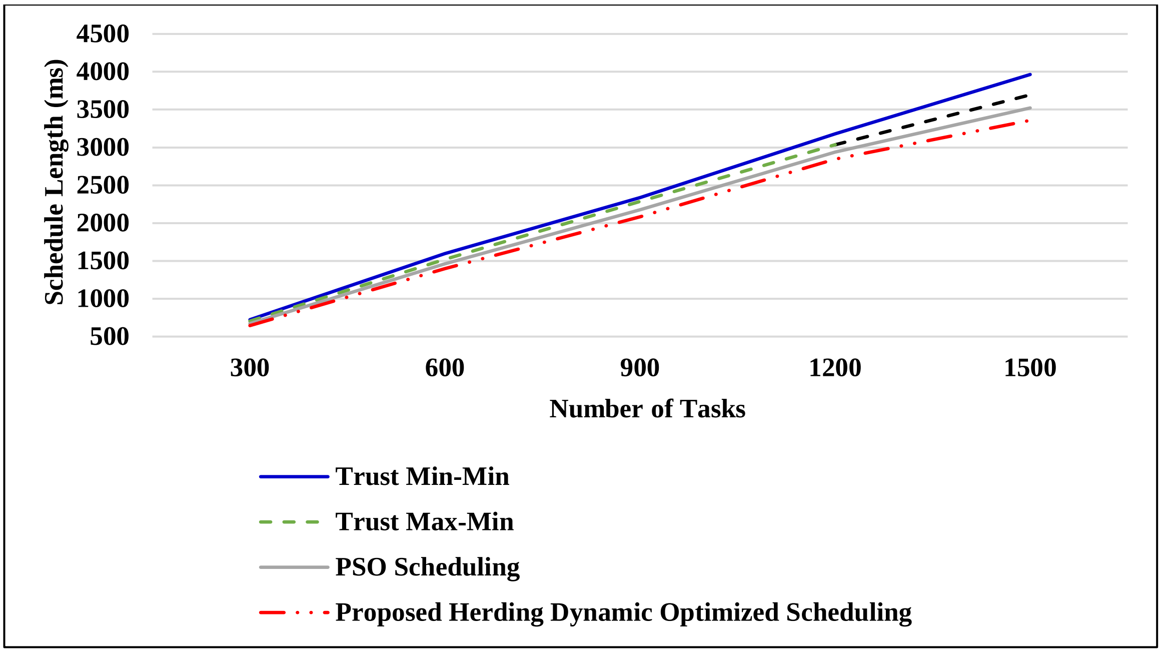

Fig. 2. Schedule Length in second for Proposed Herding Dynamic Optimized Scheduling

Table 3. Comparative Study of Trust Min-Min, Trust Max-Min, and Proposed trust based PSO, Herding Dynamic Optimized Scheduling algorithms on the basis of Schedule Length

No. of Tasks	Herding Dynamic Optimized Scheduling Algorithm Reduction over Trust Min-Min Algorithm	Herding Dynamic Optimized Scheduling Algorithm Reduction over Trust Max-Min Algorithm	Herding Dynamic Optimized Scheduling Algorithm Reduction over Trust based PSO scheduling Algorithm
300	10.7%	8.38%	4.01%
600	12.46%	8.15%	4.44%
900	10.98%	8.96%	4.36%
1200	10.53%	6.22%	3.19%
1500	15.28%	9.12%	4.65%

From the above Table 3. it can be observed that the proposed herding dynamic optimized scheduling has lower schedule length by 10.70%, 8.38% & 4.01% for 300 number of tasks and by 12.46%, 8.15% & 4.44% for 600 number of tasks and by 10.98%, 8.96% & 4.36% for 900 number of tasks, and by 10.53%, 6.22% & 3.19% for 1200 number of tasks and by 15.28%, 9.12% & 4.65% for 1500 number of tasks when compared with Trust Min-Min, Trust Max-Min and Trust based PSO scheduling respectively.

Table 4. Resource Utilization in (%) for Proposed Herding Dynamic Optimized Scheduling

Number of Tasks	Trust Min-Min Algorithm	Trust Max-Min Algorithm	Proposed trust based PSO Scheduling Algorithm	Proposed Herding Dynamic Optimized Scheduling Algorithm
300	85	87	89	92
600	84	86	88	90
900	87	89	92	94
1200	88	90	92	94
1500	85	87	89	91

From the above Table 4 it is observed that the proposed algorithms Trust based PSO Scheduling, Herding Dynamic Optimized Scheduling are significantly better with compared to existing Trust Min-Min and Trust Max-Min algorithms [5] from the perspective of system resource utilization. Hence, proposed algorithms yield optimal performance.

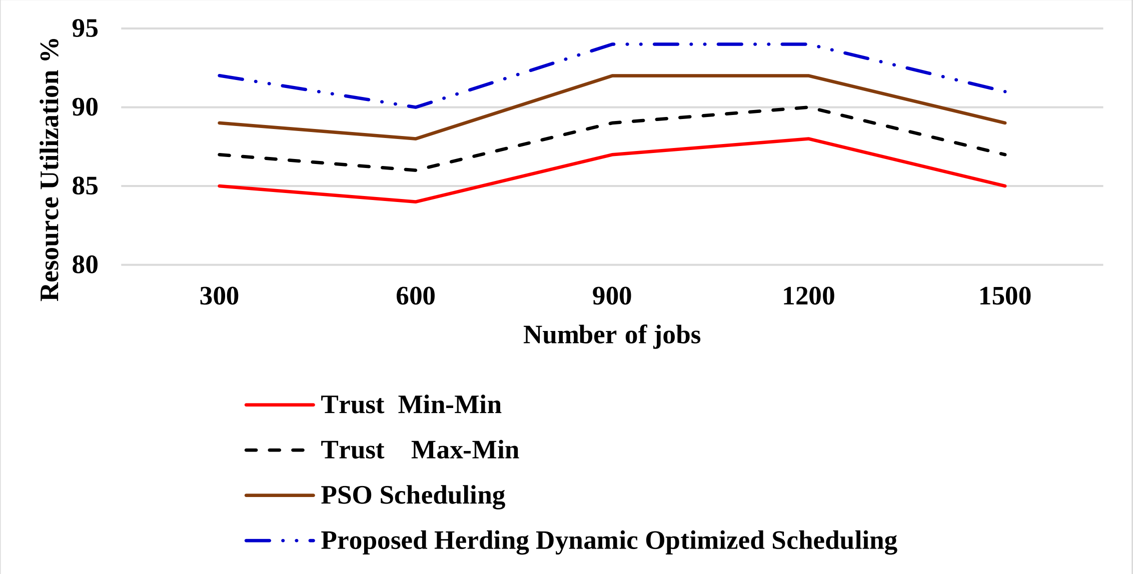

Fig. 3. Resource Utilization % for Proposed Herding Dynamic Optimized Scheduling

Table 5. Comparative Study of Trust Min-Min, Trust Max-Min, and Proposed trust based PSO, Herding Dynamic Optimized Scheduling algorithms on the basis of Resource Utilization

No. of Tasks	Herding Dynamic Optimized Scheduling Algorithm improvement over Trust Min-Min Algorithm	Herding Dynamic Optimized Scheduling Algorithm improvement over Trust Max-Min Algorithm	Herding Dynamic Optimized Scheduling Algorithm Improvement over Trust based PSO scheduling Algorithm
300	7.60%	5.43%	3.26%
600	6.66%	4.44%	2.22%
900	7.44%	5.31%	2.12%
1200	6.38%	4.25%	2.13%
1500	6.59%	4.39%	2.19%

From the above **Table 5**, it can be observed that the proposed herding dynamic optimized scheduling has higher resource utilization by 7.60%, 5.43% & 3.26% for 300 number of tasks and by 6.66%, 4.44% & 2.22% for 600 number of tasks and by 7.44%, 5.31% & 2.12% for 900 number of tasks and by 6.38%, 4.25% & 2.13% for 1200 number of tasks and by 6.59%, 4.39% & 2.19% for 1500 number of tasks when compared with Trust Min-Min, Trust Max-Min [5] and proposed Trust based PSO scheduling respectively.

Table 6. Success Probability for Proposed Herding Dynamic Optimized Scheduling

Number of Tasks	Trust Min-Min Algorithm	Trust Max-Min Algorithm	Proposed Trust based PSO Scheduling Algorithm	Proposed Herding Dynamic Optimized Scheduling Algorithm
300	0.83	0.85	0.86	0.88
600	0.78	0.81	0.82	0.84
900	0.73	0.77	0.78	0.80
1200	0.71	0.75	0.76	0.78
1500	0.69	0.74	0.75	0.77

From the above **Table 6** it is observed that the proposed algorithms Trust based PSO Scheduling, Herding Dynamic Optimized Scheduling are significantly better with compared to existing Trust Min-Min and Trust Max-Min algorithms [5] from the perspective of success probability. Hence, proposed algorithms yield optimal performance.

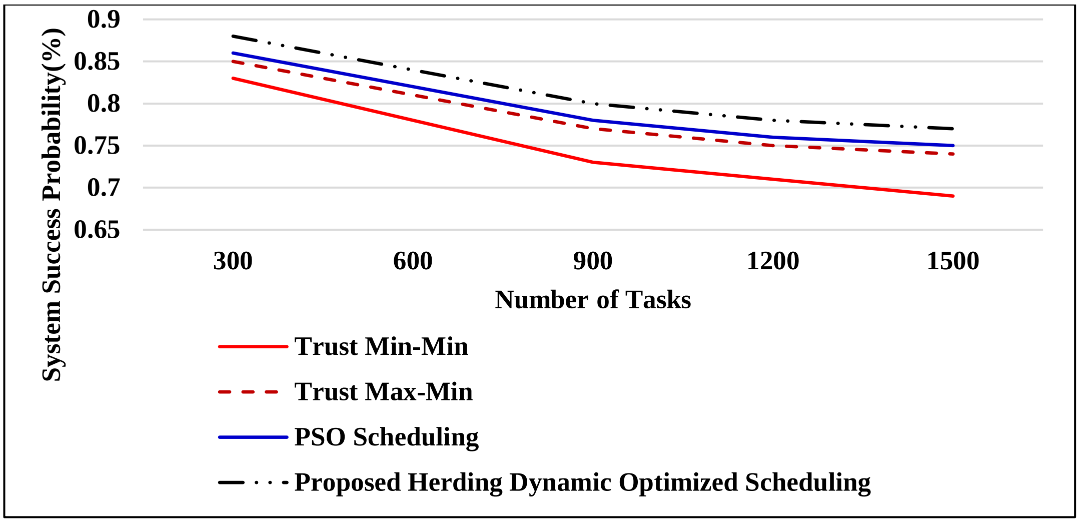

Fig. 4. System Success Probability for Proposed Herding Dynamic Optimized Scheduling

Table 7. Comparative Study of Trust Min-Min, Trust Max-Min, and Proposed trust based PSO, Herding Dynamic Optimized Scheduling algorithms on the basis of Success probability

No. of Tasks	Herding Dynamic Optimized Scheduling Algorithm improvement over Trust Min-Min Algorithm	Herding Dynamic Optimized Scheduling Algorithm improvement over Trust Max-Min Algorithm	Herding Dynamic Optimized Scheduling Algorithm Improvement over Trust based PSO scheduling Algorithm
300	5.68%	3.40%	2.27%
600	7.14%	3.57%	2.38%
900	8.75%	3.75%	3.75%
1200	8.97%	3.84%	2.56%
1500	10.38%	3.89%	2.59%

From the above **Table 7**, it can be observed that the proposed herding dynamic optimized scheduling has higher success probability by 5.68%, 3.40% & 2.27% for 300 number of tasks and by 7.14%, 3.57% & 2.38% for 600 number of tasks and by 8.75%, 3.75% & 3.753% for 900 number of tasks and by 8.97%, 3.84% & 2.56% for 1200 number of tasks and by 10.38%, 3.89% & 2.59% for 1500 number of tasks when compared with Trust Min-Min, Trust Max-Min[5] and proposed Trust based PSO scheduling respectively.

5. Conclusion and Future work

Cloud computing is opening gates to large possibilities for storage and online services. The primary purpose of the trust is inculcating faith into the entities of service providers that can influence its development in cloud computing and providing security services. In this paper an optimized task scheduling and a dynamic optimization technique is proposed to find an optimal trust weights in cloud during scheduling. Using optimal trust weights approach and dynamic herding optimization algorithm, efficiency of resource allocation in terms of task completion and response time is attained.

The proposed herding dynamic optimized scheduling has lower schedule length, higher resource utilization and higher system success probability when compared with trust min-min, trust max-min and PSO scheduling respectively as below results.

Results show that the proposed herding dynamic optimized scheduling has lower average schedule by 10.70%, 8.38% & 4.01% for 300 number of tasks, by 12.46%, 8.15% & 4.44% for 600 number of tasks, by 10.98%, 8.96% & 4.36% for 900 number of tasks, by 10.53%, 6.22% & 3.19% for 1200 number of tasks and by 15.28%, 9.12% & 4.65% for 1500 number of tasks when compared with Trust Min-Min, Trust Max-Min and Trust based PSO scheduling respectively. The proposed herding dynamic optimized scheduling has higher resource utilization by 7.60%, 5.43% & 3.26% for 300 number of tasks, by 6.66%, 4.44% & 2.22% for

600 number of tasks, by 7.44%, 5.31% & 2.12% for 900 number of tasks, by 6.38%, 4.25% & 2.13% for 1200 number of tasks and by 6.59%, 4.39% & 2.19% for 1500 number of tasks when compared with Trust Min-Min, Trust Max-Min and proposed Trust based PSO scheduling respectively. The proposed herding dynamic optimized scheduling has higher system success probability by 5.68%, 3.40% & 2.27% for 300 number of tasks, by 7.14%, 3.57% & 2.38% for 600 number of tasks, by 8.75%, 3.75% & 3.75% for 900 number of tasks, by 8.97%, 3.84% & 2.56% for 1200 number of tasks and by 10.38%, 3.89% & 2.59% for 1500 number of tasks when compared with Trust Min-Min, Trust Max-Min and PSO scheduling respectively. In the future research, we intent to adapt and to test the proposed approach on other challenges from the cloud computing realm. In future scope, the resource utilization has done in the dynamic and contemplate other nature inspired algorithm to give optimal solutions towards task scheduling in cloud.

References

[1] Arunarani, A. R., Manjula, D., &Sugumaran, V., "Task scheduling techniques in cloud computing: A literature survey," *Future Generation Computer Systems*, vol. 91, pp. 407-415, Feb. 2019. Article (CrossRef Link)

[2] B. Keshanchi, A. Souri, N.J. Navimipour, "An improved genetic algorithm for task scheduling in the cloud environments using the priority queues: formal verification, simulation, and statistical testing," *Journal of. Systems and Software*, vol.124, pp.1–21, Feb.2017. Article (CrossRef Link)

[3] Sridevi, Ch., Varma, P. S., & Krishna, S. R., "Design and Development of Efficient Cloud Scheduling Algorithm based on Load Balancing Analytics," *International Journal of Computer Applications*, vol.138(12), pp.21-27, march, 2016. Article (CrossRef Link)

[4] Kumar, P., &Verma, A., "Scheduling using improved genetic algorithm in cloud computing for independent tasks," in *Proc. of the International Conference on Advances in Computing, Communications and Informatics*, ACM, pp. 137-142, August, 2012. Article (CrossRef Link)

[5] Sai Prasad, V.H., and Raj Kumar, G.V.S., "Implementation and Evaluation of a Trust Model With Data Integrity Based Scheduling in Cloud," *International journal of Intelligent Systems Technologies and Applications*, vol.19, no.4, pp.348-361, sept. 2020. Article (CrossRef Link)

[6] Jehovah, A. N., & Desai, M. R., "Optimizing Multi Objective Based Dynamic Workflow Using ACO and Black Hole Algorithm in Cloud Computing," in *Proc. of 2019 3rd International Conference on Computing Methodologies and Communication (ICCMC)*, pp. 1144-1147, March 2019. Article (CrossRef Link)

[7] Ibrahim, E., El-Bahnasawy, N. A., &Omara, F. A, "Dynamic Task Scheduling in Cloud Computing Based on the Availability Level of Resources," *International Journal of Grid and Distributed Computing*, vol.10. no.8, pp.21-36, 2017. Article (CrossRef Link)

[8] Pang, S., Zhang, W., Ma, T., &Gao, Q, "Ant colony optimization algorithm to dynamic energy management in cloud data center," *Mathematical Problems in Engineering*, vol. 2017, Dec. 2017. Article (CrossRef Link)

[9] Dashti, S. E., &Rahmani, A. M., "Dynamic VMs placement for energy efficiency by PSO in cloud computing," *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 28, no.1, pp. 87-112, 2016. Article (CrossRef Link)

[10] Padmavathi, M., &Basha, S. M., "Dynamic and elasticity ACO load balancing algorithm for cloud computing," in *Proc. of 2017 International Conference on Intelligent Computing and Control Systems (ICICCS)*, pp. 77-81, June, 2017. Article (CrossRef Link)

[11] Shaikh, R., &Sasikumar, M., "Trust model for measuring security strength of cloud computing service," *Procedia Computer Science*, vol. 45, pp.380-389, Dec. 2015. Article (CrossRef Link)

[12] Chen, Z., Tian, L., & Lin, C., "Trust evaluation model of cloud user based on behaviour data," *International Journal of Distributed Sensor Networks*, 14(5), May, 2018. Article (CrossRef Link)

[13] Ali Al-maamari and 2 Fatma A. Omara, “Scheduling Using PSO Algorithm in Cloud Computing Environments,” *International Journal of Grid Distribution Computing*, Vol. 8, no.5, pp.245-256, 2015. Article (CrossRef Link)

[14] Manasrah, A. M., and Ba Ali, H., “Workflow scheduling using hybrid GA-PSO algorithm in cloud computing,” *Wireless Communications and Mobile Computing*, vol. 2018, pp.1-16, Jan. 2018. Article (CrossRef Link)

[15] A. H. Gandomi and A. H. Alavi, “Krill herd: A new bio-inspired optimization algorithm,” *Communications in Nonlinear Science and Numerical Simulation*, vol. 17, no. 12, pp. 4831–4845, Dec. 2012. Article CrossRef Link)

[16] Abualigah, Laith Mohammad, Ahamad Tajudin Khader, Mohammed Azmi Al-Betar, and Mohammed A. Awadallah, “A krill herd algorithm for efficient text documents clustering,” in *Proc. of Computer Applications & Industrial Electronics (ISCAIE), 2016 IEEE Symposium*, pp. 67-72, May, 2016. Article (CrossRef Link)

Dr. A.V.H. Sai Prasad is working as Associate Professor in the Department of Computer Science and Engineering, Neil Gogte Institute of Technology, Hyderabad. He received his Ph.D in Computer Science & Engineering from GITAM(Deemed to be University) Visakhapatnam in 2021. He received his Degree of Master of Engineering in Computer Science and Engineering from Sathyabama University, Chennai in 2007. He is having 24 years of experience which includes teaching and industry. He has published over 6 International journals and published one research article in national conference. He guided several students for getting their M.Tech degrees in Computer Science And Engineering. His current research interests are cloud security, cyber security and distributed systems.

Dr G.V.S Raj Kumar is working as Professor in the Department of Computer Science and Engineering, GITAM Institute of Technology, GITAM (Deemed to be University) **Visakhapatnam**. He received his Ph.D in Computer Science & Systems Engineering from Andhra University in 2012. He has published over 45 International and 2 National journals and published two books. He has guided 3 Ph.D students and guiding 6 Ph.D scholars. He guided several students for getting their M.Tech degrees in Information Technology and Data Science. His current research interests are information security, image processing. He is having several memberships and also a life member of computer society of India.