Cost and Completion Time based Sufferage Algorithm for Task Scheduling in Cloud Environment

H. Krishnaveni [1], Dr. D. I. George Amalarethinam [2], Dr. V. Sinthu Janita[3]
[1]*Cauvery College for Women, Tiruchirappalli*
[2] *Jamal Mohamed College, Tiruchirappalli*
[3] *Cauvery College for Women, Tiruchirappalli*
(E-mail: hkrish6677@gmail.com)

Abstract— **Cloud computing is omnipresent computing supporting the pay-per-use model. Therefore, the price of usage is one of the vital factors for cloud customers when selecting the service from cloud provider. Hence, the cloud service providers need to provide competitive costs of the services for their users. Also the cloud service providers need to optimize the utilization of the resources, with the competitive cost at the same time. In order to achieve this, there is a need for a new set of cost-effective task scheduling algorithms for the cloud. This paper introduces an algorithm for task scheduling based on some priorities for tasks in Sufferage heuristic and compared it to some of the traditional and recent cloud scheduling *algorithms.***

Keywords— *cloud computing, task scheduling, cost, resource utilization, makespan.*

I. Introduction

According to National Institute of Standards and Technology, Cloud computing could be a model for sanctioning convenient, on-demand network access to a shared pool of configurable computing resources which will be rapidly provisioned and free with least management effort or service-provider interaction [1]. The salient features of cloud computing are on-demand service, broad network access, resource pooling, rapid elasticity and measured service. Software as a service (SaaS), Infrastructure as a service (IaaS) and Platform as a service (PaaS) are the three basic services provided by the cloud [2]. Cloud IaaS enables the customer to use the services in a flexible environment. It provides the illusion that an unlimited resource allocation for a number of users. The creation and operation of a medium sized datacenter is a time consuming procedure and depends on the economic strength of the organization having interest to construct a datacenter. Cloud computing provides services with least cost compared to the development of the data enter. Cloud Service Providers (CSPs) allow the users to decide on the machine hours based on their requisite regardless of the costs without paying a premium for large scale. This type of resource elasticity is distinctive in the history of computing [3].

Cloud Service Providers put forward numerous online services based on Service Level Agreement between the provider and the customer. However, an important role between providers and customers relationship is the pricing model for which they must agree.

Each provider has his own scheme for calculating the price (has an accounting system) for the cloud services offered for clients. The provider’s goal is to achieve a greater benefit, while each client’s goal is to have the greatest service for low price. Therefore, satisfying each party needs the best rating methodology. The price charged is one of the most important metrics that a service provider can control to encourage the usage of its services [4].

The price has a major impact in economic aspect, where fairness and competitive pricing in a multi-provider marketplace affect the actual pricing [5].

Pricing for competition and fairness affects decisions within the style of user applications and system infrastructure. In fact, rating fairness balances user price and cloud service supplier profit. Pricing model in Cloud Computing is versatile than ancient models. Every cloud provider has his own pricing scheme. Main spotlight of Cloud Computing is to fulfill and guarantee quality of service (QoS) for customers.

The popular CSPs are Google Computed Engine and Amazon EC24. They have various schemes which vary in pricing as well as in computing capacities [6]. For example, Amazon EC2 offers micro, small, medium, large and extra large On-Demand instances prices. In Google Compute Engine, based on the type of machine needed, the charges for usage on a monthly basis will be applied. The users of the Cloud services are going to be content, if the service is reasonable to their budget. Once the user submits the collection of tasks (Application) to the cloud environment, it will take care of scheduling and monitors the execution of the tasks. The key issue here is the scheduling (matching tasks to the resources). The scheduler should minimize the overall execution time of all the tasks and therefore the financial price. This can be achieved only with the aid of the efficient scheduling algorithms. This paper describes the related works in section 2. In section 3, proposed methodology is given. Experimental results are explained in section 4.At the end, section 5 concludes the paper.

II. RELATED WORKS

A number of research work is going on the concept of scheduling especially task scheduling. Various algorithms have been designed to schedule the tasks in cloud computing. Naseem A.AL-Sammarraie et al.[7] proposed a hybrid algorithm **P**erformance and **C**ost **A**lgorithm(PCA) that aims not only to the minimization of the services cost paid by the user and maximizing the profit gained by the provider of services renting, but also aims to optimize the performance of these services by minimizing the services completion time and maximizing the resource utilization of the resources, in order to enable the provider to provide the best and most efficient services with highly competitive prices.

Yogita Chawla et al. [8] developed a dynamically optimized task scheduling algorithm that is combined with task grouping results in reducing the processing time as well as cost. This scheduling algorithm is very much useful to both user and cloud provider. The usage of task grouping for scheduling after prioritization reduces the processing time over the algorithm without task grouping.

S. Selvarani et al. [9] introduced a scheduling approach that employs an improved cost based scheduling algorithm for efficient mapping of tasks to the resources available in the cloud. This scheduling algorithm boosts the computation/communication ratio by grouping the tasks based on the processing capability of the particular cloud resource and sends the grouped tasks to that resource.

Cao. Q et al. [10] presented an algorithm which applies the concept of cost-based priority by calculating the cost of each individual use of the resources and the corresponding profit of using these resources. Based on these calculations, priorities are assigned to the tasks. Then the tasks are sorted in three levels such as High, Medium and Low levels. The tasks with highest profit are assigned with the highest priority. If new task arrives its priority is calculated and it is assigned to the end of the appropriate level.

Based on bio-inspired algorithm and economic method, dynamic resource allocation approach is proposed by Xingwei Wang et al. [11] for cloud services using intelligent combinatorial double auction method .This framework is devised to provide a comprehensive solution. A price formation mechanism is also proposed to predict price and determine eligible transaction relationship. Winner Determination Problem (WDP) is optimally solved by the improved Paddy Field Algorithm (PFA).

Ram Kumar Sharma et al. [12] proposed a scheduling algorithm in which the incoming tasks are grouped on the basis of their type deadline constrained or low cost requirement. Then the tasks are prioritized according to deadline or profit after initial grouping. So that the tasks with shorter deadline need to be scheduled first and also the tasks resulting in more profit should be scheduled on low cost machines. It is observed that this proposed algorithm enhances cost and completion time of tasks as compared to sequential assignment.

Arnav Wadonkar et al. [13] developed a hybrid non-preemptive scheduling algorithm to map the tasks to the given resources by considering both the length and deadline of the tasks. The authors also compared the results with the existing algorithms which consider either task length or task deadline and proved better. But the cost of execution is not considered.

Alka Vohra et al. [14] introduced an algorithm which considers the parameters task length and user priority to schedule the tasks based on deadline aware rank system. Makespan and task profit are improved than min-min and max-min algorithms. Also the number of failed tasks has been decreased.

As cloud environment is market oriented business model, cost is the key issue to be considered. Especially in scheduling, the execution cost is very important and to be reduced to attract the customers. So that the cloud service providers can retain their position in the competitive milieu. This paper is focussed on the task scheduling based on the cost of the resources by modifying the sufferage algorithm.

III. PROPOSED METHODOLOGY

Most of the conventional algorithms of scheduling in cloud computing don't make any consideration for the task's cost, where the task is mapped to any available resource as soon as it arrives. This presides to some problems such as over-priced cloud services in case of high-volume simple tasks and under-costed in low-volume complex tasks [10]. To vanquish these problems and since most of the people think of current cloud computing offerings as purely "pay by the drink" compute platforms [15], in this paper an algorithm is proposed which combines the time and resource cost while scheduling the tasks. Sufferage heuristic is modified by including sufferage value of time and cost, combined with load balancing. The main objective of this algorithm is not only the minimization of the services cost paid by the user but also aims to mitigate the overall completion time and lever up the utilization of the resources, in order to facilitate the provider to provide the best and most efficient services with extremely competitive prices.

After a deep study of the literature of meta task scheduling, it is observed that in the heterogeneous computing environment, the list of conventional scheduling heuristics such as min-min, max-min and sufferage, showed better completion time over other algorithms. Especially, the sufferage algorithm is suitable for both short and long tasks. Comparatively, it is better than the other two bench mark algorithms. Hence, this paper tries to modify sufferage algorithm by incorporating the execution cost of the tasks with the completion time to yield better makespan and total cost.

For any static scheduling algorithm, the input is given in the form of matrix only. Here the Expected Time to Compute matrix and cost matrix are the inputs. The costs of the resources are taken from google app engine. The completion time is calculated by using [16] and the completion cost is calculated by using (1).

$$\text{Completion Cost} = \text{Cost}_{ij} + RC_j \quad \text{------} \quad (1)$$

where,

Cost_{ij}= Execution cost of task i on resource j

RC_j = Ready cost of resource j

For each task in the unassigned task list, the first and second minimum completion time is selected and sufferage value (SVCT) is calculated. In the same way, from the completion cost matrix, first and second maximum cost is selected. Then cost sufferage (SVC) is computed.

Both the sufferage values SVCT and SVC are compared with first minimum completion time and first maximum cost respectively to select a particular task for scheduling. If the condition is satisfied, the corresponding task is assigned to the resource that gives minimum completion time for that task. Also check that task is suitable for that resource. Remove the task from the unassigned task list. Update ready time of the resource. Also update the completion time and completion cost. Then compute makespan, total execution cost. The resource utilization is computed by using the formula given in [16]. Following is the pseudo code of the proposed methodology.

PSEUDO CODE FOR COST AND COMPLETION TIME BASED SUFFERAGE ALGORITHM (CCTSA)

Input: Expected Cost to Compute (ECC) and Expected Time to Compute(ETC) Matrices

Output: Scheduled Task

```
1.  While (Unassigned_Task_Count >0)
2.      For each task
3.      For each resource
4.      Calculate completion time and completion cost
5.      For each task
6.      Find First and Second Minimum Completion Time
                        FMICT_i , SMICT_i of task T_i
        Find First and Second Maximum Cost Value FMXC_i ,
                        SMXC_i of task T_i

7.      SVCT_i= SMICT_i - FMICT_i        //Calculate
                        completion time sufferage value
8.      SVC_i =FMXC_i -SMXC_i            // Calculate cost
                        sufferage value
9.      End For
10.     Ch_Task=nul l     //Initially Chosen Task is assigned
                        as null
11.     For i = Unassigned_Task_Count to 0
12.     If (SVCT_i>FMICT_i  &&  SVC_i < FMXC_i )
13.         Ch_Task=T_i
14.            break;
15.     Else
16.         Ch_Task=Tn

17.     End if
18.     End for
19.     Assign Ch_Task (selected Task) to resource j that gives the minimum completion
        time of task and cost also check that task is suitable for this resource j
20.     Remove Ch_Task from Unassigned_Task list.
21.     Update Ready time of the resource
22.     Update completion time and completion cost

23.     Compute makespan and total cost
24.  End while
25.     Compute Resource Utilization
```

IV. EXPERIMENTAL RESULTS

The proposed algorithm CCTSA is implemented in cloud sim 3.0 and the results are compared with existing algorithm [16]. Specification of the resources such as number of resources, processing speed, bandwidth and cost of the processor are given in table I. The number of tasks, number of instructions and data details are given in table II. By using the table I and table II, ECC and ETC matrices are constructed and given as input.

TABLE I. SPECIFICATION OF THE RESOURCES

Resources	Processing Speed (MIPS)	Related bandwidth (Mbps)	Cost of the processor (INR)
R1	50	100	0.03
R2	100	200	0.12
R3	200	250	0.24

TABLE II. SPECIFICATION OF THE TASKS

Tasks	Number of Instructions(MI)	Data(Mb)
T1	206	44
T2	50	95
T3	128	64
T4	69	30
T5	118	59
T6	112	47
T7	21	39
T8	200	61
T9	90	23
T10	45	23

The execution time for each task on each resource can be computed using (2).

$$\text{Execution time} = (MI/MIPS + Mb/Mbps) \quad (2)$$

The execution cost of each task on each resource is calculated by using (3)

$$\text{Execution cost}=MI * \text{Cost of the processor} \quad (3)$$

Where,

MI - Million Instructions
MIPS – Million Instructions per Second
Mb – Mega Bits
Mbps – Mega Bits per Second

TABLE III COMPARISON OF THE RESULTS

Algorithm	Evaluation metrics		
	Makespan (sec.)	Cost(Rs)	Resource utilization (%)
CCTSA(proposed)	4.172	21.30	91.40
ETSA (existing)	3.915	25.76	88.98

Table III shows that the CCTSA outperforms the existing ETSA in terms of cost and resource utilization. But the makespan obtained by CCTSA is greater than ETSA.

Table IV summarizes the makespan produced by CCTSA and ETSA for different set of inputs.

Table IV Makespan

No. of tasks	No. of resources	Makespan (sec.)	
		CCTSA (proposed)	ETSA
25	5	10.23	8.9
50	7	27.51	22.4
75	8	48.3	46.1
100	10	74.5	67.9
150	12	122.4	113.7
200	14	153.8	137.8

Table V summarizes the total cost produced by CCTSA and ETSA for different set of tasks and resources.

Table V Total cost

No. of tasks	No. of resources	Cost(Rs)	
		CCTSA (proposed)	ETSA
25	5	30.1	33.3
50	7	54.7	59.5
75	8	82.5	93.9
100	10	111.6	123
150	12	143.4	162.8
200	14	171.2	190.6

Table VI summarizes the resource utilization obtained by CCTSA and ETSA for different combination of inputs. Fig.1 discloses that the makespan resulted from CCTSA is greater than that of ETSA. This is due to the high task heterogeneity. The cost analysis graph in fig. 2 expresses the total execution cost produced by the proposed CCTSA is lesser than the existing ETSA. The graphical representation of the metric resource utilization in fig.3 implies that CCTSA utilized the resources efficiently than ETSA.

Table VI Resource utilization

No. of tasks	No. of resources	Resource utilization (%)	
		CCTSA (proposed)	ETSA
25	5	90.01	87.00
50	7	91.40	88.20
75	8	90.50	86.92
100	10	90.00	86.70
150	12	89.21	86.50
200	14	89.05	85.11

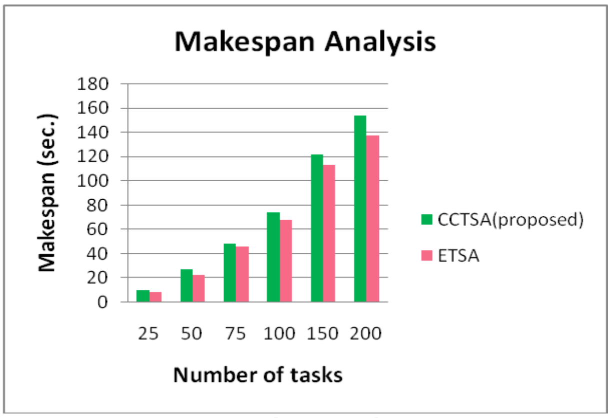

Fig.1 Makespan Analysis

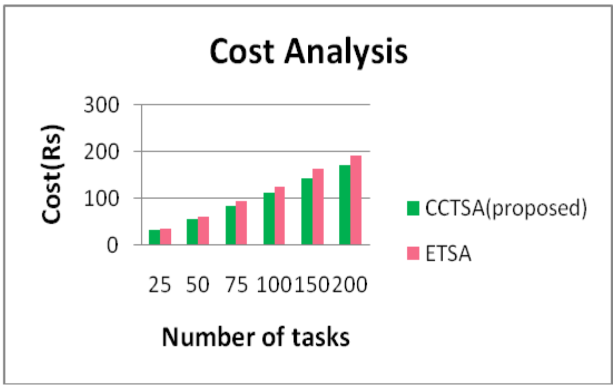

Fig.2 Cost Analysis

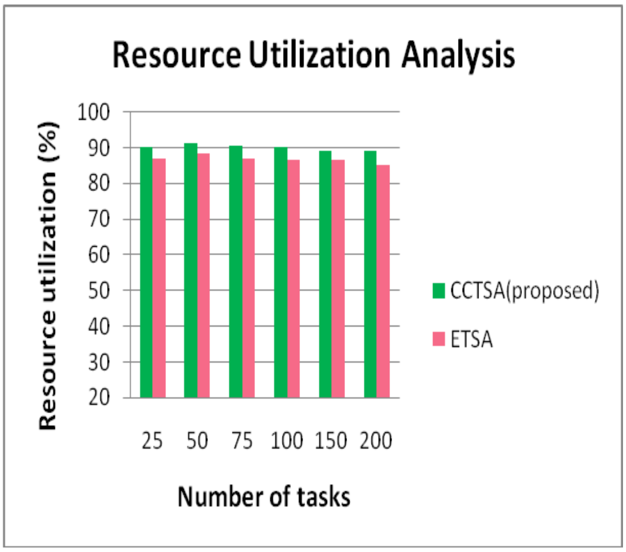

Fig.3 Resource Utilization Analysis

Fig.4 displays implementation window of the proposed CCTSA using Cloud Sim.

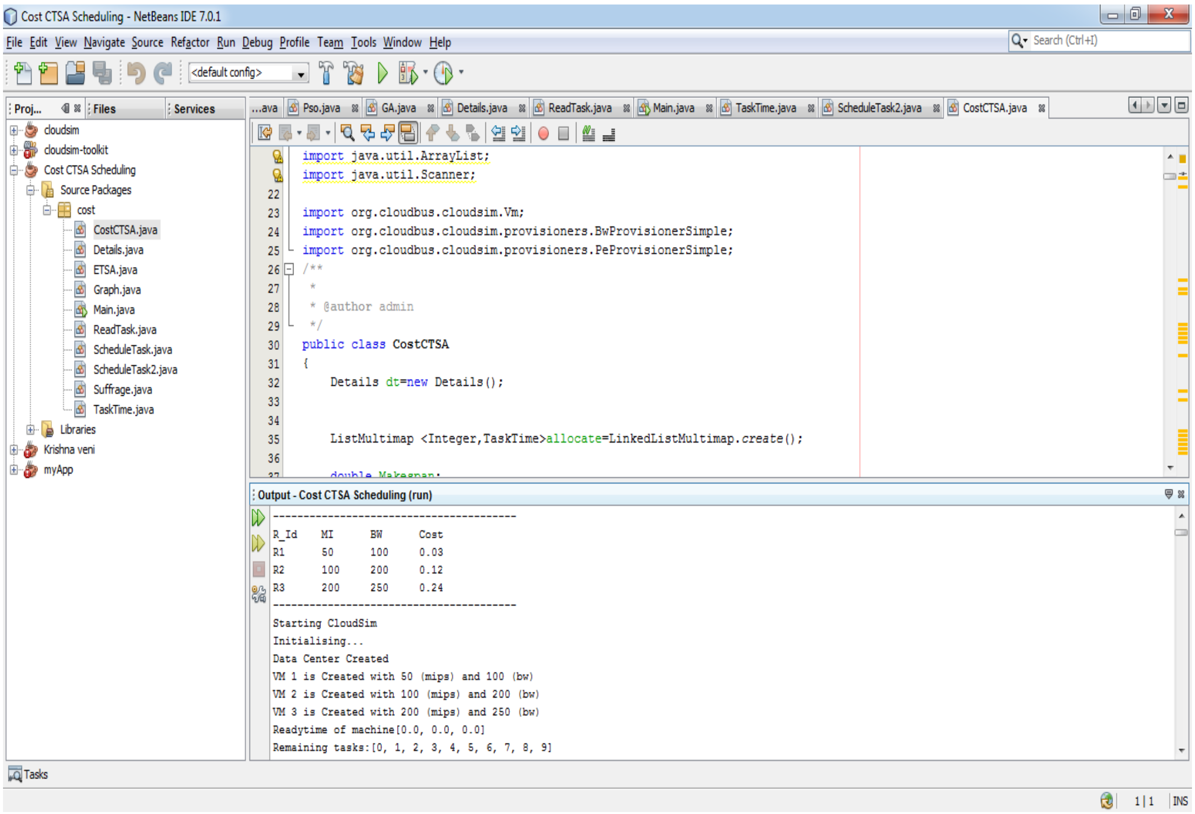

Fig. 4 Implementation window

V. CONCLUSION

In most of the traditional cloud scheduling algorithms, the schedulable task is mapped either to the resource which executes it in the short span of time or to the available machine as soon as it arrives without taking in to account the cost of the tasks. Some of the algorithms do consider the priority or the completion time but pay no attention to the cost of execution. This paper, presented the cost and completion time based sufferage algorithm to schedule the tasks to the machines in an effective manner. The outcome of the algorithm also proves the same as the cost is minimized and utilization of the resources is increased. Since completion time and cost are inversely proportional to each other, there is a need of efficient algorithms to balance them. It can be achieved by applying powerful meta heuristic algorithm for task scheduling. Even then, many issues are open such as the heat generation and energy consumption by the resources to be resolved, are left as future work.

References

[1] V.E.Unnamalai, J.R.Thresphine,"Service-Oriented Architecture for cloud computing", International Journal of Computer Science and Information Technologies, Vol. 5, Iss. 1, pp. 251-255, 2014.

[2] H.Krishnaveni, Dr.V.Sinthu Janita Prakash, " A Survey on task scheduling algorithms in Cloud Computing Environment", International Journal of Applied Engineering Research, Vol.10, Iss.82, pp.394-400, 2015.

[3] D.I. George Amalarethinam, T. Lucia Agnes Beena, "Customer Facilitated cost-based Scheduling (CFCSC) in Cloud", Elsevier, Science Direct, Procedia Computer Science, Vol. 46, pp. 660-667, 2015.

[4] M. Al-Roomi, Sh. Al-Ebrahim, S. Buqrais and I. Ahmad, "Cloud Computing Pricing Models: A Survey", International Journal of Grid and Distributed Computing, Vol. 6, iss. 5, pp. 93-106, 2013.

[5] H. Wang, Q. Jing, R. Chen, B. He, Zh. Qian and L. Zhou, Distributed Systems Meet Economics: Pricing in the Cloud, HotCloud '10, June 2010.

[6] Google AppEngine. Available from URL:http://code.google.cm/appengine

[7] Naseem A.AL-Sammarraie, Mohammed .F. Al-Rahmawy and Magdi Z. Rashad," A scheduling algorithm to enhance the performance and the cost of Cloud Services", Computer Engineering and Intelligent Systems, Vol. 6, iss. 8, 2015.

[8] Yogita Chawla, Mansi Bhonsle, "Dynamically optimized cost based task scheduling in cloud computing", International Journal of Emerging Trends and Technology in Computer Science, Vol. 2, Iss. 3, ISSN: 2278-6856, June 2013.

[9] S. Selvarani, Dr. G. Sudha Sadhasivam, "Improved cost based algorithm for task scheduling in cloud computing", IEEE, 2010.

[10] Cao, Q., Wei. Z., Gong, W. M." An optimized algorithm for task scheduling based on activity based costing in cloud computing", In Proceedings of the 3rd International Conference Bioinformatics and Biomedical Engineering (ICBBE). IEEE Computer Society, Beijing, China, 1-3. 2009.

[11] Xingwei Wang, Xueyi Wang, Hao Che, "An Intelligent Economic Approach for Dynamic Resource Allocation in Cloud Services", IEEE Transactions on Cloud Computing, Vol. 3, iss. 3, July-September 2015.

[12] Ram kumar Sharma, Nagesh Sharma, "A dynamic optimization algorithm for task scheduling in cloud computing with resource utilization", International Journal of Scientific Engineering and Technology, Vol. 2, iss. 10, pp. 1062-1068.

[13] Arnav Wadhonkar, Deepti Theng, "A task scheduling algorithm based on task length and deadline in cloud computing ", International Journal of Scientific & Engineering Research, Vol. 7, iss. 4, April 2016.

[14] Alka Vohra, Suchitra, "Deadline aware Sufferage based task scheduling in cloud computing", International Journal of Innovative Research in Computer and Communication Engineering, Vol. 5, iss. 5, May 2017.

[15] Sun Microsystems: Take Your Business To A Higher Level - Sun Cloud Computing Technology Scales Your Infrastructure to Take Advantage of New Business Opportunities. Technical Report, 2009.

[16] Krishnaveni H., Sinthu Janita Prakash V.(2019) " Execution Time based Sufferage Algorithm for Static Task Scheduling in Cloud ",In: Peter J.,Alavi A., Javadi B.(eds) Advances in Big Data and Cloud Computing . Advances in Intelligent Systems and Computing, Vol.750. Springer, Singapore. Online ISBN: 978-981-13-1882-5

Authors Profile

Mrs H Krishnaveni pursed Bachelor of Science from Bharathidasan University, Trichy in 1997 and Master of Computer Applications in Bharathidasan University in year 2000. She is currently pursuing Ph.D. and currently working as Associate Professor in Department of Computer Science, Cauvery College for Women, Tiruchirapalli since 2001. She is a member of ISSE. She has published more than 5 research papers in reputed international journals and conferences including Springer and it's also available online. Her main research work focuses onCloud Scheduling, Cloud Security and Privacy, Big Data Analytics and IoT. She has 18 years of teaching experience.

Dr. D.I.George Amalarethinam is currently working as an Associate Professor, Bursar and Director of MCA , Jamal Mohamed College, Tiruchirappalli. He has 31 years of teaching experience. He is more interested in doing research in parallel processing, grid computing and cloud computing. 52 scholars finished their MPhil and 11 candidates completed their Phd under his guidance.He has published 120 research papers in various reputed international journals and presented more than 90 papers in national and international conferences. He also acted as resource person in more than 60 workshops, seminars and conferences.He has published around 8 books. He is a member in various professional bodies.

Dr. V. Sinthu Janita pursed Master of Computer Applications and Ph.D from Bharathidasan University. She is currently working as a Professor and Head in the PG &Research Department of Computer Science, Cauvery College for Women, Tiruchirapalli. She is a member of IEEE, Computer Society of India and a life member of ISSE.She has published more than 35 research papers in reputed international journals including Thomson Reuters (SCI & Web of Science) and conferences including IEEE, Elsevier, ACM, Springer and it's also available online. Her main research work focuses on Communication Networks, Network Security, Big Data Analytics, Data Mining and Mobile Communications. She has 23 years of teaching experience and 14 years of Research Experience..