THE MULTI SESSION OPTIMIZATION BASED SCHEDULING ALGORITHM FOR SMART CLOUD COMPUTING

T. Kiruthiga[1] and S. Praveena[2]

[1]*Department of Electronics and Communication Engineering, Vetri Vinayaha College of Engineering and Technology, India*

[2]*Department of Biomedical Engineering, Paavai College of Engineering, India*

Abstract

The Cloud computing is considered to be one of the most important emerging technologies today. Its important functions are to create situations in which different tasks can be effectively completed in different contexts. Its resource utilization and planning algorithms play a key role in making its work more efficient. The main focus here is time management. That means you need to have cloud servers ready to plan certain tasks in advance and get them done on time. Thus, various planned tasks are corrected. This greatly reduces the allocation time for the overall transfer on the servers. And its overall working time is doing the work that is available at the specified time. It can be calculated by subtracting the non-working time from the total time. The currently proposed algorithm is based on a whole set of computational scenarios. Based on the slot's allocation, the task is divided into total working hours. It can also be divided into fixed slot for scheduled data and variable slot for instant work. Work is being carried out on this basis. As its efficient working time increases, the total cloud server working hours will be improved.

Keywords:

Cloud Computing, Resource Utilization, Planning Algorithm, Slot

1. INTRODUCTION

. The Cloud computing is a system in which computing capabilities are available over the Internet. You can obtain and use the required amount of computing skills from companies that provide computing capabilities. This technology allows a company to obtain its computing needs from various locations and computer companies on an application fee basis without having to build a framework on computers and software [1]. If you want to easily understand about cloud computing, then you assume the below example. Suppose we build a wind farm or a system in our house that can generate electricity to generate the electricity we need for our home. The cost of building and maintaining that system will be much, much higher than the cost of purchasing the electricity we use from outside organizations, including the government. This same idea is the main reason for the emergence of cloud computing [2]. Big companies like Google are developing a system called Cloud Computing. They provide high performance servers, storage, database, networking, software application. If you want to use these you can use the Internet for a very small amount of money (Internet). Cloud computing is a term used to describe the advanced technology of the computer and application we are currently using. Any cloud computing that is connected to the Internet can be used for storage, including applications [3]. The largest companies are integrating and maintaining billions of electronic devices in the largest locations for cloud computing technology. If you want to create a website, you need storage space to store the relevant files. Purchasing high-performance computers and drives and keeping them ON 24 hours a day and doing maintenance work can be costly and time consuming. For that you can buy the required storage at the lowest price through the cloud computing of the largest company [4]. With millions of servers, high-speed technology, refrigeration, and maintenance technicians in cloud computing, you can get high-speed service. Larger companies manage larger data centers to provide cloud computing services. There are current technological improvements being made in hardware and software [5]. Doing this when you have separate data centers will cost more. Failure to do so will result in a defect in the speed included. Internet security is emerging as the biggest problem. Larger companies will continue to improve the security features of their data centers themselves. Therefore, all your information will be secure.

With the development of cloud computing, many developers of web applications / services no longer have the fundamental problem of where to run it from. We can provide our own processors and services without the need for our own hardware, storage and permanent uninterrupted internet connection [6]. Many leading companies, including Amazon, Google, and Microsoft, offer cloud computing as a service. We call these service providers. Just like renting a car, bicycle, etc., we can get and use the computer resources we need only for the time we need. It can be returned as soon as the need arises. We can understand these as services built on top of each other. Configuration service is the provision of basic necessities such as hardware, internet connection and memory [7]. Ex: DigitalOcean, AWS EC2, Google Compute Engine, Microsoft Azure. Operating services are services that help create and maintain an operating system, operating system (Runtime/SDKs), and database database on these objects. Programming knowledge and clarity are required to use these. Ex: AWS DMS, Google App Engine. Finally, a software service is a device that runs software on a platform and serves it to its users. These can be free services or paid services. e.g. Google processors including GMail.

2. RELATED WORKS

Medina-Acosta et al. [1] performed the planning process using some of the popular slot selection algorithms used in cloud computing. The barriers of this system are solved as it makes the planning process using certain actions that are optimal for better resource allocation.

Bassa et al. [2] have developed an efficient system based on a customer. This method can improve long-term events with the company's customers. They are inputting customer data based on the Hungarian algorithm. Shaw et al. [3] defined the limits of the algorithms already in the classification. That is, they say that when the number of processors is greater than the number of jobs, there is more time and less data. Thus, they have made some changes without coming to the peak of this system. Penner et al. [4] made some changes to the load transfer and execution mode.

Thus formed the temporary clouds. This means that a temporary network is created by combining nearby achievements that can provide temporary services. Centaurs will leave the network when usage stops. Its use will be calculated according to its use.

Penner et al. [5] proposed some demo models regarding the Federation of Temporarily Designed Clouds. Its connectivity and possibilities were established based on the classification of Wi-Fi modules that are integrated with Android modules. The problem with this is that it is a makeshift mega-federation that is unable to deploy the existing Islamic missions there. Li et al. (2014) conducted a study of some of the industry markets that have emerged in the commodity system. Its interface was calculated based on the basis of service and time for them. Thus, its resource allocation methods worked better and revealed the volume.

3. PROPOSED METHOD

The proposed method is improving the speed with which users can quickly and inexpensively lead to the return of technological infrastructure resources shown in fig.(1). The cost of computing as a whole will not change; however, providers will have to incur obvious costs and extended costs over the long term. The cost is greatly reduced and the cost of capital (CapeEx) is converted into operating cost (OpEx). This pretentiousness can reduce barriers to entry, as the infrastructure is provided by a third party and does not have to be purchased for simultaneous or continuous intensive computing work [8]. Pricing for utility computing is fine-grained with application-based options and requires a few IT skills (within the company) to implement. Some would argue that spending less on computing resources would shift the IT burden from the company to outsourced providers. Furthermore, any cost reduction benefits should be weighed against the risks associated with loss of control, access and security [9].

Step 1: Set input tasks with scheduling and partition cloud details

Step 2: Expected cloud output to task scheduling; Set output parameters;

Step 3: for ($a = a_1, a_2, a_3\ldots,a_n$)

Step 4: Calculate the task end time

Task end time ET(a)=Start time of ST(a)+Duration of task D(a)

Step 5: end for

Step 6: Initiate the matrix scheduling procedure; find LTU

DISCOVER_LTU (*TL*, *ST*(a), *ET*(a), *T*)

Step 7: Initiate the matrix scheduling procedure, find CLTU

DISCOVER_CLTU (*TL*, *ST*(a), *ET*(a), *T*)

Step 8: Identify the RPM, CPM, TPM values

where, ROM – Row prospect Matrix, CPM - Column prospect Matrix, TPM - Total prospect Matrix

Step 9: Cover all zero values in matrix CPM with the help of horizontal lines (*HL*)

Step 10: if ($HL == 0.5$)

Then the concluding transfer matrix TPM is similar to CPM

Step 11: else

To find the smallest element (uncovered)

Subtract that smallest uncovered element to the total uncovered element

Add the elements to cover twice with the matrix

Step 12: end if

The Reliability is enhanced by the use of multiple redundant platforms that ensure cloud computing is optimal for business continuity and disaster recovery. However, many major cloud computing services are vulnerable to service tracks and IT and business executives are less vulnerable when they are compromised. The Security data centralization and the Resources focused on security are improving rapidly due to overcrowding, etc., but concerns about the loss of control over particularly sensitive data persist [10].

The Security is better than when there were outdated methods, because providers are able to provide resources to solve security issues that many customers cannot afford. Permits can be requested by providers, but accessing the audit permit on their own can be complicated or even impossible. Ownership, control and access to data can be made more complex by cloud providers, which can sometimes be as difficult as accessing "direct" assistance with current utilities. Managing sensitive data under the cloud model is in the hands of cloud providers and third parties. Sustainability comes from improved resource utilization, more efficient systems and carbon balance [11]. However, computers and related infrastructure are major energy consumers. Given (server-based) computing work it will use x amount of power whether it is onsite or offsite.

4. RESULTS AND DISCUSSION

The Cloud Construction Systems of software systems involved in the delivery of cloud computing include hardware and software developed by a cloud architect working for the cloud coordinator. Commonly referred to as web services, the application is associated with multiple cloud components that interact with each other in the programming interface. It is largely reminiscent of the UNIX philosophy of having multiple programs do one thing well and work together in a global interface. The Complexity is controlled and the resulting systems are better manipulated than their monolithic alignment systems. In order to calculate the nature of the proposed algorithm its modules are classified as follows.

Table.1. Task scheduling parameters

Task Name	Start Time (ST)	Task Duration (D)	End time (ET)
T_1	01:10	00:20	01:30
T_2	02:40	00:40	03:20
T_3	04:50	00:22	05:12
T_4	06:22	00:31	06:53
T_5	08:45	00:15	09:00
T_6	12:10	00:20	12:30
T_7	15:15	00:25	15:40

4.1 CALCULATION OF LEASE TIME

The lease time of a cloud environment in scheduling was calculated with the help of the following equation. (1) This is nothing but the time taken to complete the task a. meanwhile it's describe the difference between the end time (ET) and start time (ST) to complete the task a. the Table.2, presents the various start and end time data set.

$$DISCOVER\_LTU\ (TL, ST(a), ET(a), T) \quad (1)$$

where,

LTU = Least time taken to complete the tasks a and start the next task a_1, *TL* = Length of the task, *ST*(*a*) = Start time of the task '*a*', ET(*a*) = End time of the task '*a*' and *T* = task duration

Table.2. Calculation of Lease time

Task Name	Start Lease Time (SLT)	Task Duration (D)	End Lease time (ELT)
T_1	1:00	0:10	1:20
T_2	2:30	0:30	3:10
T_3	4:40	0:12	5:02
T_4	6:12	0:21	6:43
T_5	8:35	0:05	8:50
T_6	12:00	0:10	12:20
T_7	15:05	0:15	15:30

4.2 CALCULATION OF CONVERSE LEASE TIME

The Converse lease time of a cloud environment in scheduling was calculated with the help of the Eq.(2) This is nothing but the time taken to complete the task a. meanwhile it's describe the difference between the end time (ET) and start time (ST) to complete the task a. the below Table.3, presents the various start and end time data set.

$$DISCOVER\_CLTU\ (TL, ST(a), ET(a), T) \quad (2)$$

Table.3. Calculation of Converse Lease time

Task Name	CSLT	Task Duration (D)	CELT
T_1	0:57	0:07	1:17
T_2	2:27	0:27	3:07
T_3	4:37	0:09	4:59
T_4	6:09	0:18	6:40
T_5	8:32	0:02	8:47
T_6	11:57	0:07	12:17
T_7	15:02	0:12	15:27

Based on the above calculations, the scheduling process was completed by the cloud data server. The particular start and end time data sets are introduced by the algorithms and the start time slots are easily spitted. Each time slots have 1 hour gap. This may further enhance to assigned time slot and unassigned (free) slots. Based on this the tasks were allocated in each time slots.

Table.4. Final task scheduling sequence

Start Time	End Time	Task Scheduling
0:00	1:00	
1:01	2:00	T1
2:01	3:00	T2
3:01	4:00	
4:01	5:00	T3
5:01	6:00	
6:01	7:00	T4
7:01	8:00	
8:01	9:00	T5
9:01	10:00	
10:01	11:00	
11:01	12:00	
12:01	13:00	T6
13:01	14:00	
14:01	15:00	
15:01	16:00	T7
16:01	17:00	
17:01	18:00	
18:01	19:00	
19:01	20:00	
20:01	21:00	
21:01	22:00	
22:01	23:00	

The Cloud architecture extends to the client in access to web browsers and / or software utility cloud applications. The Cloud storage is loosely integrated into a centralized metadata function that scales hundreds of data nodes, each of which sends data to applications or users arbitrarily.

5. CONCLUSION

In general, modules of these special features in the cloud network are designed to easily modify a variety of functions. Subsequent time slots look like they're quicker when the first and final scheduling were done in less time calculations. And its dynamic scheduling methods can be divided into slots at different times as appropriate. The space required building and maintaining your own data center (Data Center), the knowledge of handling a variety of hardware, and the costs associated with it make it easier to use the cloud. In addition, by providing databases in many parts of the world, service providers take care that our processors and services are not compromised, even during natural disasters. This proposed method is designed to allow the crew a few minutes before the scheduling time and a few minutes to get the job done. Thus, buffering time is available. This makes it possible to perform more operations in less time.

REFERENCES

[1] G. Medina Acosta and J. Delgado Penin, "On the Feasibility of a Channel Dependent Scheduling for the SC-FDMA in 3GPP-LTE (Mobile Environment) based on a Prioritized-Bifacet Hungarian Method", *EURASIP Journal on Wireless Communications and Networking*, Vol. 71, pp. 1-10, 2011.

[2] C. Bassa, and Gil-Lafuente, "The Hungarian Algorithm for Specific Customer Needs", *Proceedings of International Conference on Soft Computing in Management and Business Economics*, pp. 363-379, 2012.

[3] X. Ji, C. Shao and B. Wang, "Stochastic Dynamic Traffic Assignment Model under Emergent Incidents", *Proceedia Engineeering*, Vol. 137, pp. 620-629, 2016.

[4] T. Penner, A. Johnson, B. Slyke and Q. Gu, "Transient Clouds: Assignment and Collaborative Execution of Tasks on Mobile Devices", *Proceedings of International Conference on Global Communications*, pp. 2801–2806, 2014.

[5] T. Penner, A. Johnson, B. Slyke and Q. Gu, "Demo: Transient Clouds", *Proceedings of International Conference on Mobile Computing, Applications and Services*, pp. 153-154, 2014.

[6] T. Li, Y. Li and Y. Qian, "Improved Hungarian Algorithm for Assignment Problems of Serial-Parallel Systems", *Journal of Systems Engineering and Electronics*, Vol. 27, No. 4, pp. 858-870, 2016.

[7] N.V. Kousik, M. Sivaram and R. Mahaveerakannan, "Improved Density-Based Learning to Cluster for User Web Log in Data Mining", *Proceedings of International Conference on Inventive Computation and Information Technologies*, pp. 813-830, 2021.

[8] K. Praghash and T. Karthikeyan, "Data Privacy Preservation and Trade-off Balance Between Privacy and Utility using Deep Adaptive Clustering and Elliptic Curve Digital Signature Algorithm", *Wireless Personal Communications*, Vol. 78, 1-16, 2021.

[9] G. Dhiman, A.V. Kumar, R. Nirmalan and S. Sujitha, "Multi-Modal Active Learning with Deep Reinforcement Learning for Target Feature Extraction in Multi-Media Image Processing Applications", *Multimedia Tools and Applications*, Vol. 23, pp. 1-25, 2022.

[10] T. Chatterjee, V.K. Ojha and V. Snasel, "Design and Implementation of a New Datacenter Broker Policy to Improve the QoS of a Cloud", *Proceedings of International Conference on Advances in Intelligent Systems and Computing*, pp. 281–290, 2014.