Cloud Based Task Scheduling Using Fuzzy C-Means and Linear Programming Approach

Mahesh S. Shinde[1], **Anilkumar Kadam**[2]

[1, 2]Department of Computer Engineering, AISSMS COE, Savitribai Phule Pune University, Pune, India.

Abstract: ***Cloud computing is a latest technology that deals with on-demand, online distribution of computing resources and services within managed environment on pay per use scenario. Cloud Service Provider (CSP) manages all the requests of cloud users and allocates the resources (processors) to them. Cloud service provider thus plays crucial role of scheduling the available resources to the user tasks in order to reduce overall cost (cost that end user required to pay to cloud) in user specified time. In the proposed system, cloud resources are scheduled by using linear programming as an optimization technique along with Fuzzy C-Means (FCM) as a clustering technique. Cloud user submits the batch of tasks (images for processing) through smart phone to a cloud service provider. Cloud service provider schedules these tasks among the number of resources (smart phones and processors) using FCM and linear programming. The proposed system schedules these tasks in such a way that it requires minimum cost under the time constraint given by submitter. The results of proposed scheduling strategy are compared with the results obtained by First Come First Served (FCFS) scheduling.***

Keywords: Cloud computing, cloud service provider, linear programming, Fuzzy C-Means, cloud scheduling

1. Introduction

Today, smart phones have become integral part of human life. A wide range of smart phone applications is available to users which have in turn made human life more easy and comfortable. But smart phones have lack of processing power to run some applications. So there is need to use the cloud environment for mitigation of this problem [1].Cloud computing is subscription (pay per use) and web based service where user can acquire resources of cloud for their storage and computational purpose as per requirement [2]. Cloud infrastructure consists of layers like applications, runtime, middleware, operating system and hardware. Services provided by cloud are categorized into three major types namely SaaS (Software as a Service), PaaS (Platform as a Service) and IaaS (Infrastructure as a Service). In SaaS, all the layers are operated by CSP. While in case of PaaS, only applications are managed (developed) by end user and other platform supports are provided by CSP. But in IaaS, CSP only provides hardware components such as processors, storage devices, network devices etc. and all other layers of cloud infrastructure are managed by end user [2][3]. Clouds are categorised into four categories: private cloud (which is managed and used only by individual person or organization), community cloud (managed and accessed by group of organizations for their common purpose), public cloud (which can be accessed by all the worldwide users on pay per use basis) and hybrid cloud (combination of private and public cloud) [2]. Typical cloud environment is as shown in Figure 1.

A scheduler is major component of cloud service provider which performs optimal task scheduling to cloud resources in order to serve the purpose of proper resource utilization, time-cost optimization, energy(power) saving, load balancing etc [4]. Since cloud computing is becoming more and more popular, number of users and their applications (tasks) of using the cloud services increased tremendously. So proper task scheduling among the cloud resources is critical factor in business point of view of cloud service provider. This has made cloud task scheduling as a hot research area in the field of distributed computing.

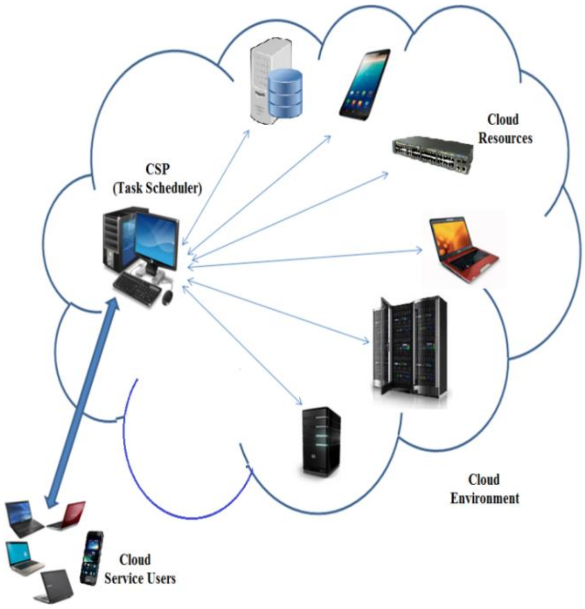

Figure 1: Typical cloud environment

In this paper, we proposed the system in which user submitted tasks are scheduled optimally among the cloud resources using Fuzzy C-Means (FCM) and Linear Programming approach in order to reduce execution cost within minimum (user specified) time. Following section consists of major contributions of some researchers in the area of cloud scheduling.

Remaining of this paper is arranged in following sequence. Section 2 provides some related work in the area of cloud task scheduling. Section 3 gives proposed architecture and description of algorithms used in this architecture. Then section 4 and 5 mentions the experimental results and conclusions that we got from the execution of proposed system. And finally section 6 states further possible enhancements to the proposed system.

Licensed Under Creative Commons Attribution CC BY

2. Related work

Sokol Kosta et al. [1] introduced the new framework called ThinkAir, which enables to migrate and execute the smart phone applications on the cloud in order to solve the problem of lack of sufficient processing power and energy of the smart phones. But problem with this framework is that it does not provide scheduling of user tasks among number of cloud resources since it only decides whether the task to be executed on smart phone (on user side) or it should be sent on cloud for execution. So the cost and time minimization problem is not addressed.

Zixue Cheng et al. [5] introduced a three layered framework for code (applications) migration from wearable devices in first layer to the smart phones in second layer and to the cloud in third layer. This framework mitigates the problem of insufficient processing power for running the application at wearable devices. But this system does not address the issue of cost and time optimization.

Ning Liu et al. [6] presented a framework for energy saving by scheduling the users tasks among the cloud resources according to their efficiency order so that the number of resources used kept as minimum as possible.

Sivadon Chaisiri et al. [7] proposed an optimal scheduling algorithm which handles the tradeoffs between the resource reservation (which is cheaper but requires the future prediction about resource requirements) and on-demand resource allocation (which is costlier) strategies of resource provisioning to the cloud user. This algorithm considers the future resource requirements and its cost. The main purpose of this scheduling algorithm is to minimize the resource provision cost.

Jignesh Lakhani et al. [8] introduced system which minimizes the communication overhead by sending inter dependent user tasks in a group to the particular cloud resource instead of sending them individually.

R. Vijayalakshmi et al. [9] proposed the approach in which tasks submitted by cloud user are scheduled according to their priority. In this approach, task with higher priority is scheduled to the resource having higher computing power. This system minimizes the time required to execute user submitted tasks but cost minimization issue is not addressed.

Shivani Dubey et al. [10] presented cluster based TANH algorithm (Task duplication based scheduling Algorithm for Network of Heterogeneous systems) which groups the inter-dependent tasks into a single cluster so that intercommunication time between them is reduced. But again this algorithm does not minimize the execution cost of user s tasks.

AV. Karthick et al. [11] proposed a Tri Queue Scheduling (TQS) algorithm which groups the tasks into three queues namely small, medium and large queue based on their sizes and then resources are scheduled by using dynamic time slice. This system improves the cloud resource utilization and performance. Thus this system minimizes the investment cost by reducing number of processors by proper utilization. But this scheduling system does not reduce the charges (cost) of execution of user tasks.

So no one of above system minimizes the overall execution cost of tasks i.e. cost which user requires to pay to CSP on pay per use basis where each resource demands different cost. So it is required to satisfy cloud user by executing his tasks with minimum cost and within user specified time bound by scheduling these tasks properly among heterogeneous cloud resources. Therefore in order to provide the execution of user tasks with minimum cost and within user specified time, we proposed the Fuzzy C-Means and Linear Programming approach.

3. Proposed Framework

In this paper, we proposed the architecture in which user submitted tasks are scheduled among number of cloud resources (or processors) using Fuzzy C-Means (as a clustering technique) and Linear programming (as an optimization technique). In our system, we have taken batch image processing as a user task (application) which is submitted by the user through smart phone [1][12]. In this system, we also want to utilize the processing power of underutilized smart phones. So we use these smart phones along with computers as cloud resources which are used by CSP for executing cloud users task (batch image processing).

In this proposed architecture, we are using two algorithms: Fuzzy C-Means algorithm and Linear Programming approach.

3.1 Fuzzy C-Means (FCM) Algorithm

Fuzzy C-Means (or soft clustering) is a method of clustering in which one data point can be present in more than one cluster. Each data element is associated with a set of membership levels where each membership level (between 0 and 1) indicates the amount of belongingness of that element to the particular cluster. Point on the edge of a cluster has lesser membership value for that cluster than the point in the middle of that cluster. Fuzzy clustering is the method of giving these membership values which are then used to assign data elements to one or more clusters. Fuzzy C-Means (FCM) is widely used fuzzy clustering algorithm.

Suppose there are m elements, $X = \{x_1, x_2,\ldots.. x_m\}$ to be partitioned using FCM in K clusters, $C = \{C_1, C_2,\ldots.. C_k\}$ then

1. Decide how many number of final clusters that we want
2. Randomly assign the membership value w_{ij} (the degree to which point x_i belongs to cluster C_j)
   Where, $i = \{1, 2, \ldots..,m\}$ and $j = \{1, 2,\ldots..,k\}$
3. Repeat until the algorithm has converged (when the change in the membership values between two iteration is not more than threshold, ϵ)
   a) Calculate centroid for every cluster using following formula,

$$c_j = \frac{\sum_{i=1}^{m} w_{ij}^{p}\, x_i}{\sum_{i=1}^{m} w_{ij}^{p}} \qquad (1)$$

Where,
c_j = centroid of the j^{th} cluster and
P = level of fuzziness (exponent which determines the influence of membership value).

b) Find membership value for each point using following formula,

$$w_{ij} = \frac{\left({}^{1}\!/_{dist(x_i, c_j)^2} \right)^{\frac{1}{p-1}}}{\sum_{q=1}^{k} \left({}^{1}\!/_{dist(x_i, c_q)^2} \right)^{\frac{1}{p-1}}} \qquad (2)$$

Membership value w_{ij} of any point in the cluster is the reciprocal of square of the distance between that point and centroid of that cluster divided by summation of all membership values of that point (i.e. the reciprocal of square of the distance between that point and centroids of all clusters) [13][14]

3.2 Linear Programming

Linear Programming is a well known optimization technique generally used in business in order to maximize the profit and minimize the cost under some constraints. For applying the linear programming, we have to represent the system in the form which consists of an objective function (that we want to maximize or minimize) and set of constraints formulated by linear inequalities and equalities. Linear problem can be represented by using two forms: standard form (where problem constraints are either equality or inequality equations) and slack form (where all the problem constraints are equality constraints). Linear problem can be represented in standard form as follows [15]:

Objective function:
Maximize (or Minimize),

$$z = \sum_{j=1}^{n} c_j x_j \qquad (3)$$

Subject to:

$$\sum_{j=1}^{n} a_{ij} x_j \le b_i \qquad (4)$$

for i = 1, 2, 3, …..,m

Non-negativity constraints:

$$x_j \ge 0 \qquad (5)$$

for j = 1, 2, 3, ……,n
where,
c_j, a_{ij} and b_i are set of real numbers and x_j are set of variables.

Our aim is to find the values of x_j in such a way that objective function and constraints must be satisfied. In this proposed system, we are using simplex algorithm to solve the linear problem. General Simplex algorithm is as follows [16]:

Step 1: First convert linear programming problem to a system of linear equations (standard form).
Step 2: Convert linear program into slack form by adding slack variables to inequality constraints in order to convert them into equality constraints.
Step 3: Find a basic solution by setting all the non-basic variables (variables on the right hand side in the equalities in slack form) to 0 and then computing the values of all (basic) variables on the left hand side.
Step 4: Iteratively rewrite the set of equations and reconstruct linear equations so that the basic solution changes and has a greater objective value than previous iteration. Following steps are followed for this purpose:

1) Choose a non-basic variable whose coefficient in objective function is positive.
2) Increase value of this selected non-basic variable as much as possible until no one constraint is violated.
3) Thus this non-basic variable is changed to basic variable and some other existing basic variable to non-basic variable.

Step 5: Repeat steps 3 and 4 until all coefficients in the objective function become negative.
Step 6: Now put the final basic solution in original linear program to get the required objective value (i.e. values of x_j in the objective function).

3.3 System Architecture

We proposed the task scheduling using Fuzzy C-Means (FCM) and Linear Programming as shown in Figure 2. In the proposed system, cloud user submits the batch of images for processing to the cloud service provider (CSP) or cloud task scheduler. Task scheduler (within the cloud service provider) uses two algorithms: Fuzzy C-Means and Linear Programming.

As shown in system architecture in Figure 2, user input i.e. batch of images is given to FCM. Now FCM clusters this batch of images into the three clusters: small, medium and large based on the size of images. Generally processing power of computers (or CPU) is greater than the processing power of smart phones. Therefore large images and small images are directly given to computers and smart phones in the cloud resources respectively. While the medium images (images that can be both small and large) are given to the linear programming algorithm so that to schedule them properly among the computers and smart phones in order to reduce the overall cost of processing (i.e. cost user have to pay to cloud resources). But while doing this scheduling, time constraint given by the cloud user (task submitter) should not be violated.

Licensed Under Creative Commons Attribution CC BY

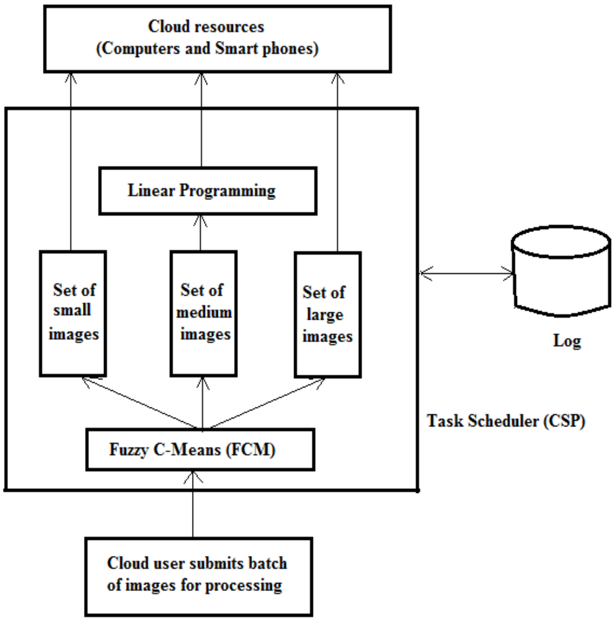

Figure 2: System architecture of proposed task scheduling

Now while applying linear programming, first we have to represent this scheduling problem into the linear programming equations. The linear programming equations for our system are:

Objective function:

Minimize,

$$z = \left(C_{Smart\ phone} + C_{Computer}\right) \quad (6)$$

Subject to:

$$T_{Smart\ phone} + T_{Computer} \leq T_{User} \quad (7)$$

$$I_{Smart\ phone} + I_{Computer} = I_{Total} \quad (8)$$

Non-negativity constraints:

$$I_{Smart\ phone},\ I_{Computer} \geq 0 \quad (9)$$

Where,

$C_{Smart\ phone}$ = Total cost charged by smart phone resources.
$C_{Computer}$ = Total cost charged by computer resources.
$T_{Smart\ phone}$ = Total time required for smart phones for execution of the images scheduled for them.
$T_{Computer}$ = Total time required for computers for execution of the images scheduled for them.
T_{User} = User submitted time within which execution of his tasks must be completed
$I_{Smart\ phone}$ = Total number of images processed by smart phone resources
$I_{Computer}$ = Total number of images processed by computer resources
I_{Total} = total number of images submitted by user

Equation 6 represent a objective function in terms of execution cost that we want to minimize. Thus goal is to minimize the total cost as represented in objective function under the time constraint provided by user as shown in equation 7. While equation 8 states that all the user submitted tasks (images) must be processed.

4. Experimental Results

The results of the proposed system are calculated in terms of cost required for execution of different sets of images (cloudlets). Each resource (computer and smart phone) processes the images by demanding specific price per pixel. In order to testing the performance of the proposed system, it is assumed that smart phones and computers demands different cost per pixel. In this experiment, each smart phone processes images by taking 5 units of cost per 1000 pixels from user. While each computer resource demands 10 units of cost per 1000 pixels for the processing of images scheduled to it. The results in terms of cost of proposed scheduling are compared with the results obtained by First Come First Served (FCFS) scheduling by giving same set of images and with the same setup. Therefore as shown in table 1, same five sets of images (cloudlets) are used to compare the results in terms of cost required for execution.

Table 1: Cost comparison of proposed and FCFS scheduling

Sr. No.	*Number of cloudlets*	*Cost required in FCFS scheduling*	*Cost required in proposed scheduling*
1	3	3290	2910
2	6	6720	5135
3	9	8730	7960
4	12	10940	8820
5	15	15280	12835

Cost graph of proposed and FCFS scheduling is as shown in figure 3.

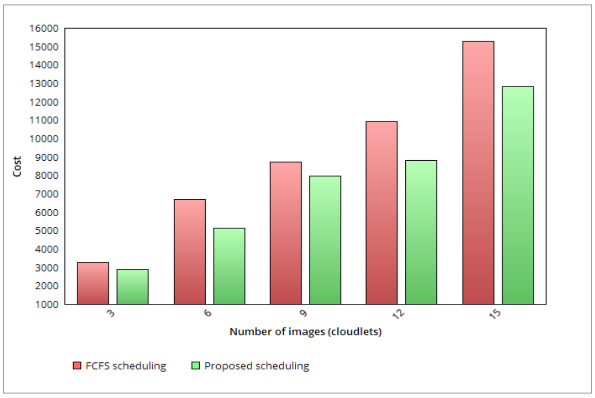

Figure 3: Cost comparison with FCFS scheduling

It shows that the cost required to execute number of tasks (image processing task) for proposed system is less than the First Come First Served (FCFS) scheduling.

5. Conclusion

Task scheduling plays vital role in managing and sharing cloud resources among the different tasks of cloud user. An efficient task scheduling policy provides proper resource utilization, load balancing and optimization of execution cost and time. Therefore today task scheduling is main research topic in the area of cloud computing. In this paper, we have proposed the task scheduling using Fuzzy C-Means (FCM) and Linear Programming approach. Main objective of this system is to schedule the user submitted tasks among the cloud resources in such a way that it minimizes the cost that user (task submitter) required to pay to the cloud owner without exceeding execution time limit (constraint) provided by him. We can conclude from obtained results that by using well known linear programming approach as optimization technique along with Fuzzy C-Means as clustering

Licensed Under Creative Commons Attribution CC BY

technique, the objective of cost and time optimization is achieved.

6. Future Enhancements

The proposed system has successfully achieved its objectives of execution time and cost optimization. While applying linear programming, this proposed system can be improved by considering more number of parameters like resource utilization, load balancing, power consumption etc.

References

[1] Sokol Kosta, Andrius Aucinas, Pan Hui, Richard Mortier and Xinwen Zhang, “ThinkAir: Dynamic resource allocation and parallel execution in the cloud for mobile code offloading”, Proceedings IEEE INFOCOM, IEEE 978-1-4673-0775-8/12, PP: 945-953, 2012.

[2] Borko Furht and Armando J. Escalante, Handbook of Cloud Computing, ISBN 978-1-4419-6523-3, Springer, 2010.

[3] Vaishali H Pardeshi, “Cloud Computing for Higher Education Institutes: Architecture, Strategy and Recommendations for Effective Adaptation”, Symbiosis Institute of Management Studies Annual Research Conference (SIMSARC13), Elsevier, 2014.

[4] Mahesh S. Shinde and Anilkumar Kadam, “Survey of Recent Task Scheduling strategies in Cloud Computing”, International Journal of Advanced Research in Computer Science and Software Engineering,Volume 4, Issue 11, November 2014.

[5] Zixue Cheng, Peng Li, Junbo Wang and Song Guo, “Just-in-Time Code Offloading for Wearable Computing”, IEEE Transactions on Emerging Topics in Computing, 2015.

[6] Ning Liu, Ziqian Dong and Roberto Rojas-Cessa, “Task Scheduling and Server Provisioning for Energy-Effcient Cloud-Computing Data Centers”, IEEE 33rd International Conference on Distributed Computing Systems Workshops, IEEE 978-0-7695-5023-7/13, 2013, PP: 226-231.

[7] Sivadon Chaisiri, Bu-Sung Lee and Dusit Niyato, “Optimization of Resource Provisioning Cost in Cloud Computing”, IEEE TRANSACTIONS ON SERVICES COMPUTING, IEEE 1939-1374/12, PP: 164-177, 2012.

[8] Jignesh Lakhani and Hitesh A. Bheda, “An Approach to Optimized Resource Scheduling using Task Grouping in Cloud”, International Journal of Advanced Research in Computer Science and Software Engineering, ISSN:2277 128X, Volume 3, Issue 9, September 2013, PP: 594-599.

[9] R. Vijayalakshmi and Mrs. Soma Prathibha, “A novel approach for task scheduling in cloud”, 4th ICCCNT, IEEE-31661, July 4-6, 2013.

[10] Shivani Dubey, Vismay Jain and Shailendra Shrivastava, “An Innovative Approach for Scheduling of Tasks in Cloud Environment”, IEEE 4th ICCCNT, 2013.

[11] AV. Karthick, Dr. E. Ramaraj and R. Kannan, “An Efficient Tri Queue Job Scheduling using Dynamic Quantum Time for Cloud Environment”, International Conference on Green Computing, Communication and Conservation of Energy (ICGCE), IEEE 978-1-4673-6126-2/13, 2013.

[12] Majid Altamimi, Atef Abdrabou, Kshirasagar Naik and Amiya Nayak, “Energy Cost Models of Smart phones for Task Offloading to the Cloud”, IEEE Transaction on Emerging Topics in Computing, 2015.

[13] Paan-Ning Tan, Michael Steinbach and Vipin Kumar, Introduction to Data Mining, Pearson Education, ISBN 978-81-317-1472-0, 2013.

[14] YIN Zhonghang, TANG Yuangang, SUN Fuchun and SUN Zengqi, “Fuzzy Clustering with Novel Separable Criterion”, TSINGHUA SCIENCE AND TECHNOLOGY, ISSN 1007-0214, Volume 11, Number 1, PP: 5053, February 2006.

[15] John W. Chinneck, Practical Optimization: A Gentle Introduction, [Online] Available at: http://www.sce.carleton.ca/faculty/chinneck/po.html, 2001.

[16] Coremen, Leinserson and Rivest, Algorithms, PHI.

Author Profile

Mahesh S. Shinde, received the BE degree in Computer Engineering from PREC, Loni. He is now student of ME at AISSMS COE, Pune. His area of interest includes cloud computing and current trends and techniques in distributed systems.

Anilkumar Kadam, received the BE degree in Computer Engineering from SGGSE&T, Nanded and M.TECH degree in Computer Engineering from COEP, Pune. He has 13 years of teaching experience and 2 years of industrial experience. He is currently working as assistant professor at AISSMS COE, Pune. His area of interest includes parallel processing and distributed computing.

Licensed Under Creative Commons Attribution CC BY