

ONLINE CAPABILITY RESERVE UNDER ROUNDED COST FOR CLOUD COMPUTING

¹M. Chandana, ²M. Anil kumar, ³A. Geetha
^{1,2,3}Visvesvaraya College of Engg. & Tech.,M.P Patelguda, Ibrahimpatnam

Abstract— online concave pricing have been developed computational resources are readily and elastically available to the customers. In order to attract customers with various demands, most Infrastructure-as-a-service (IaaS) cloud service providers offer several pricing strategies such as pay as you go, pay less per unit when you use more (so called volume discount), and pay even less when you reserve. The diverse pricing schemes among different IaaS service providers or even in the same provider form a complex economic landscape that nurtures the market of cloud brokers. By strategically scheduling multiple customers' resource requests, a cloud broker can fully take advantage of the discounts offered by cloud service providers. In this paper, we focus on how a broker can help a group of customers to fully utilize the volume discount pricing strategy offered by cloud service providers through cost-efficient online resource scheduling. We present a randomized online stack-centric scheduling algorithm (ROSA) and theoretically prove the lower bound of its competitive ratio. Three special cases of the offline concave cost scheduling problem and the corresponding optimal algorithms are introduced. Our simulation shows that ROSA achieves a competitive ratio close to the theoretical lower bound under these special cases. Trace-driven simulation using Google cluster data demonstrates that ROSA is superior to the conventional online scheduling algorithms in terms of cost saving.

Keywords— Cloud Computing, ROSA, Online Scheduling.

I. Introduction (Cloud Computing)

Parallel computing is a type of computation in which many calculations are carried out simultaneously, operating on the principle that large problems can often be divided into smaller ones, which are then solved at the same time. There are several different forms of parallel computing: bit-level, level, data, and task parallelism. Parallelism has been employed for many years, mainly in high-performance computing, but interest in it has grown lately due to the physical constraints preventing frequency scaling.

Cloud Computing Benefits:

- Redundant data storage - even if one hard disk fails your data is safe and will continue to be available automatically on another one;
- Dedicated resources - the latest cloud computing technology ensures that your resources are dedicated to you and not shared to others. This includes RAM and CPU according to the plan you choose.
- No single point of failure - each part of the service is ensured to be redundant and distributed. This means that there are no hardware problems, nor software problems from the underlying technology.
- Resource usage efficiency - best options to scale to your usage and naturally pay only what you use.
- Security - each cloud user can be reliably isolated in his / her own environment.

Online Scheduling:

A system of planning, placement and arrangement of appointments, issues and meetings via an online platform.

The Features of the Online Scheduling System:

- Online Calendar. Let your clients' book online 24/7 with no intervention even when your business is closed. ...
- Simultaneous Appointments. You can receive reservations from several people at the same time slot with the shifts configuration. ...
- Prepayment

Infrastructure-as-a-Service (IaaS):

The infrastructure services are divided into three sublevels. One level comprises of providers of simple co-location (facilities) services. The next is the hardware that includes memory, computation and storage facility. The next is the virtualization which includes the provisioning, virtualization and billing of the services. Amazon Elastic Compute Cloud (EC2) is an example of Infrastructure-as-a-Service (IaaS).

II. Randomized Online Stack-Centric Scheduling Algorithm (ROSA):

Real time cloud computing environment has a great impact among the Infrastructure as a service Providers. There are various services which are being offered for various kinds of requirements for the customers. Cloud broker would be very much engaged in providing profitable outcomes for both the Cloud service provider and the end user. There are two types of multiplexing which are being followed for providing pricing discounts to the customer. There are various scheduling algorithms for performing scheduling of resources. In our paper we propose randomized scheduling algorithm (ROSA) which is having its efficient significance

in scheduling and allocating the resources effectively. We deploy our cloud environment with Cloud Simulator tool and we use Net beans IDE and Xampp server for implementing ROSA algorithm. We can be able to prove that ROSA would be superior to the various conventional scheduling algorithms in terms of reduction of cost. Keywords: Cloud computing, concave pricing, ROSA, Spatial multiplexing, temporal multiplexing.

Problem Definition:

Online concave pricing Minimization with a concave cost function usually falls into the class of NP-hard problems, for example, the concave network flow problem. This partially suggests the hardness of our scheduling problem. Though we have not formally proved its NP harness, we have discovered the properties of optimal scheduling with a general concave cost function. These properties provide us with valuable insights on making cost-client decisions in online and online resource scheduling. Furthermore, these properties have inspired us to find an optimal online scheduling algorithm for a special concave cost function. In this section, we present the properties that an optimal schedule should have and point out why it is hard to come up with an optimal scheduling algorithm with polynomial complexity.

Proposed Solution:

Online concave pricing Google cluster data is suitable for evaluating the proposed scheduling algorithm as it provides job requests with a large variety of resource requirement. The job requests submitted by different user’s exhibit different patterns in term of inter-arrival time and job length. At any time instance, the number of scheduled jobs is limited by a constant. All jobs can be processed in a unit time interval. All jobs can be preempted only at integer time points. They showed that the problem can be solved online using linear programming. In this section, constraint is relaxed. We assume Jobs can be preempted at any time points.

Experimental Evaluation

In our proposed system, for deploying a cloud environment, we have used Cloud Simulator tool. We deploy a GUI application for cloud environment by using Java language in Net beans IDE. For the back end connectivity, we have deployed Xampp server. We have developed the application for processing of user task request. In our GUI application process, we have taken user file download and user file upload as two main functions which user would perform.

In the process, for multiple users job request, randomized scheduling with delay time algorithm is carried

out in which job with the maximum workload is being processed first. We have partially implemented our proposed system. Further implementation is in progress. By this, we would be able to determine the processing time as well as the waiting time of the Job by which we can prove that our proposed randomized algorithms are superior to the traditional algorithms.

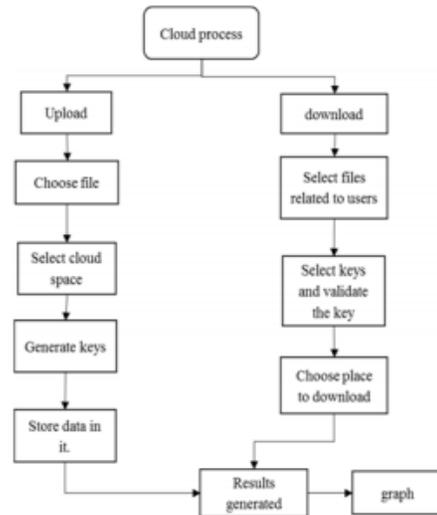


Fig 1: Data flow diagram of user Tasks

Advantages of Proposed System:

1. They showed that the problem can be solved online using linear programming
2. Broker can schedule the jobs of users to leverage the pricing model with volume discounts so that the maximum cost saving can be achieved for its customers.
3. An optimal online algorithm which schedules jobs greedily and sequentially is introduced.

Motivation:

While there are many based on cloud computing only now days it’s a big market. PROVIDE CLOUD TO A NORMAL CUSTOMER WITHOUT A RISK then the broker can help to then to providing fewer amounts of data. In reality, there are many different types of attackers with different reasons to attack users. The following contains some examples to steal valuable data with access to valuable data, (ROSA) they can then generate revenue.

Feasibility Study

Feasibility study is an important phase in the software development process. enables the developer to have an assessment of the product being developed. It refers to the feasibility study of the product in terms of outcomes of the product, operational use and technical support required for

implementing it. Feasibility study should be performed on the basis of various criteria and parameters.

The various feasibility studies are:

- Operational feasibility
- Technical feasibility
- Economic feasibility
- Feasibility Report

Operational Feasibility:

This examination is finished to check the specific achievability, that is, the particular necessities of the structure. Any system made must not have an interest on the available particular resources. This will incite levels of prominence on the available specific resources. This will provoke levels of prominence being determined to the client. The made structure must have an unassuming essential, as simply irrelevant or invalid changes are required for realizing this system.

Technical Feasibility:

The specialized issue typically raised amid the plausibility phase of the examination incorporates the accompanying:

- Does the important innovation exist to do what is recommended?
- Do the proposed types of gear have the specialized ability to hold the information required to utilize the new framework?
- Will the proposed framework give sufficient reaction to request, paying little mind to the number or area of clients?
- Can the framework be overhauled if created?
- Are there specialized certifications of exactness, unwavering quality, simple entry and information security?

Prior no framework existed to meet the patron desires with less postponement. The present framework created is in fact practicable. In this way it gives a simple get right of entry to to the customers. In this way, it gives the specialised certification of precision, unwavering quality and protection. The product and hard necessities for the advancement of this assignment are very few and are as of now reachable in-residence at NIC or accessible as loose as open source. The paintings for the assignment are completed with the present hardware and current programming innovation. Fundamental switch velocity exists for giving a short input to the clients no matter the amount of clients utilizing the framework.

Economic Feasibility:

This examination is done to check the budgetary impact that the system will have on the affiliation. The measure of store that the association can fill the inventive work of the structure is limited. The utilizations must be upheld. Thusly the made system too inside the monetary arrangement and this was expert in light of the way that a huge segment of the progressions used are straightforwardly open. Simply the adjusted things must be gained.

Module Descriptions:

CSP: Admin Scheduling the cloud offers with prices given to request brokers. And admin view the both users and customers’.

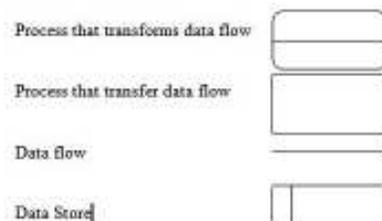
BROKER: Broker sends request TO CSP and get the data and he schedule the cloud data as per customer view. Given fewer amounts of data to customers according to their scheduling.

USER: Customer send the request to a particular broker schedule then the cloud data .he can upload the files to their cloud storage and they can see the status of the cloud.

DFD Symbols:

In the DFD, there are four symbols

1. A square defines a source (originally) or destination of system data.
2. An arrow identifies data flow. It is the pipe line through which the information flows.
3. A circle or a bubble represents a process that transforms income data flow into outgoing data flows.
4. An open rectangle is a data store, data at rest or a temporary repository of data.



The development of DFD’S is done in several levels. Each process on lower level diagrams can be broken down into a more detailed DFD in the next level.

III.Unified Modeling Language (Uml) Diagrams

The Unified Modeling Language (UML) is a Standard dialect for determining, imagining, developing and recording the product framework and its parts. The UML concentrates on the theoretical and physical portrayal of the framework. It catches the choices and understandings about frameworks that must be developed.

Class Diagram:

A “Class Diagram” shows a set of classes, interfaces and collaborations and their relationships. These diagrams are most common diagram in modeling object oriented systems. Class diagrams are the backbone of almost every object – oriented methods, including UML. They describe the static structure of a system. An object class describes a group of objects with similar properties (attributes), common behavior (operations), common relationships to the other objects, and common semantics. The Abbreviation Class is often used instead of Object Class. Project manager in a class have same attributes and behavior patterns. Most objects derive their individuality from differences in their attribute values and relationships to other objects. The Different classes identified in the system are the Data Identification, and Data Operation

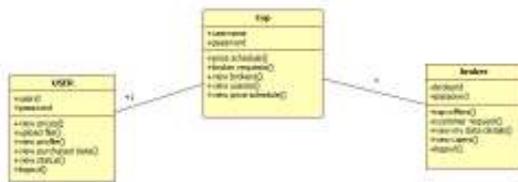


Fig 2: Class Diagram

Use Case Diagram:

Use case diagrams are one of the five diagrams in the UML for modeling the dynamic aspects of the systems (activity diagrams, sequence diagram, state chart diagram, collaboration diagram are the four other kinds of diagrams in the UML for modeling the dynamic aspects of systems). use case diagram are central to modeling the behavior of the system, a sub-system, or a class. Each one shows a set of use cases and actors and relations. A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use case analysis.

Actor

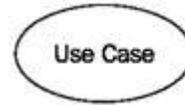
You can picture an on-screen character as a client of the IT framework, for instance Mr. Steel or Mrs. Smith from registration. Since singular people are unessential for the model, they are dreamy. So the on-screen characters are called "registration worker" or "traveler"



(Fig. 4.5 Representation of Actor)

Use Case

Use cases describe the interactions that take place between actors and IT systems during the execution of business processes:



(Fig. 4.6: Representation of Use case)

A use case represents part of the functionality of the IT device and allows the consumer (modeled as an actor) to get entry to this capability

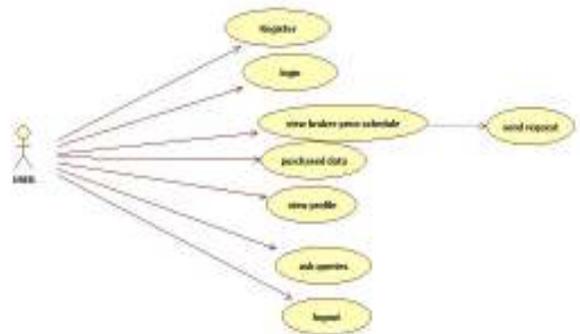


Fig 3: Use Case Diagram

Activity Diagram:

Activity diagrams describe the workflow behavior of a system. Activity diagrams are similar to state diagrams because activities are the state of doing something. The diagrams describe the state of activities by showing the sequence of activities performed. Activity diagrams can show activities that are conditional or parallel. Activity diagrams should be used in conjunction with other modeling techniques such as interaction diagrams and state diagrams. The main reason to use activity diagrams is to model the workflow behind the system being designed. Activity Diagrams are also useful for: analyzing a use case by describing what actions needs to take place and when they should occur; describing a complicated sequential algorithm and modeling applications with parallel processes.

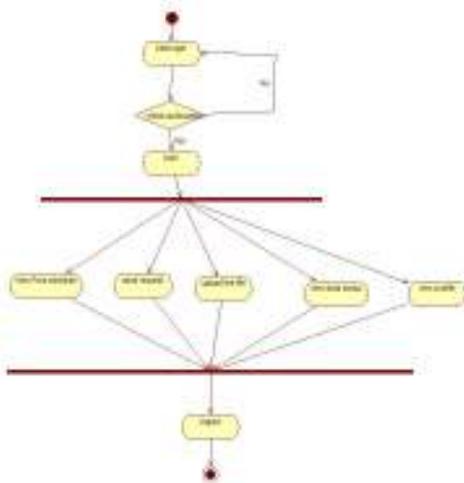


Fig 4: Activity Diagram

Collaboration diagram:

Joint effort Diagrams depict communications among classes and affiliations. Here, in this task the cooperation graph contains the stream that client can make a record subsequent to making the record the administrator will transfer the connections then client will look for joins then administrator will to see the client points of interest and give the customized joins from nearby archive then client has chance to see the connections content through web and send his customized connects to neighborhood storehouse at that point recover the documents finally administrator will see the client subtle elements to know either that record is sure or negative and transfer records as indicated by that necessity.

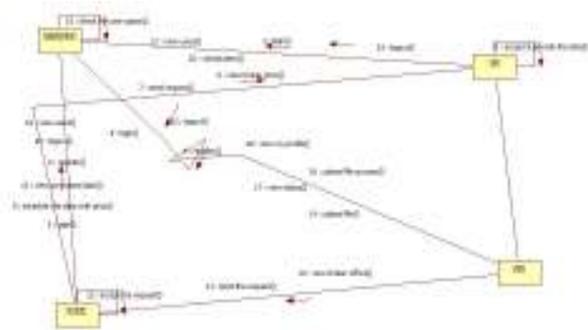


Fig 5: Collaboration Diagram

Sequence Diagram: Sequence diagram is an interaction diagram which is makes a speciality of the time ordering of messages. It indicates a set of objects and messages exchanged among these items. This diagram illustrates the dynamic view of a gadget. Sequence diagrams belong to a collection of UML diagrams referred to as Interaction Diagrams. Sequence diagrams describe how gadgets have interaction over the direction of time through an exchange of messages. A single collection diagram regularly

represents the go with the flow of occasions for a single use case.

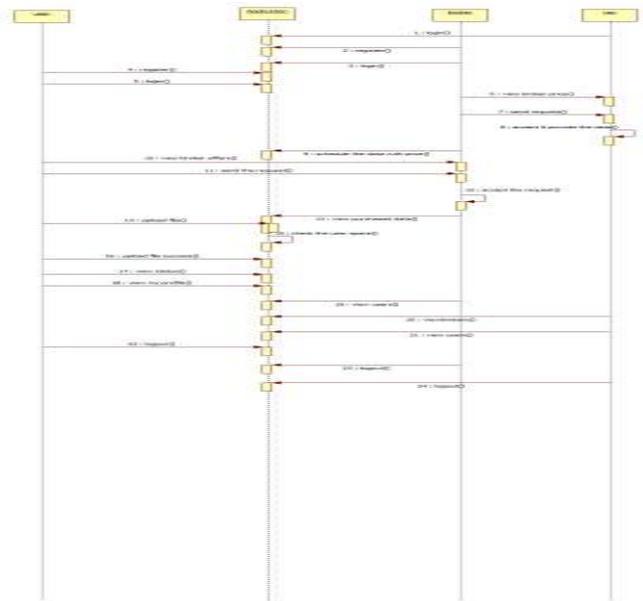
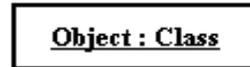


Fig 6: Sequence Diagram

Class roles

Class roles describe the manner an object will behave in context. Use the UML item image to illustrate class roles, but don't list object attributes.



(Fig: Class roles diagram)

Activation

Activation boxes represent the time an object needs complete a task.

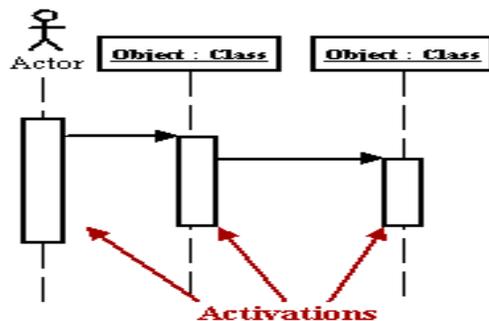


Fig 7: Activation diagram

Messages

Messages are arrows that represent communication between gadgets. Use half-of-arrowed traces to symbolize asynchronous messages. Asynchronous messages are sent from an item that will now not await a response from the receiver earlier than persevering with its responsibilities.

Arrow	Message type
	Simple
	Synchronous
	Asynchronous
	Balking
	Time out

Loops

A repetition or loop within a chain diagram is depicted as a rectangle. Place the condition for exiting the loop at the bottom left corner in square brackets [].

Relationships:

Association

An affiliation is a connection between an actor and a use case. An association suggests that an actor can carry out a use case. Several actors at one use case imply that each actor can perform the use case on his or her own and no longer that the actors carry out the use case together:

According to UML, association only means that an actor is involved in a use case. Use associations in a restricted manner.

Testing:

Testing is a process of executing a program with the intent of finding an error. Testing is a crucial element of software quality assurance and presents ultimate review of specification, design and coding. System testing is an important phase. Testing represents an interesting anomaly for the software .thus a series of testing are performed for the proposed system before the system is ready for user acceptance testing. A good test case is one that has a high probability of finding an as undiscovered error. A successful test is one that uncovers an as undiscovered error.

Testing Objectives:

- Testing is a process of executing a program with the intent of finding an error.
- A good test case is one that has a high probability of finding an as yet discovered error.

- A successful test is one that uncovers an as yet undiscovered error.

Testing Principles:

- All tests should be traceable to customer requirements.
- Tests should be planned long before testing begins.
- Testing should begin ‘in the small’ an progress toward testing “in the large”
- Exhaustive testing is not possible.
- To be most effective and independent third party should conduct testing.

Types of Testing Tools and Environment

Black Box Testing:In this strategy some test cases are generated as input conditions that fully execute all functional requirements for the program. This testing has been uses to find errors in the following categories.

- Incorrect or missing functions.
- Interface errors.
- Errors in data structure or external database access
- Performance errors.
- Initialization and termination errors.

In this testing only the output is checked for correctness. The logical flow of the data is not checked.

White Box Testing:

In this test cases are generated on the logic of the each module by drawing flow graphs of that module and logical decisions are tested on all the cases. It has been uses to generate the test cases in the following cases.

- Guarantee that all independent paths have been executed.
- Execute all logical decisions on their true and false sides.
- Execute all loops at their boundaries and within their operational bounds.
- Execute internal data structures to ensure their valid.

IV.Conclusion

Cloud is a rising processing market where cloud suppliers, dealers, and clients share, intercede, and expend processing asset. With the advancement of distributed computing, Pay-as-you-go valuing model has been enhanced with volume rebates to empower the clients' appropriation of cloud processing. This paper concentrates how an agent can plan the employments of clients to influence the estimating model with volume rebates so that the most extreme cost

sparing can be accomplished for its clients. We have examined the properties that an ideal arrangement ought to have and considered three exceptional instances of the curved cost planning issue. We created a web planning calculation and inferred its focused proportion. Recreation results on a Google information follow have demonstrated that the proposed internet planning calculation beats other customary planning calculations.

Future Enhancement:

In this paper we can propose for a FEATURE ENHANCEMENT like we need to give pricing fixed system, for cloud storages then directly a user can communicate with the **cloud Broker** and then directly you can communicate with **Cloud Service Provide and** maintain cost based up on how much data that user wants with that your user money not to be **cloud Broker**

advantage. And we can provide a scheduling system for **Cloud Storage** based up on user need. That time also user can no need to discuss with **Cloud Broker** in this we can provide a **Feature Enhancement.**

References

- [1]. An Online Mechanism for Resource Allocation and Pricing in Clouds.
- [2]. Cost-Deadline Based Task Scheduling in Cloud Computing.
- [3]. Advances in Computing and Communication Engineering.
- [4]. Amazon elastic computes cloud.
- [5]. Web Sites & Books.