A CAPABLE RESOURCE SHARING FOR CLOUD COMPUTING BASED ON DYNAMIC SCHEDULING PRIORITY ALGORITHM

Ms. S. Sriendmapriya, II\textsuperscript{nd} ME
Ms. S. Kayalvili, M.E, Assistant Professor/CSE
Velalar College of Engineering and Technology, Erode, Tamilnadu, India
MAIL ID: sriendmapriya@gmail.com

ABSTRACT: Cloud computing is the long dreamed vision of computing as a utility, users remotely storing their data into the cloud so as to enjoy the on-demand high quality applications and services from a shared pool of configurable computing resources. Collaborative Cloud Computing operating in a large-scale environment involving thousands or millions of resources across disparate geographically distributed areas, it also inherent dynamic entities may enter or leave the system on Cloud environment for resource utilization owned by different parties and partitioned in arbitrary ways rather than a single way of partitioning. The main work is to improve resource utilization based on optimal time period to allocate resources by proposing neural network training and to calculate load factor by using dynamic priority for the nodes based on the scheduling of virtual machines. The scheduling of the VMs to the nodes is depending upon their priority value, which dynamically varies based on their load factor. This dynamic priority concept leads to better utilization of the resources. Priority of a node is assigned depending upon its capacity and load factor. The priority scheduling algorithm strikes the right balance between performance and power efficiency. Once the highest and next highest priority nodes have been identified, and then the scheduling is very quick. It prevents the particular node from overloading by considering the load factor.

Keywords: CCC, efficiency, node, reputed, security, VM.

INTRODUCTION

Cloud computing is the long dreamed vision of computing as a utility, where users can remotely store the data into the cloud so as to enjoy the on-demand high quality applications and services from a shared pool of configurable computing resources. Data outsourcing users can relieve from the burden of local data storage and maintenance. The fact that users no longer have physical possession of the possibly large size of outsourced data makes the data integrity protection in cloud computing a very challenging and excellent task, especially for users with constrained computing resources and computing capabilities.

Cloud computing is the delivery of computing as a service relatively to a product, whereby shared resources, software and information’s are provided to computer and other devices as a measuring a service over a network. Cloud computing allocates computation, software applications, data access and storage resources without requiring cloud users to know the location and other details of the computing infrastructure. End users in cloud application access through a web browser or through a light weight desktop or mobile app while the business software and data are stored on servers at a faraway location. Cloud application providers strive to give the same or better service and performance as if the software programs were installed locally on end-user computers.

Data centers are crucial to provide the large volumes of compute and storage resources needed by today’s Internet businesses including web search, content distribution and social networking. Through cost efficiencies and on demand scaling services, cloud data centers are highly multiplexed in shared environments, with VMs and tasks from multiple tenants coexisting in the same cluster. Since these applications come from unrelated customers, they are largely uncoordinated and mutually untrusting.

Network flexibility must adapt in real-time to changing requirements, while optimizing efficiencies & minimizing cost. Cloud computing is the ability to share and multiplex resources across users. One uncertain resource is the network, as an application’s performance can be significantly impacted by the share of network capacity it receives. Cloud networks are shared in a best-effort manner making it hard for both users.
RELATED WORK

Characterizing workload in production clouds has received much attention in recent years, as both scheduler design and capacity upgrade require a careful understanding of the workload characteristics in terms of arrival rate, requirements, and duration. For example, Mishra et al. have analyzed the workload of a Google compute clustering, and proposes an approach to task classification using k means clustering. Following the same line of research, Chen et al. provided a characterization of Google cluster workload at job-level applying the k-means algorithm. Sharma et al. and Zhang et al. studied the problem of finding accurate workload characterizations through benchmark generation and validation. Recently, Reiss et al. provided a comprehensive analysis of the heterogeneity and dynamism found in Google cluster traces, and found that both machine configurations and workload composition are highly heterogeneous and dynamic over time. They also pointed out the importance of considering workload heterogeneity for designing adaptive schedulers. The goal of these studies was to understand the workload composition in production clouds, rather than using workload characterization for resource allocation and capacity provisioning.

There is a large body of literature on energy-aware dynamic capacity provisioning in data centers. pMapper [11] is a migration-aware workload placement framework for optimizing application performance and power consumption in data centers. It does not consider the cost of turning on and off machines when making provisioning decisions, nor does it consider task arrival rate and scheduling delay objectives. Mistral is a framework that dynamically adjusts VM placement to find a tradeoff between power utilization, demanding an implementation, and unstructured costs. It does not consider the arrival rate of task requests in its formulation. More recently, Ren et al. studied the problem of scheduling heterogeneous batch workload across geographically distributed data centers. Different from our work, they assume that the workload has already been divided into well-defined types. They further assume that every task can be scheduled on any machine, which is not always the case as we shall indicate. While no other previous work has applied task classification to dynamic capacity provisioning problem in a heterogeneous data centers. Thus, we design Harmony as a workload-aware DCP framework that can achieve both higher application performance and efficiency in terms of energy savings.

TRUSTWORTHY RESOURCE SHARING PLATFORM FOR COLLABORATIVE CLOUD COMPUTING

Cloud computing has become customary, which only offers scalable resources over the Internet to customers. Currently clouds like, Amazon’s EC2 which allows the computers to work on their own, Google’s AppEngine, IBM’s Blue-Cloud, and Microsoft’s Azure, are providing various services. Amazon [1] provides a simple storage service (s3) Dropbox [2]. Cloud customers are charged to the actual usage of computing resources, hardware, and increasing in bandwidth.

The demand for scalable resources in some applications has been increased recently. Dropbox currently has the turn of five million cloud users, which had increased three times in number last year. Also, researchers may need to build a virtual lab environment connecting multiple clouds for petascale supercomputing capabilities or for fully utilizing resources. As a matter of fact most desktop systems are underutilized in most organizations; they are idle around 95 percent of the time [3]. Te developments in cloud computing are inevitably leading to a promising future for collaborative cloud computing (CCC), where globally-scattered distributed cloud resources belonging to different organizations or individuals are collectively pooled and used in a cooperative manner to provide services [4], [5].

A CCC platform interconnects physical resources to enable resource sharing, and provides a virtual view of huge resources to customers. This effective virtual organization is very clear to cloud customers. When a cloud provider does not have adequate resources requested by its customers, it identifies and uses the resources from other cloud providing servers.

Importance of Resource and Reputation Management

Collaborative Cloud Computing operates in a large-scale environment involving thousands or millions of resources across intended areas, and it is has essential entities which may enter or leave the system according to resource utilization and availability of changing [4], [6]. This environment makes utilized resource management a non-trivial task. Further, due to the autonomous and separate characteristics of entities in Collaborative Cloud, conflicting nodes supplies different quality of service (QoS) in resource provisioning services. A node may allocate low QoS because of system problems or because it is not willing to provide high QoS in order to save costs. Also, nodes may be attacked by viruses and malicious code programs. This weakness is disclosed in all the cloud platforms built by Google, IBM, and Amazon [7], and security has been recognized as an important factor in networks. Then the resMgt needs reputation management to measure resource provision QoS for guiding resource provider selection [4], [7]. As in eBay and Amazon, a repMgt
system computes each node’s reputation value based on evaluations from others about its performance in order to provide guidance in selecting trustworthy resources.

To ensure the successful usage of Collaborative Cloud, the problems of resMgt and repMgt must be jointly addressed for both efficient and trustworthy resource sharing in three tasks:

1. Efficiently locating required trustworthy resources.
2. Choosing resources from the located options.
3. Fully utilizing the resources in the system while avoiding overloading any node.

Previous Methods and Challenges. The three tasks must be executed in a distributed manner since centralized methods are not suitable for large-scale CCC. However, though many distributed resMgt and repMgt systems for grids have been proposed previously, and cloud resource orchestration [6] has been studied in current years, the two of the issues have been typically addressed separately. Simply building and combining individual resMgt and repMgt systems in CCC will generate duplicate, raising of proscriptive. Additionally previous resMgt and repMgt techniques are not sufficiently efficient or effective in the large-scale and dynamic environment of CCC.

Previous repMgt systems neglect resource heterogeneity by assigning each node one reputation value for providing all of its resources. Cloud user declares that node reputation is multi-faceted and should be differentiated across multiple resources. For example, a person trusts a doctor for giving advice on medical issues but not on financial problems. Likewise, a node which performs well for computing services does not necessarily perform well for storage utilities. Then the previous repMgt systems are not effective enough to provide correct guidance for trustworthy individual resource selection. In task (1), RepMgt needs to rely on resMgt for reputation differentiation across multiple resources.

Previous resMgt approaches only assume a single QoS demand of users as expertise and guarantee. Given a number of resource providers (i.e., servers), the efficiency-oriented resMgt policy would choose the one with the highest available resource, while the security-oriented repMgt policy would choose the one with the highest reputation. The former may lead to a low service success rate while the latter may overload the node with many resource requests. Thus, uncoordinated deployment of repMgt and resMgt will exhibit contradictory behaviors and significantly affect the effectiveness of both, finally leading to degraded overall performance. The results of the single-QoS-demand assumption and contradictory behaviors pose two challenges. First, in task (2), how can we jointly consider multiple QoS demands such as cost, proficiency, and available resources in resource selection? Second, in task (3), how can we enable each node to actively control its reputation and resource supply so that it avoids being overloaded while gaining high reputation and profit?

Our Proposed Method

By identifying and understanding the interdependencies between resMgt and repMgt, we introduce Harmony, a CCC platform with harmoniously integrated resMgt and repMgt. It can achieve improvement in management of resources and reputation across distributed resources in CCC. Unlike the prior resMgt and repMgt methods, Harmony enables the node to locate its desired resources and also find the reputation of resources, so that a client can choose resource providers not only by resource availability but also by the provider’s reputation of providing the resource. In addition, Harmony can deal with the challenges of large scale and dynamism in the complex environment of CCC. The contributions of this work can be summarized as below:

1. Preliminary study on real trace and experimental results. By undertaking the transaction and response of rating data we collected from online trading policies. We found that some sellers have high QoS in providing some merchandise but offer low QoS in others, and buyers tend to buy merchandise from regarding as great vendors. The findings verify the importance of multi-faceted reputation and the drawback of the highest-reputed node selection policy.

2. Integrated multi-faceted resource/reputation management. Relying on a distributed hash table overlay (DHT), Harmony offers multi-faceted reputation evaluation across multiple resources by indexing the resource information and the reputation of each type of resource to the same archiving node. By this way, it enables nodes to quick access the information and reputation of available individual resources.


4. Price-assisted resource/reputation control. In a resource transaction, a resource requester pays a resource provider for its resource. The transactions are conducted in a distributed manner in Harmony. Harmony employs a trading model for resource transactions in resource sharing and leverages the resource price to control each node’s resource use and reputation. The node refines to its resource for maximizing its profit and maintains a high reputation.
with excess existence, in order to exploit resources in the system.

We have conducted extensive trace-driven experiments with PlanetLab [14] and simulations. Experimental results show the superior performance of Harmony in comparison with previous resMgt and repMgt systems, and the effectiveness of its three components. This work is the first to integrate repMgt with resMgt for multi-faceted node reputation evaluation to provide precise guidance for individual resource selection. In addition to CCC, Harmony can also be applied to other areas such as large-scale distributed systems, grids and P2P systems.

The preliminary work of this paper was presented in [10]. We introduced an integrated resource and reputation management system and an efficient and locality-aware resource organising system. This work is the combination the two systems and identified CCC as an example of large-scale distributed systems for the application area of this assembled system. This also has a new system component called *price-assisted resource/reputation control. It also presents new sign analysis of results to prove the necessity of the proposed algorithms. We further extensively evaluate the performance of the system through many experiments.

**PROBLEM STATEMENT**

By giving a number of resources-to cloud users, resMgt selects with one of the available resource, the collateral repMgt policy only deals with high cost. A Collaborative Cloud Computing platform performs with symmetrical resMgt and repMgt. It supervises the resources and reputation on distributed resources in cloud. Harmony permits a set of nodes to locate its required resources and also find the reputation of the centrally suited resources, so that a client can choose cloud resource providers not only by resource availability but also by the provider’s reputation of providing the resource.

The drawback of the highest-reputed node selection policy is uncoordinated deployment of either one resources will exhibit contradictory and significantly reduce the virtue of both lowest reputation for the new nodes. The job distribution in cloud is always steady and very low.

**RESOURCE SELECTION AND PRICING SCHEMES FOR CLOUDS**

Each stage of resource marketing: location, selection and transaction. Below, we describe the motivations of the components.

5.1. Integrated Multi-Faceted Res/Rep Mgt

We first study whether a high-reputed node provides high QoS for every type of resource. There are no resource markets with a reputation system. We thus studied an online merchandise trading platform, where each seller sells a variety of merchandise. The sellers and merchandise are regarded as nodes and resources, respectively. The data analytical results, to a certain extent, reflect the selling and buying behaviors.

Zol is a top online trading platform in China similar to Amazon and eBay. Zol is chosen for market data analysis because neither Amazon nor eBay provides the historical rating record of each transaction. Zol provides the historical reputation record of each transaction, which enables to calculate the reputation for each type of a seller’s merchandise for the multi-faceted reputation study. We collected trace data including 1,562,548 transaction records from Zol covering the period from 9/20/2006 to 6/26/2010. In addition to the overall reputation values of sellers, Zol provides the ratings within (0, 100) for five QoS attributes for each transaction: 1) price, 2) distance, 3) quality, 4) service, and 5) efficiency. The number of transactions versus the seller’s overall reputation from this entire trace data set. The figure shows that clients tend to choose higher-reputed nodes for transactions. Thus, if the resource a node possesses is limited, the highest-reputed nodes can easily become overloaded.

Na is used to denote the set of transactions of low ratings in a QoS attribute a, and Nt to denote the set of transactions of low individual reputation. A higher coherence value for a QoS attribute means that a low rating of this QoS attribute is more likely to lead to low individual reputation. The relationship between each QoS attribute and individual reputation. It shows that the quality, service, and efficiency QoS attributes have high coherence values and hence are the primary factors in the low rating of the individual reputation. Most transactions with low individual reputation may still have relatively high ratings in the price and distance QoS attributes. Therefore, a seller’s individual reputation cannot reflect its QoS for each QoS attribute, which confirms the need to consider multiple QoS attributes in selecting resources. Based on this observation, a multi-QoS-oriented resource selection algorithm that enables clients to choose resources based on their priority considerations of the different QoS attributes.

5.2. Multi-QoS-Oriented Resource Selection and Price-Assisted Control

Simply combining multi-resMgt and repMgt will lead to a few problems. First, resMgt and repMgt always have their own infrastructures. For example, Mercury partitions nodes into groups based on resources for resMgt and PowerTrust builds a P2P system with links connecting interacting nodes for repMgt. Maintaining two infrastructures generates high maintenance overhead. Second, most resMgt approaches are driven by either efficiency or security
through choosing the highest-reputed or highest-capacity node. Hence, a direct combination will lead to contradictory behaviors. Third, since a node refers to the overall reputation in selecting individual resources, it may receive incorrect guidance because a high-overall-reputed node may provide low QoS for individual resources.

With the assumption that there is a single resource bottleneck, an experiment on two node selection policies was conducted: MaxTrust and MaxCap each of which chooses the node with the highest reputation and highest available capacity, respectively. The simulation assumed three types of nodes: altruistic, neutral and egotistic, each of which provides its service successfully with a probability 1, 0.5, and 0.1, respectively. Each node is randomly assigned to one of the three types, and every request has 10 servers that are able to satisfy the request. The utilization of a node is defined as the ratio of its load to its capacity. The maximum utilization value of each node during the experiment was recorded. QoS attributes. Referred to as the 99.9th percentile maximum utilization. A service failure occurs when the selected node is unwilling to provide a service with maximum utilization of different policies. It shows that the efficiency-oriented policy, MaxCap, performs best in controlling node utilization but incurs a high service failure rate; in contrast, the trust-oriented policy, MaxTrust, performs the best in controlling service failure rate but leads to high node utilization. The fundamental reason for these severe problems is the neglect of the interdependencies between repMgt and resMgt; the uncoordinated deployment of either one will exhibit contradictory behaviors and significantly reduce the effectiveness of both. Therefore, a method is needed to jointly consider efficiency and trust in resource selection, and also need a method to avoid overloading nodes with resource requests.

NEURAL NETWORK BASED RESOURCE UTILIZATION SCHEME FOR CLOUDS

To improve resource utilization based on optimal time period to allocate resources propose neural network training and for load factor propose dynamic priority for the nodes based on which the virtual machines are organized.

The dynamic priority concept technique leads to better utilization of the resources and a particular node from being overloaded by the load factor.

6.1. Creation of Cloud Environment
A cloud environment often contains a large number of machines that are connected to a system. A Resource Locator addresses a system on global registry. The resources are sending through the internet to a machine inside the data which enters that processes on request. Clouds are hostile environments in large cloud users which utilize the temporal resource so that the cloud user requests a resource from cloud providers.

Multiple cloud users frequently request large number of services, so they must provide in resources. Cloud computing makes for both users and cloud operators to ask about how network resources are allocated.

CCC operates in a large-scale environment involving thousands or millions of resources across disparate disperse areas, and it is also acts as an entity which may come or leave the system and hence the resource utilization and availability are not ending. It makes efficient resource management (resMgt) an unusual task directly to the autonomous and separate features in Collaborative Cloud Computing, various nodes issues dissimilar quality of service (QoS) in resource provision.

A node may provide low QoS because of system problems or because it is not willing to provide high QoS in order to save costs. A Hash table provides set of values in a system and a node can retrieve the value associated with a given key, Harmony offers multi-faceted cost evaluation across multiple resources by indexing the resource information of each type of resource to the same indexed node. The eligibility nodes access to the information and reputation of available individual resources.

6.3. Multi-Qos-Oriented Resource Selection
A single QoS demand of users, Harmony enables a client to perform resource selection with joint consideration of diverse QoS requirements, with different priorities. The problem is how to consider individual or combined QoS attributes in selection. EBay’s quotation agrees the users to yield a feedback on different perspective such as item description and shipment charges. Harmony uses attributes uniformly.

It requires nodes to give ratings for each QoS attribute. The overall QoS for a resource is in addition to the reputation for a server. The ratings are collected at the directory node of the provided resource on a server. As a result, Harmony depends on a neural network to find out the influence weight of each attribute on the overall and further users attribute consideration priority. A neural network can make use of pattern by giving values which is used to derive meaning from unknown data, to extract

Figure 6.1 Neural Network Based Resource Utilization Scheme for cloud
patterns. It communicates with the resources on how to do tasks based on the data given for training and functions as an authority to the data that are given to examine.

6.4. Dynamic Priority Scheduling
A dynamic scheduling algorithm is proposed with dynamic priority for the nodes based on the virtual machines. This leads to better utilization of the resources. This algorithm strikes the right balance between performance and power efficiency. By any chance if a highest and the next highest priority nodes have been identified the scheduling as fast as possible. It inhibits a particular node from being overloaded by considering the load vector.

```
{Flag=0;
If (P =/0)
P1=max available resource node
If (load vector of P1<0.8)
  Assign VM to P1;
If (P2 is set AND load of P2<0.8) AND
  Swap P1 and P2;
  Assign VM to P2;
Else if
  P2=P1
  P1=current max available resource node
  Assign VM to P1;
}
```

Algorithm Scheduling Priority

6.5. Price Assisted Resource/Reputation Management
In a resource transaction, a requester pays a resource provider for its resource. Harmony employs in resource sharing and leverages the resource price to control each node’s use and reputation. It enables each node to adjust its resource price to maximize its profit and maintain a high reputation while avoiding being overloading resources in the system. A provider normally specifies the price of resources according to the cost. The resource prevents difficulties and encourages nodes to issue QoS. All the spacing in the system is utilized, and is not overloaded. A network reputation can truly reflect in its QoS for offering resources without overloading and by specifying the problem of low cost for the newly joined nodes.

CONCLUSION
To improve resource utilization based on optimal time period to allocate resources, propose neural network training and for load factor, propose dynamic priority for the nodes based on which the virtual machines are arranged. This dynamic priority concept leads to better utilization of the resources. Predominance of a network is assigned contingency upon its capacity and the load factor. Based on the comparison and the results from the experiment show the proposed approach works better than the other existing system in terms of pretence time and node allocation.

REFERENCES