International Journal of Emerging Research in Management & Technology ISSN: 2278-9359 (Volume-6, Issue-9)

Research Article **September 2017**

A Novel Approach for Dynamic Apportion and De-Allocate Resources from the Cloud

K Valli Madhavi*
Department of MCA & GIET, India

CH Kalyani
Department of CSE & CEC, India

S Durga Prasad
Department of ECE & GEC, India

Abstract-

Cloud computing is on demand as it offers dynamic flexible resource allocation for reliable and guaranteed services in pay as-you-use manner to public. In Cloud computing multiple cloud users can request number of cloud services simultaneously. So there must be a provision that all resources are made available to requesting user in efficient manner to satisfy their need. Hence there is no need for getting licenses for individual products. Virtual Machine (VM) technology has been employed for resource provisioning. It is expected that using virtualized environment will reduce the average job response time as well as executes the task according to the availability of resources. Effective and dynamic utilization of the resources in cloud can help to balance the load and avoid situations like slow run of systems.

Keywords- Cloud Computing, Dynamic Resource Allocation, Resource Management, Resource Scheduling, Virtual Machine.

I. INTRODUCTION

Cloud computing is the delivery of computing as a service rather than a product, whereby shared resources, software and information are provided to users over the network. Cloud computing providers deliver application via the Internet, which are accessed from web browser, while the business software and data are stored on servers at a remote location. Cloud computing really is accessing resources and services needed to perform functions with dynamically changing needs. The cloud is a virtualization of resources that maintains and manages itself. Cloud computing nowadays becomes quite popular among a community of cloud users by offering a variety of resources. Cloud computing platforms, such as those provided by Microsoft, Amazon, Google, IBM, and Hewlett-Packard, let developers deploy applications across computers hosted by a central organization. These applications can access a large network of computing resources that are deployed and managed by a cloud computing provider. Developers obtain the advantages of a managed computing platform, without having to commit resources to design, build and maintain the network. Yet, an important problem that must be addressed effectively in the cloud is how to manage QoS and maintain SLA for cloud users that share cloud resources.

In cloud platforms, resource allocation takes place at two levels. First, when an application is uploaded to the cloud, the load balancer assigns the requested instances to physical computers, attempting to balance the computational load of multiple applications across physical computers. Second, when an application receives multiple incoming requests, these requests should be each assigned to a specific application instance to balance the computational load across a set of instances of the same application. For example, Amazon EC2 uses elastic load balancing (ELB) to control how incoming requests are handled. Application designers can direct requests to instances in specific availability zones, to specific instances, or to instances demonstrating the shortest response times. In the following sections a review of existing resource allocation techniques like Topology Aware Resource Allocation, Linear Scheduling and Resource Allocation for parallel data processing is described briefly.

II. CLOUD COMPUTING

A cloud refers to a distinct IT environment that is designed for the purpose of remotely provisioning scalable and measured IT resources. The term originated as a metaphor for the Internet which is, in essence, a network of networks providing remote access to a set of decentralized IT resources. Prior to cloud computing becoming its own formalized IT industry segment.

A. Cloud Computing Essential Characteristics

On-demand self-service: A consumer can order several services to each cloud provider, wherever and whenever, without any human interaction requirement.

Broad network access: Resource Pools and services which are located in geographically distributed cloud providers can be reachable over the network and accessed through standard mechanisms that boosts use by inharmonious thin or thick client platforms (e.g. PCs, workstations, laptops, tablets and mobile phones).

Resource pooling: Providing non-united either physical or virtual resource (e.g. Processing, memory, storage etc.) for the consumers in a way that it appears like a united infinite resource pool. This means that the users have no vision about the certain location of resource because of abstraction techniques. With a higher level of abstraction users may be able to define location of their requested resource (e.g. country, state, or datacenter).

Rapid elasticity: Cloud users can request resources as much as they need and scale rapidly up or down them whenever they want. To the consumer, this capability means resource scalability on demand which can be done automatically in some cases.

Measured service: Cloud systems automatically monitor, control, and report resource usage by different level of abstraction based on resource or capability type (e.g. Processing, memory, storage, balancing services etc.). It would allow making transparent billing services to cloud users and also transparent user accounting for cloud providers.

B. Cloud Delivery Models

A cloud delivery model represents a specific, pre-packaged combination of IT resources offered by a cloud provider. Three common cloud delivery models have become widely established and formalized:

- Infrastructure-as-a-Service (IaaS)
- Platform-as-a-Service (PaaS)
- Software-as-a-Service (SaaS)

Infrastructure-as-a-Service (IaaS)

The IaaS delivery model represents a self-contained IT environment comprised of infrastructure-centric IT resources that can be accessed and managed via cloud service-based interfaces and tools. This environment can include hardware, network, connectivity, operating systems, and other "raw" IT resources. In contrast to traditional hosting or outsourcing environments, with IaaS, IT resources are typically virtualized and packaged into bundles that simplify up-front runtime scaling and customization of the infrastructure.

Platform-as-a-Service (PaaS)

The PaaS delivery model represents a pre-defined "ready-to-use" environment typically comprised of already deployed and configured IT resources. Specifically, PaaS relies on (and is primarily defined by) the usage of a ready-made environment that establishes a set of pre-packaged products and tools used to support the entire delivery lifecycle of custom applications.

Software-as-a-Service (SaaS)

A software program positioned as a shared cloud service and made available as a "product" or generic utility represents the typical profile of a SaaS offering. The SaaS delivery model is typically used to make a reusable cloud service widely available (often commercially) to a range of cloud consumers. An entire marketplace exists around SaaS products that can be leased and used for different purposes and via different terms

C. Cloud Deployment Models

A cloud deployment model represents a specific type of cloud environment, primarily distinguished by ownership, size, and access.

There are four common cloud deployment models:

- Public Clouds
- Community Clouds
- Private Clouds
- Hybrid Clouds

Public Clouds

A public cloud is a publicly accessible cloud environment owned by a third-party cloud provider. The IT resources on public clouds are usually provisioned via the previously described cloud delivery models and are generally offered to cloud consumers at a cost or are commercialized via other avenues.

The cloud provider is responsible for the creation and on-going maintenance of the public cloud and its IT resources. Many of the scenarios and architectures explored in upcoming chapters involve public clouds and the relationship between the providers and consumers of IT resources via public clouds.

Private Clouds

A private cloud is owned by a single organization. Private clouds enable an organization to use cloud computing technology as a means of centralizing access to IT resources by different parts, locations, or departments of the organization. When a private cloud exists as a controlled environment, the problems described in the Risks and Challenges section do not tend to apply.

The use of a private cloud can change how organizational and trust boundaries are defined and applied. The actual administration of a private cloud environment may be carried out by internal or outsourced staff.

Hybrid Clouds

A hybrid cloud is a cloud environment comprised of two or more different cloud deployment models. For example, a cloud consumer may choose to deploy cloud services processing sensitive data to a private cloud and other, less sensitive cloud services to a public cloud. The result of this combination is a hybrid deployment model.

Community Clouds

A community cloud is similar to a public cloud except that its access is limited to a specific community of cloud consumers. The community cloud may be jointly owned by the community members or by a third-party cloud provider that provisions a public cloud with limited access. The member cloud consumers of the community typically share the responsibility for defining and evolving the community cloud.

Membership in the community does not necessarily guarantee access to or control of all the cloud's IT resources. Parties outside the community are generally not granted access unless allowed by the community.

III. VIRTUALIZATION TECHNOLOGY

Virtualization is the process of converting a physical IT resource into a virtual IT resource.
Most types of IT resources can be virtualized, including:

- *Servers* - A physical server can be abstracted into a virtual server.
- *Storage* - A physical storage device can be abstracted into a virtual storage device or a virtual disk.
- *Network* - Physical routers and switches can be abstracted into logical network fabrics, such as VLANs.
- *Power* - A physical UPS and power distribution units can be abstracted into what are commonly referred to as virtual UPSs.

The first step in creating a new virtual server through virtualization software is the allocation of physical IT resources, followed by the installation of an operating system. Virtual servers use their own guest operating systems, which are independent of the operating system in which they were created.

Both the guest operating system and the application software running on the virtual server are unaware of the virtualization process, meaning these virtualized IT resources are installed and executed as if they were running on a separate physical server. This uniformity of execution that allows programs to run on physical systems as they would on virtual systems is a vital characteristic of virtualization. Guest operating systems typically require seamless usage of software products and applications that do not need to be customized, configured, or patched in order to run in a virtualized environment.

Virtualization software runs on a physical server called a *host* or *physical host*, whose underlying hardware is made accessible by the virtualization software. The virtualization software functionality encompasses system services that are specifically related to virtual machine management and not normally found on standard operating systems. This is why this software is sometimes referred to as a virtual machine manager or a virtual machine monitor (VMM), but most commonly known as a *hypervisor*.

IV. HYPERVISOR

A hypervisor or virtual machine monitor (VMM) is computer software, firmware, or hardware that creates and runs virtual machines. A computer on which a hypervisor runs one or more virtual machines is called a host machine, and each virtual machine is called a guest machine. The hypervisor presents the guest operating systems with a virtual operating platform and manages the execution of the guest operating systems. Multiple instances of a variety of operating systems may share the virtualized hardware resources: for example, Linux, Windows, and OS X instances can all run on a single physical x86 machine. This contrasts with operating-system level virtualization, where all instances (usually called containers) must share a single kernel, though the guest operating systems can differ in user space, such as different Linux distributions with the same kernel. The term hypervisor is a variant of supervisor, a traditional term for the kernel of an operating system: the hypervisor is the supervisor of the supervisor with hyper- used as a stronger variant of super. A hypervisor is a function which abstracts -- isolates -- operating systems and applications from the underlying computer hardware. This abstraction allows the underlying host machine hardware to independently operate one or more virtual machines as guests, allowing multiple guest VMs to effectively share the system's physical compute resources, such as processor cycles, memory space, network bandwidth and so on. A hypervisor is sometimes also called a virtual machine monitor.

V. RESOURCE MANAGEMENT

From our perspective, research management includes resource discovery, allocation and monitoring process. These processes manage physical resources such as CPU cores, disk space, and network bandwidth. This resources must be sliced and shared between virtual machines running potentially heterogeneous workloads. We outline the taxonomy for resource management elements.

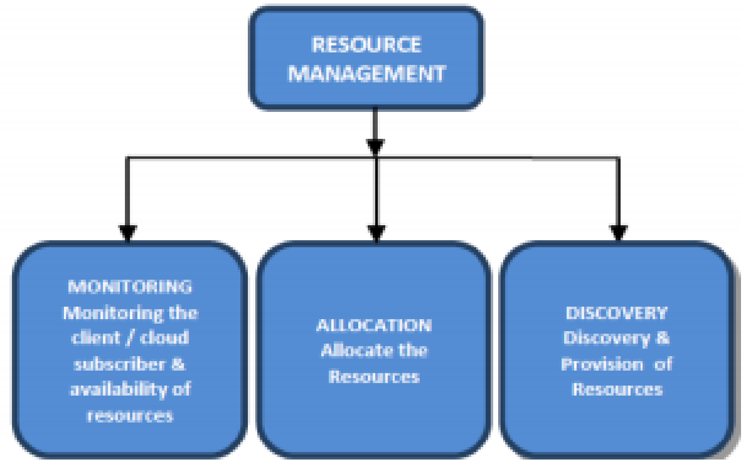

Fig. 1 Elements of Resource Management.

The fundamental element of resource management is the discovery process. It involves searching for the appropriate resource types available that match the application requirements. The process is managed by the cloud service provider. This process is being taken by the resource broker or user broker to discover available resources. Discovery consists of detailed description of resources available. According to, resource discovery provide a way for a resource management system (RMS) to determine the state of the resources that are managed by it and other RMSs that interoperate with it. The resource discovery works with dissemination of resources to provide information about the state of resources to the information server.

A. Significance of Dynamic Resource Allocation

In cloud computing, Dynamic Resource Allocation is the process of assigning available resources to the needed cloud applications over the internet. Resource allocation starves services if the allocation is not managed precisely. Resource provisioning solves that problem by allowing the service providers to manage the resources for each individual module. Dynamic Resource Allocation Strategy (DRAS) is all about integrating cloud provider activities for utilizing and allocating scarce resources within the limit of cloud environment so as to meet the needs of the cloud application. It requires the type and amount of resources needed by each application in order to complete a user job. The order and time of allocation of resources are also an input for an optimal DRAS. An optimal DRAS should avoid the following criteria as follows: a) Resource contention situation arises when two applications try to access the same resource at the same time. b) Scarcity of resources arises when there are limited resources. c) Resource fragmentation situation arises when the resources are isolated. There will be enough resources but not able to allocate to the needed application. d) Over-provisioning of resources arises when the application gets surplus resources than the demanded one. e) Under-provisioning of resources occurs when the application is assigned with fewer numbers of resources than the demand. The dynamic resource allocation based on distributed multiple criteria decisions in computing cloud.

VI. METHODOLOGY

Cloud computing has taken the degree of efficiency and agility realized from virtualization. Virtualization helps efficient use of hardware resources. Hence Virtual Machines are allocated to the user based on their job in order to reduce the number of physical servers in the cloud environment. But most VM resources are not efficiently allocated based on the characteristics of the job to meet out Service Level Agreements (SLA). Hence, we propose a dynamic VM allocation model based on the characteristics of the job, which can dynamically reconfigure virtual resources and thereby increasing the resource utilization.

The greedy module enumerates resource allocations for the VMs based on the estimated cost of the given workloads.

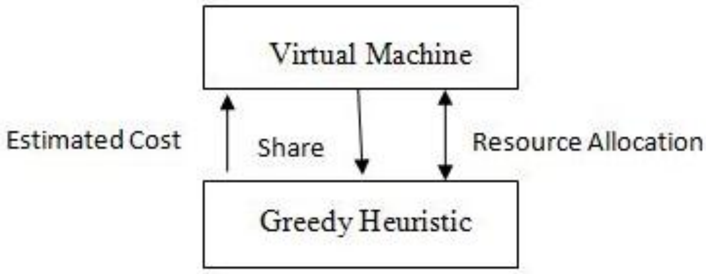

Fig. 2 Block Diagram for GRA

The GRA algorithm

Step 1: Initially, equal allocation of each resource is assumed as the initial configuration for all VMs ($1/N$ of each resource is allocated to each VM).

Step 2: The fitness function is defined to minimize the cost as described in Eq. (3), and then the positions (*share* values) of the particles are chosen randomly. The search space includes all the possible fractions except the fractions that cause a resource allocation that is either greater than the maximum allocation (100%) or less than the minimum allocation (0%). These constraints reduce error occurrence and can be described by the following:

Min(Ri)-share>0

Max(Ri)+share<100

Moreover, the search space boundaries $[X_{min}, X_{max}]^D$ are restricted in [0.001, 0.1]. This restriction means that each share parameter can be any value between 0.1% and 10%. In this work, only one resource, CPU, is used (i.e., one-dimensional vectors for particles), and thus, GPSO is used to find a best particle (share value) to tune CPU allocation

$X = (x_1, x_2, x_n)$. ….. (3)

Step 3: GRA operates then in iterations. Iteratively, each particle evaluates its position by running the greedy algorithm and determines its personal best position. The global best share and VM configuration are then determined. The initial VM configuration of the greedy algorithm for each particle is the VM configuration which was tuned by the global best particle of the previous iteration. Each particle then updates its own velocity using its previous velocity, the inertia weight, its previous position, its personal best position, and best particle in terms of fitness in the entire population (global best position). Each particle then uses the calculated velocity to adjust its new position.

Step 4: After the iterations terminate, the configuration of the best particle so far is output as the final VM configuration R.

As stated previously in the listed steps, for each iteration, the greedy algorithm uses the new share and the previous optimal configuration as the initial state.

Advantages:

1) The biggest benefit of resource allocation is that user neither has to install software nor hardware to access the applications, to develop the application and to host the application over the internet.
2) The next major benefit is that there is no limitation of place and medium. We can reach our applications and data anywhere in the world, on any system.
3) The user does not need to expend on hardware and software systems.
4) Cloud providers can share their resources over the internet during resource scarcity.

VII. CONCLUSION

Cloud computing technology is increasingly being used in enterprises and business markets. In cloud paradigm, an effective resource allocation strategy is required for achieving user satisfaction and maximizing the profit for cloud service providers. This paper summarizes the classification of RAS and its impacts in cloud system. Some of the strategies discussed above mainly focus on CPU, memory resources but are lacking in some factors. Hence this survey paper will hopefully motivate future researchers to come up with smarter and secured optimal resource allocation algorithms and framework to strengthen the cloud computing paradigm.

REFERENCES

[1] Armbrust M et al (2009) Above the clouds: a Berkeley view of cloud computing. UC Berkeley Technical Report.

[2] R. B. Bohn, J. Messina, F. Liu, J. Tong, and J. Mao, “NIST cloud computing reference architecture,” in Proc. 2011 IEEE World Congress on Services, 2011, pp. 594-596.

[3] M. Armbrust, A. D. Joseph, R. H. Katz, and D. A. Patterson. (February 2009). Above the clouds: A Berkeley view of cloud computing. Electrical Engineering and Computer Sciences. 53(UCB/EECS-2009-28). [Online]. pp. 7-13. Available: http://www.eecs.berkeley.edu/Pubs/TechRpts/2009/EECS-200928.html

[4] Y. Hu, J. Wong, G. Iszlai, and M. Litoiu, “Resource provisioning for cloud computing,” in Proc. the 2009 Conference of the Center for Advanced Studies on Collaborative Research, 2009, pp. 101–111.

[5] R. Buyya and R. Ranjan, “Special section: Federated resource management in grid and cloud computing systems,” Future Generation Computer Systems, vol. 26, no. 8, pp. 1189–1191, June, 2010.

[6] Mohd Hairy Mohamaddiah, Azizol Abdullah, Shamala Subramaniam, and Masnida Hussin “A Survey on Resource Allocation and Monitoring in Cloud Computing”, International Journal of Machine Learning and Computing, Vol. 4, No. 1, February 2014.

[7] Ms Jayshri Damodar Pagare, Dr. Nitin A Koli, “A technical review on comparison of Xen and KVM hypervisors: An analysis of virtualization technologies”, International Journal of Advanced Research in

Computer and Communication Engineering Vol. 3, Issue 12, December 2014, ISSN (Online) : 2278-1021, ISSN (Print) : 2319-5940, pp. 8828-8832.

[8] V RaviTeja Kanakala, V.Krishna Reddy, K.Thirupathi Rao, "Analysis on Virtualization Technologies in Cloud", International Journal of Advanced Research in Computer Engineering & Technology (IJARCET), Vol 3 Issue 7, July 2014, ISSN: 2278 – 1323, pp 2567-2574.

[9] G. Cephas Obasuyi, Arif Sari, "Security Challenges of Virtualization Hypervisors in Virtualized Hardware Environment", Int. J. Communications, Network and System Sciences, 2015, vol. 8, pp. 260-273

[10] Hamid Reza Qavami, Shahram Jamali "A Survey On Resource Provisioning In Cloud Computing", International Journal Of Research In Computer Applications And Robotics Issn 2320-7345

[11] J. Kaplan, W. F., N. Kindler (2008). Revolutionizing Data Center Energy Efficiency, McKinsey & Company.

[12] Peter Jacko "Resource Capacity Allocation to Stochastic Dynamic Competitors: Knapsack Problem for Perishable Items and Index-Knapsack Heuristic" in Annals of Operations Research.

[13] K. Krauter, R. Buyya, and M. Maheswaran, "A taxonomy and survey of grid resource management systems for distributed computing," Software: Practice and Experience, vol. 32, no. 2, pp. 135–164. 2002.

