International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

VIRTUAL MACHINE MANAGEMENT IN CLOUD COMPUTING ENVIRONMENT

Mr. Prathamesh V. Kawale[1], Miss. Poonam M. Tajne[2], Miss. Punam R. Thakare[3], Miss. Gayatri S. Chaudhari[4]

Abstract:

Cloud computing has emerged as a cornerstone technology for delivering scalable and flexible computing resources to meet the ever-growing demands of modern applications. Central to the efficiency and effectiveness of cloud computing infrastructures is the management of virtual machines (VMs), which serve as the fundamental units for resource allocation and deployment. This paper presents a comprehensive review and analysis of virtual machine management in cloud computing environments, focusing on the challenges, existing solutions, and emerging trends.

The research begins by examining the key components of VM management, including provisioning, scheduling, monitoring, and optimization. It investigates the complexities associated with these tasks in large-scale cloud environments, such as dynamic resource allocation, workload balancing, and cost optimization. Moreover, the paper explores the impact of factors like network latency, security, and energy efficiency on VM management strategies.

Furthermore, the study evaluates current approaches and techniques employed in VM management, ranging from traditional methods to more advanced algorithms and frameworks. It identifies the strengths and limitations of existing solutions and highlights areas for improvement and innovation. In particular, the paper explores the potential of machine learning, artificial intelligence, and automation in enhancing VM management processes.

Drawing on insights from the analysis, the paper proposes a novel framework for optimizing virtual machine management in cloud computing environments. This framework integrates adaptive resource allocation, predictive analytics, and policy-based decision-making to dynamically optimize VM placement, performance, and utilization. It emphasizes the importance of flexibility, scalability, and resilience in modern cloud infrastructures.

International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

1. INTRODUCTION:

1.1 Cloud Computing Overview

Cloud computing is a technology model that enables access to and delivery of computing services over the internet. Instead of owning and maintaining physical servers and infrastructure, users can leverage computing resources, such as servers, storage, databases, networking, software, and analytics, provided by third-party vendors. These resources are hosted in data centers and made available to users on a pay-as-you-go or subscription basis.

1.2 Key Characteristics:

a. On-Demand Self-Service: Users can provision computing resources as needed without requiring human intervention from the service provider.
b. Broad Network Access: Cloud services are accessible over the network and can be accessed through standard mechanisms, allowing users to connect from various devices.
c. Resource Pooling: Computing resources are pooled to serve multiple customers, with the provider dynamically assigning and reallocating resources based on demand.
d. Rapid Elasticity: Resources can be quickly scaled up or down to accommodate changing workloads, providing flexibility and efficiency.
e. Measured Service: Cloud systems automatically control and optimize resource usage through metering capabilities, enabling users to pay for only the resources they consume

The recent materialization of this new computing model has radically changed everyone's perception of infrastructure paradigm, development models and delivery of software. It frees them from setting up IT infrastructures and thus enables them to focus on innovation. It is revolutionizing the IT industry by enabling them to offer access to their infrastructure and application services on a "pay- as- you- use" basis. As a result, several enterprises including IBM, Microsoft, Google and Amazon have started to offer different Cloud services to their customers. It inherits many features from Grid Computing and Utility Computing. Cloud computing has come out as a model to deliver on demand resources to consumers similar to other utilities like electricity, gas, water. Earlier, small and medium enterprises had to make high capital investment for procuring IT infrastructure, manpower which results in a high cost of ownership. Cloud computing aims to deliver services that user can access from anywhere, irrespective of their location, on subscription basis. Therefore, these enterprises now need not to make large investment in hardware to deploy their services or human powerCloud systems are less expensive to operate, consume less energy, and have higher utilization rates than traditional datacenters, which lead to the belief that much of the work done in traditional datacenters today will be pushed to the cloud by the end of the decade.

2. Definition

The National Institute of Standards and Technology (NIST) has defined cloud computing as a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources, e.g. networks, servers, storage, applications, and services, that can be rapidly provisioned and released with minimal management effort or service provider interaction. In contrast to the conventional computing model, where end-user data and computing power are located in the user computer systems, Cloud computing resources are provided in massive, Virtualized infrastructures managed by professional

service providers.

3. Service Models

Cloud Computing Architecture offers following Service Models:

A. Infrastructure as a Service (IaaS)- Cloud clients directly use IT infrastructures which may include computing, storage, network elements, and other essential computing resources available in the IaaS cloud. Virtualization plays a vital role in IaaS cloud in order to assimilate or divide physical resources to meet rapidly changing resource demand from Cloud consumers. Virtualization creates independent Virtual Machines (VM) that are separated from both the underlying hardware and other VMs.

Characteristics of IaaS:

1. Flexibility: IaaS provides a high level of flexibility, allowing users to choose the operating system, applications, and configurations that suit their specific needs.
2. Scalability: Resources can be easily scaled up or down based on demand. This elasticity enables organizations to adapt to changing workloads without significant upfront investments.
3. Cost Efficiency: IaaS follows a pay-as-you-go model, where users are billed based on their actual usage of resources. This cost-effective approach eliminates the need for large upfront capital expenditures.
4. Self-Service: Users can provision and manage their virtual machines, storage, and networking resources through a web-based interface or API, reducing the need for manual intervention from the service provider.

B. Platform as a Service (PaaS)- PaaS offers complete software development platform facilitating cloud clients to develop cloud services and applications directly on the PaaS cloud. Infect Cloud PaaS is the use of tools and resources running on a cloud infrastructure to provide services to end-users. Clients can develop applications on top of the tools provided.

Characteristics of PaaS:

1. Abstraction of Infrastructure: PaaS abstracts the underlying infrastructure, allowing developers to focus solely on application development without dealing with hardware or networking concerns.
2. Productivity and Efficiency: PaaS enhances developer productivity by providing a streamlined development environment. Developers can leverage pre-built components and focus on coding instead of infrastructure management.
3. Automatic Scaling: PaaS platforms often support automatic scaling, allowing applications to scale up or down based on demand. This ensures optimal performance without manual intervention.
4. Rapid Development and Deployment: PaaS accelerates the development and deployment cycles. Developers can quickly build, test, and deploy applications, reducing time-to-market for new features.

C. Software as a Service (SaaS)- Cloud SaaS is the use of applications running on a cloud

infrastructure to provide services to end-users. SaaS can deliver business applications such as Customer Relationship Management (CRM), Enterprise Resource Planning (ERP), and Accounting etc. Examples of cloud SaaS are Google Apps [3,32] and Salesforce CRM.

Characteristics of SaaS:

1. Cost Efficiency: SaaS eliminates the need for organizations to invest in and maintain software infrastructure. Users typically pay a subscription fee based on usage, reducing upfront costs and total cost of ownership.
2. Automatic Updates: SaaS providers handle software updates and maintenance, ensuring that users always have access to the latest features and security enhancements.
3. Subscription Model: SaaS follows a subscription-based pricing model, where users pay a recurring fee for access to the software. This model often includes service and support from the provider.
4. Reduced IT Overhead: Organizations using SaaS can reduce the burden on their internal IT teams, as the provider is responsible for managing and maintaining the software infrastructure.

4. Deployment Models

Cloud computing offers various deployment models that organizations can choose based on their specific needs, preferences, and security requirements. These deployment models define how cloud services are provisioned, managed, and accessed. In this comprehensive overview, we'll explore the four main deployment models: Public Cloud, Private Cloud, Hybrid Cloud, and Community Cloud.

1. Public Cloud: The public cloud deployment model is perhaps the most well-known and widely used. In a public cloud, cloud services are provided by third-party vendors and made available to the general public over the internet. These services are hosted in data centers, and customers can access resources such as virtual machines, storage, and applications on a pay-as-you-go basis.

Key Characteristics:

1. Accessibility: Public clouds offer broad network access, enabling users to access services from anywhere with an internet connection.
2. Cost Efficiency: Since resources are shared among multiple users, the cost is distributed, making it cost-effective for individual users or small to medium-sized businesses.
3. Scalability: Public clouds provide on-demand scalability, allowing users to quickly scale up or down based on their requirements.

Use Cases:

Public clouds are suitable for a wide range of use cases, including web hosting, development and testing environments, collaboration tools, and applications with varying or unpredictable workloads.

2. Private Cloud: In a private cloud deployment model, cloud resources are used exclusively by a single organization. The infrastructure may be owned, managed, and operated by the organization itself (on-premises) or by a third-party provider (off-premises). Private clouds offer enhanced control, security, and customization compared to

public clouds.

Key Characteristics:

1. Control: Organizations have direct control over their private cloud infrastructure, allowing for tailored configurations based on specific security and compliance requirements.
2. Security: Private clouds provide a higher level of security since resources are not shared with other organizations. This is particularly important for industries with strict compliance regulations.
3. Customization: Users can customize the private cloud environment to meet the unique needs of their applications and workloads.

Use Cases:

Private clouds are often preferred by enterprises and industries with stringent security and compliance requirements, such as finance, healthcare, and government. They are suitable for hosting sensitive data, critical applications, and workloads that demand high levels of customization.

3. Hybrid Cloud: The hybrid cloud deployment model combines elements of both public and private clouds. In a hybrid cloud, data and applications can move seamlessly between the two environments, allowing for greater flexibility and optimization of resources. Organizations can choose where to run specific workloads based on factors like cost, performance, and security.

Key Characteristics:

1. Flexibility: Hybrid clouds provide the flexibility to run applications and store data in the most suitable environment, whether in the public or private cloud.
2. Scalability: Organizations can leverage the scalability of the public cloud for certain workloads while maintaining control over sensitive data and critical applications in the private cloud.
3. Cost Optimization: Hybrid clouds allow organizations to balance costs by using cost-effective public cloud resources for non-sensitive workloads while investing in a private cloud for mission-critical applications.

Use Cases:

Hybrid clouds are ideal for organizations with dynamic workloads, seasonal demands, or those undergoing digital transformation. They provide a strategic approach to managing resources efficiently while maintaining control over critical assets.

4. Virtual Private Cloud (VPC): Amazon Web Services has recently offered a new deployment architecture, a secure and seamless bridge between an organization's existingIT infrastructure and the Amazon public cloud. This is a blend of Private Cloud and Public Cloud. VPC offers perfect balance between plus side of Private Cloud i.e. Control and plus side of Public Cloud i.e. flexibility.

Key Characteristics:

1. Collaboration: Community clouds facilitate collaboration and resource sharing among organizations with similar requirements, ensuring that shared concerns are adequately addressed.
2. Customization: Users in a community cloud can customize their environment to meet specific industry standards and compliance regulations.

3. Cost Sharing: Similar to public clouds, community clouds allow for cost sharing among organizations, making it more affordable for community members.

Use Cases:

Community clouds are commonly used in industries such as healthcare, where organizations need to collaborate on data sharing while adhering to strict privacy and security regulations. They are also suitable for research communities and educational institutions with shared interests and requirements.

5. VIRTUALIZATION:

5.1 Brief Overview

Visualization is a powerful means of communication that transforms complex data and information into a visual representation, enabling individuals to gain insights, make informed decisions, and comprehend patterns and trends. Through the use of graphical elements such as charts, graphs, maps, and diagrams, visualization transcends the limitations of raw data, making it more accessible, understandable, and actionable.

Types of Visualizations:

1. Charts and Graphs: Bar charts, line graphs, pie charts, and scatter plots are common types of visualizations for quantitative data. They help depict relationships, trends, and distributions.
2. Maps: Geographic information is often presented using maps. They can show spatial patterns, distribution of data across regions, and help in location-based decision-making.
3. Infographics: Infographics combine various visual elements such as charts, images, and text to convey information concisely. They are commonly used in marketing, education, and journalism.
4. Dashboards: Dashboards provide a consolidated view of multiple visualizations, allowing users to monitor key metrics and data points in real-time. They are frequently used in business intelligence and analytics.
5. Network Diagrams: Network diagrams represent relationships between entities, showing connections and dependencies. They are used in IT for illustrating network architectures and relationships.
6. Tree Diagrams: Tree diagrams illustrate hierarchical structures, relationships, and classifications. They are valuable for representing organizational hierarchies, taxonomies, and family trees.

Virtualization technologies enable the execution of multiple operating system instances, or Virtual Machines on the same physical piece of hardware. Each VM works just like Physical Machines functions as if it is its own Physical Machine with a dedicated Operating System and hosted applications. Each VM requires access control, sometimes different between different VMs on the same Physical hardware platform. Some Virtualization platforms require an external host operating system; others are embedded directly in the hardware. There are several common approaches to Virtualization. The significant difference between the various approaches lies in the component that has visibility and control over the Virtual Machines. Recently emerged Cloud computing paradigm leverages Virtualization and provides on- demand resource provisioning over the Internet on a pay-as-you go basis. This facilitates enterprises to reduce the expenditure on maintenance of their own computing environment and outsource the computational needs to the Cloud. Therefore, Virtualization forms the basis of Cloud Computing, as it provides the capability of

pooling computing resources from clusters of servers and dynamically provisioning of Virtual resources to applications on-demand. While convenient, the use of VMs gives rise to further challenges such as the intelligent allocation of physical resources for managing competing resource demands of the users.

5.2 Server Consolidation

Virtualization in cloud centers refers to the process of demultiplexing a Physical server into two or more Virtual Machines and allocating one Virtual Machine to each application, giving illusion of full control of Physical Machine. Virtualization provides a new way to improve the power efficiency of the datacenters through consolidation. Consolidation means assigning more than one VM to a Physical server. As the traffic volume varies throughout the day, Virtual Machines can be consolidated and mapped to subset of Physical Machine thereby allowing shut down of few Physical Machines when load is less. A key benefit of Virtualization technology is the ability to contain and consolidate the number of servers in a datacenter. This allows businesses to run multiple applications and OS workloads on the same server. In current trend, 10 server workloads running on a single Physical server is typical, but some companies are consolidating around 35 workloads onto one server. But this is not a rule of thumb, number depends upon workload condition. As a result, server utilization increases and the datacenter energy and cooling costs are lowered.

Key Components of Server Consolidation:

1. Hypervisor: The hypervisor, also known as a Virtual Machine Monitor (VMM), is a critical component in server consolidation. It allows multiple VMs to run on a single physical server by managing and allocating resources among them.
2. Virtual Machines: Virtual machines are self-contained software instances that emulate the functionality of physical servers. They share the resources of a host server but operate independently.
3. Resource Pooling: Server consolidation involves pooling and optimizing resources such as CPU, memory, storage, and network bandwidth. The hypervisor allocates these resources dynamically based on the needs of each virtual machine.
4. Dynamic Resource Allocation: The hypervisor dynamically allocates resources to VMs based on demand. This ensures that each VM receives the necessary resources to perform efficiently while preventing resource wastage.
5. Workload Balancing: Workload balancing is a process where the hypervisor distributes VMs across physical servers to ensure an even load. This helps prevent resource bottlenecks and enhances overall system performance.

As nothing comes free of cost, Consolidation is not without performance penalty. All the techniques involve Performance-Power tradeoff. More precisely, if workloads are consolidated on fewer servers, performance of the consolidated VMs may deteriorate because of the nonavailability of sufficient resources to applications in Virtual Machines running in Physical Machine, but the power efficiency will improve because fewer servers will be used to service the VMs . Virtualization makes sense in cases where we have several underutilized Virtual servers and can gain higher efficiency by combining them and raising server utilization. In a situation, such as with Google or Facebook (highly loaded servers), where servers already run at maximum utilization, Virtualization makes less

sense and, in fact, can add overhead due to the CPU resource used up by a hypervisor.

Implementation Steps for Server Consolidation:

1. Assessment: Conduct a thorough assessment of the existing server infrastructure, identifying underutilized servers and analyzing resource usage patterns.
2. Capacity Planning: Determine the capacity and resource requirements of each workload. This involves understanding peak usage, resource demands, and growth projections.
3. Virtualization Platform Selection: Choose an appropriate virtualization platform or hypervisor that aligns with organizational requirements. Popular choices include VMware, Microsoft Hyper-V, and KVM.
4. Virtual Machine Configuration: Create virtual machines based on the capacity planning results. Define the number of CPUs, amount of memory, and storage requirements for each VM.
5. Migration Strategy: Plan the migration strategy for moving workloads from physical to virtual servers. Migration methods include physical-to-virtual (P2V) conversions, live migrations, or reinstallation.
6. Performance Monitoring: Implement performance monitoring tools to track the performance of VMs and the overall infrastructure. This helps in identifying any performance bottlenecks and making necessary adjustments.
7. Workload Balancing: Use workload balancing features provided by the virtualization platform to distribute VMs across physical servers evenly. This ensures optimal resource utilization.
8. Security Measures: Implement security measures to address the potential risks associated with consolidating multiple workloads on a shared infrastructure. This may include network segmentation, access controls, and encryption.
9. Documentation: Maintain comprehensive documentation of the virtualized environment, including VM configurations, network settings, and security policies.

6. PROMISING FIELD FOR RESEARCH:

According to James Hamilton, VP Cloud Computing Services, Amazon, the actual cost of power consumed by the servers plus the cost of cooling the servers in a year is 34% of the total cost of ownership of a datacenter, whereas the amortized server costs in a 10-year lifetime of a datacenter is 54% of the total cost. This implies if we concentrate on getting maximum throughput from servers by consolidation, we will be able to reduce datacenter power consumption and cooling costs and that will be the biggest savings for a datacenter operator. From 1993 to 2007 the performance of the top 500 systems had increased from 59.7 Gflops to 280.6 Tflops, roughly a 4700-fold speedup. But the power required to run the Machines was hardly taken into account. For example, the 2004's top ranking Machine, the Japanese Earth Simulator, required 12 MW of power to operate, roughly the amount required to power a small town. That amount may be translated to $10 million of yearly operating costs. The US Environmental Protection Agency calculated that the US spent $4.5 billion on electrical power to operate and cool ICT and HPC servers in 2006, forecasting expenditures of more than $7.4 billion in 2013. This trend results in a significant increase in operating costs of modern

servers. For many organizations, the cooling and power costs of running a server for two years are equivalent to the price of its purchase. Due to the environmental and more important economical gains, researchers as well as leading Entrepreneurs, pioneers and technological companies have begun work targeting different aspects of the power consumption problem. The approaches can be roughly grouped into two categories: static and dynamic approaches. Static approaches constitutes of solutions to enhance the power efficiency of equipments of data centers and single servers. The dynamic approaches deals with analysis of workload of different applications and adjusting maximum Virtual Machines to minimum number of servers keeping constraints into consideration. This enables shut down of remaining idle servers. Low utilization is due two reasons. Firstly, cloud service providerand clients sign SLA between them. If SLA is violated then service provider has to pay penalty. So resources are over provisioned in order to meet peak time resource demand which results in underutilization of resources in off peak hours. Secondly, cohosted application may require separate OS images which may results in separate OS image for each application or separate Physical Machine. Similarly when servers are overloaded with Virtual Machines, then decision; which Virtual Machine to migrate, where to migrate is of important concern.

7. CHALLENGES:

However, valuable consolidation is not as easy as packing the maximum workload in the minimum number of servers, keeping each resource on every server at 100% utilization.

Performing consolidation, keeping energy efficiency in consideration while giving quality of services, poses several concerns.

i) Firstly, consolidation techniques must carefully decide which workloads are compatible and should be multiplexed on a common Physical server. Workload resource usage, performance, and energy usages are not simply additive. Understanding the flavor of their composition is thus significant to decide which workloads can be packed together.

ii) Secondly, there exists an optimal performance and energy point . This happens because consolidation leads to performance degradation that causes the executiontime to increase, eating into the energy savings from reduced idle energy. Further, the optimal point changes with acceptable degradation in performance and application mix. Determining the optimal point and tracking it as workloads change, thus becomes important for energy efficient consolidation .

8. RELATED WORK:

Research has been going on related to power saving resource management approacheswhich limit the power consumption of servers and consequently that of datacenters to predefined thresholds. In a model is proposed for Virtual Machine allocation. But this model is restrictive in use as several conditions are stated like nature of workload, which may not be workable in real life. These models may not be helpful in multiple dimension of resource consumption. In an algorithm is proposed for VM provisioning keeping SLA in considerations. This algorithm uses first fit method. Their approach is based on focusing on pastscenario of demand, predicting the future scenario of demand and then mapping VMs to PMs. Resource demands are anticipated at periodic intervals. These anticipated values are used by a placement module to compute

VM to PM mappings. In an algorithm is proposed which find out the destination PM for remapping of VM of the highly loaded nodes using dot products of capacity usage and resource requirement vectors. In an algorithm is presented which reduces the expenditure in the reservation and on-demand subscription plans generally offered by cloud service providers. Here expenditure is the price charged by provider for each resource. It considers resource allocation in three stages: reservation, utilization and on-demand. Applying stochastic integer programming approach, algorithm calculates the number of Virtual Machines demanded in reservation phase and then calculates the number of Virtual Machines allocated in both utilization and on-demand phase. But as demand and cost may rise in current real-world scenario, scalability will be the problem. In instead of CPU, network interfaces, switches, routers, links are considered. They have suggested methodology to put these devices into sleep mode when they are idle. In two approaches are proposed which estimates energy consumption of eachVirtual Machine dynamically by measuring its resource usage. First approach considers energy consumption as product of processor time and average power consumption of processor, which shows poor efficiency. Second approach which is refinement of first incorporates an integrated power measurement device which takes power consumption of the whole system. When a new Virtual Machine is introduced, it takes pattern of power consumption for 200s. This gives energy consumption behavior of new Virtual Machine after linear regression. Negative side of this approach are: first it requires integrated power measurement device and second, if workload scenario changes significantly, model shows poor output because it has assumed that behavior of Virtual Machine remains same as in first 200s. In [22] authors come up with an idea of Gang scheduling. Their approach is selecting those Virtual Machines for migration which share memory contents. They have tracked the identical contents of co hosted Virtual Machines and transfer those contents only oncewhile transferring all those Virtual Machines simultaneously to another Physical Machine.PM. This gives the advantages of achieving optimization of network and memory overhead of migration. This approach is taking care of memory and network elements efficiency only while putting more load on CPU for running this complex algorithm.

9. SUITABLE APPROACHES:

As the theme of our work is energy efficiency in Virtual Machine allocations, following mathematical models are favorable to model the VMs to PMs mapping.

9.1 Constraint Programming

9.1.1. Brief Overview

Constraint programming (CP) is a powerful paradigm for representing and solving combinatorial problems. At its core, CP involves declaring a set of variables, each with a domain of possible values, and a set of constraints restricting the acceptable combinations of values for these variables. The fundamental goal of constraint programming is to find solutions that satisfy all imposed constraints. The process of solving these constraints is typically performed by constraint solvers, which utilize a variety of algorithms for search and inference to efficiently explore the solution space and identify valid assignments to the variables.

Foundations of Constraint Programming: At the heart of constraint programming lies the concept of "variables" and "constraints." Variables represent the unknowns in a given problem, while constraints enforce relationships among these

variables. By defining variables with domains of possible values and specifying constraints on their combinations, constraint programming facilitates the modeling and solution of complex problems. Consider a simple scheduling problem where a set of tasks needs to be assigned to a set of resources, subject to various constraints such as resource availability and task dependencies. In this scenario, the resources and tasks can be represented as variables with defined domains, and the constraints will denote the relationships between these variables. For instance, a constraint might specify that a task cannot be assigned to a resource if it is already assigned to another task during the same time interval.

9.1.2. Domains

Constraint programming finds application across a wide array of domains due to its ability to model and solve complex combinatorial problems efficiently and flexibly. Some key domains where constraint programming is extensively used include:

1) Logistics and Scheduling: Constraint programming is widely applied in logistics and scheduling problems, such as vehicle routing, crew scheduling, timetabling, and resource allocation. It allows for the efficient optimization of resource utilization and scheduling based on various constraints and objectives.
2) Manufacturing and Production Planning: In manufacturing environments, CP is employed for production scheduling, equipment allocation, and inventory management. It helps in optimizing production processes, minimizing downtime, and managing resources effectively.
3) Telecommunications and Networking: Constraint programming is utilized for network optimization, routing, and bandwidth allocation in telecommunications and networking systems. It aids in designing efficient communication networks while adhering to constraints such as capacity limitations and connectivity requirements.
4) Bioinformatics and Computational Biology: CP techniques are increasingly employed in bioinformatics for DNA sequence analysis, protein structure prediction, and genome assembly. CP facilitates the systematic exploration of possibilities while considering biological and biochemical constraints.
5) Finance and Investment: Constraint programming is utilized for portfolio optimization, risk management, and financial planning. It aids in determining investment strategies that satisfy various risk, return, and diversification requirements.
6) Configuration and Design: CP is used in product configuration and design optimization, notably in industries such as automotive, aerospace, and custom manufacturing. It enables the automated selection of components and features while respecting design constraints and requirements.
7) Supply Chain Management: Constraint programming plays a crucial role in supply chain optimization, inventory management, and demand forecasting. It assists in streamlining logistics, minimizing costs, and ensuring efficient distribution of goods.
8) Puzzle Solving and Games: Constraint programming finds application in solving puzzles, logic games, and combinatorial challenges. It enables the systematic exploration of solution spaces and aids in generating valid solutions to complex

puzzles and games.

9.1.3. Applicability to VM placement

In the context of virtual machine (VM) placement, constraint programming (CP) plays a pivotal role in optimizing the allocation of virtual machines to physical hosts while adhering to various constraints and objectives. The applicability of constraint programming to virtual machine placement is particularly significant in cloud computing environments, data centers, and other distributed computing infrastructures where efficient resource management is essential.

Modeling Virtual Machine Placement with Constraint Programming:

1. Resource Allocation and Utilization: CP allows for the formalization of constraints related to resource allocation, such as CPU, memory, storage, and network bandwidth. By representing physical hosts and virtual machines as variables with associated domains of resource capacities, and by specifying constraints that capture resource requirements and limitations, CP facilitates the systematic exploration of valid allocations that optimize resource utilization.
2. Load Balancing: Constraints programming can model load balancing requirements, ensuring that the distribution of virtual machines among physical hosts is balanced to avoid overloading specific hosts while underutilizing others. Constraints related to CPU load, memory usage, or network traffic can be integrated into the placement model to achieve load distribution objectives.
3. Affinity and Anti-Affinity Rules: Constraints pertaining to affinity and anti-affinity between virtual machines, such as requirements for co-locating or isolating certain VMs, can be expressed in the CP model. This allows for the enforcement of rules that govern the placement of VMs based on their interdependencies, communication patterns, or specific deployment requirements.

9.2 Stochastic Integer Programming

9.2.1 Brief Overview

Stochastic Integer Programming (SIP) is a powerful optimization framework that addresses decision-making under uncertainty by combining elements of integer programming with stochastic modeling. SIP is particularly valuable in scenarios where the decision variables are subject to random variations, uncertainties, or probabilistic outcomes. This approach aims to optimize decisions by accounting for the randomness or variability in the problem parameters, yielding robust and resilient solutions to complex decision-making problems.

Foundations of Stochastic Integer Programming:

1. Representation of Uncertainty: In SIP, uncertainty is typically represented using probability distributions, scenarios, or stochastic processes. This can include uncertain demand patterns, fluctuating resource availability, variable market conditions, or other stochastic influences that impact decision variables in the optimization problem.
2. Integer Decision Variables: The inclusion of integer decision variables in SIP introduces an element of discreteness to the decision space, capturing scenarios where decisions must be

made in whole units (e.g., selecting the number of facilities to open, the quantity of items to produce, or the allocation of resources).

3. Integration of Stochastic Elements with Integer Constraints: SIP models integrate stochastic elements, often represented by random parameters or variables, into integer programming formulations. This integration involves defining constraints and objective functions that account for the variability of the stochastic parameters and their impact on the decision variables.

9.2.2 Applications

1. Supply Chain and Logistics: SIP is applied in supply chain optimization to address uncertainties in demand, lead times, and inventory levels, leading to robust and effective distribution and inventory management strategies.
2. Production and Capacity Planning: In manufacturing settings, SIP aids in decision-making related to production levels, resource allocation, and capacity expansion while considering probabilistic factors that affect production processes.

9.2.3 Applicability to VM placement

The applicability of stochastic integer programming (SIP) to the domain of virtual machine (VM) placement is particularly relevant in cloud computing, data centers, and distributed computing environments where the optimal allocation and resource management of virtual machines are essential. By incorporating elements of stochastic modeling, SIP provides a powerful framework for addressing uncertainties and variabilities in VM placement decisions, enabling the development of robust placement strategies that account for probabilistic influences and fluctuating resource demands.

9.3 Bin Packing

9.3.1 Brief Overview

Bin packing is a classic optimization problem concerned with efficiently allocating items of different sizes into a fixed number of containers, referred to as bins, with the objective of minimizing the number of bins used. The problem has numerous real-world applications, including resource allocation in computing systems, inventory management, and logistics planning. In Bin packing problem objects of different volumes, weights, shapes are placed in containers (bins) in such a way so that minimum number of bins is used. Many heuristics have been developed: for example, the first fit algorithm provides a fast solution but may not be optimal, involving placing each item into the first bin in which it will fit. The algorithm can be made much more effective by first sorting the list of elements into decreasing order (known as the first-fit decreasing algorithm), but in this also running time may increase for large lists.

9.3.2 Applications

1. Data visualization optimization: Bin packing algorithms can be used to optimize the layout of visual elements in data visualizations, such as charts, graphs, and dashboards. By efficiently packing the visual elements into available screen space, bin packing algorithms can help improve the overall readability and usability of the visualization.
2. Resource allocation visualization: In resource allocation scenarios, such as scheduling tasks or allocating computational resources, bin packing algorithms can be used to visualize how resources are being assigned to different tasks or processes. This can help in identifying bottlenecks, optimizing resource usage, and improving system performance.
3. Visualizing packing and shipping processes:

In logistics and supply chain management, bin packing algorithms can be used to visualize how items are packed into containers for shipping. This can help in optimizing the packing process, reducing waste, and minimizing shipping costs.

9.3.3 Applicability to VM placement

The VM placement problem can be designed as a bin packing problem by considering Physical Machines as bins and the Virtual Machines to be placed as objects to be filled in the bin. Then we can go as follows. Study the Previous demands for resources. Predict the future demands based upon past. Then Map Virtual Machines to Physical Machines.

9.4 Genetic Algorithm

9.4.1 Brief Overview

Genetic Algorithms (GAs) are adaptive heuristic search algorithm having roots in evolutionary ideas of natural selection and genetic. Genetic algorithms are a type of evolutionary algorithm inspired by the process of natural selection and genetics. They are used to solve optimization and search problems by mimicking the process of natural selection to evolve solutions to a given problem Genetic algorithms can be employed to optimize the allocation of resources in cloud computing environments. This includes tasks such as virtual machine placement, load balancing, and scheduling of workloads across the available resources. By evolving solutions over multiple generations, genetic algorithms can help in finding near-optimal resource allocation and scheduling strategies that minimize costs and maximize performance. Genetic algorithms can be utilized to optimize the energy efficiency of cloud data centers. In a genetic algorithm, a population of candidate solutions called individuals to an optimization problem is evolved toward better solutions. Each candidate solution has a set of properties (its chromosomes or genotype) which can be mutated and altered; traditionally, solutions are represented in binary as strings of 0s and 1s, but other encodings are also possible. The evolution usually starts from a population of randomly generated individuals and happens in generations. In each generation, the fitness of every individual in the population is evaluated, the more fit individuals are stochastically selected from the current population, and each individual's genome is modified (recombined and possibly randomly mutated) to form a new population . The new population is then used in the next iteration of the algorithm.

A typical genetic algorithm requires:

9.2.2.1 A genetic representation of the solution domain,

9.2.2.2 A fitness function to evaluate the solution domain.

Once the genetic representation and the fitness function are defined, a GA proceeds to initialize a population of solutions and then to improve it through repetitive application of the mutation, crossover, inversion and selection operators. Generally, the algorithm terminates when either a maximum number of generations has been produced, or a acceptable fitness level has been achieved for the population.

9.4.2 Applications

Genetic algorithms have a wide range of applications across various fields due to their ability to solve complex optimization and search problems. Some common application areas of genetic algorithms include:

1. Engineering and Design: Genetic algorithms are used for optimizing the design of complex systems, such as aircraft, automobiles, and industrial equipment. They can help in finding optimal configurations

and parameters that meet performance requirements while minimizing costs.

2. Financial Modeling and Investment Strategies: Genetic algorithms are applied in financial modeling to optimize investment portfolios, develop trading strategies, and forecast market trends. They can be used to evolve trading rules and investment allocations that maximize returns and minimize risks.
3. Robotics and Control Systems: Genetic algorithms are used in the design and optimization of robotic systems, including path planning, motion control, and robot configuration. They can help in evolving control strategies that enable robots to perform tasks efficiently and adapt to changing environments.

9.4.3 Applicability to VM placement

The VM placement problem can be visualized as a genetic programming problem as follows. The solution domain can be represented as the Physical Machines in which Virtual Machines are to be placed. The fitness function can be defined over the number of Physical Machines in the problem. The aim would be to deliver a solution that is nearly optimal in terms of the number of Physical Machines used and the efficiency of packing of the Physical Machines. All of these approaches lead to efficient mapping of Virtual Machines on available Physical Machines satisfying some conditions like SLA Compliance. The expenditure can be reduced by optimally using the resources available and by shutting down the servers which are now free of any load. The algorithm to be used varies as per the workload conditions, resources available, SLA specifications.

10. PROPOSED WORK:

A lot of energy can be saved by prudent decision in VM management which will result in reduced energy consumption, more profit to cloud service providers, consequently lower price to customers and most importantly leading to Green Cloud computing environment which is envisioned by researchers.

The objective of this thesis proposal is to

- Design an effective framework for energy aware Virtualization in Cloud Computing Environment.
- Analyze different aspects involved in taking decision about which Virtual Machine to migrate, when to migrate and where to migrate.

11. REFERENCES:

[1] R. Buyya, C. Yeo, S. Venugopal, J. Broberg and I. Brandic, “Cloud computing and emerging IT platforms: vision, hype, and reality for delivering computing as the 5th utility,” *Future Generation Computer Systems* 25 (6) pp. 599– 616, 2009.

[2] Tharam Dillon, Chen Wu and Elizabeth Chang, “Cloud Computing: Issues and Challenges,” *24th IEEE International Conference on Advanced Information Networking and Applications, IEEE Computer Society*, 2010

[3] Marios D. Dikaiakos, George Pallis, Dimitrios Katsaros, Pankaj Mehra, and Athena Vakali, “Cloud Computing :Distributed Internet Computing for IT and Scientific,” *1089-7801/09 IEEE Computer Society*, 2009.

[4] C. Vecchiola, R.N. Calheiros, D. Karunamoorthy, and R. Buyya , “ Deadline-driven provisioning of resources

for scientific applications in hybrid clouds with Aneka," *Future Generation Computer Systems* 28, pp. 58–65, 2012.

[5] Xiaofei Liao, Hai Jin and Haikun Liu, "Towards a green cluster through dynamic remapping of Virtual Machines," *Future Generation Computer Systems* 28 pp. 469-477, 2012

[6] Zhengkai Wu, Christopher Giles and Jun Wang, " Classified power capping by network distribution trees for green computing," *Springer Science + Business Media*,2011

[7] C. Lefurgy, X. Wang and M. Ware, "Server-level power control," *Proc. Fourth International Conference on Autonomic Computing*, 2007.

[8] S. Govindan, J. Choi, B. Urgaonkar, A. Sivasubramaniam, and A. Baldini, "Statistical profiling-based techniques for effective power provisioning in data centers," Proc. EuroSys'09, 2009.

[9] Dimitris Economou, Suzanne Rivoire, Christos Kozyrakis and Partha Ranganathan, "Full-system power analysis and modeling for server environments," *Proc. 2nd WS Modeling, Benchmarking and Simul., Boston, MA*, pp. 158–16, 2006.

[10] Norman Bobroff, Andrzej Kochut, and Kirk A. Beaty, " Dynamic placement of Virtual Machines for managing SLA violations,*" Integrated Network Management,IEEE Computer Magazine,"* pp. 119-128, 2007.

[11] Aameek Singh, Madhukar Korupolu, and Dushmanta Mohapatra, " Server-storage Virtualization: integration and load balancing in data centers*," Proc. 2008 ACM/IEEE conference on Supercomputing, pages 1–12, Piscataway, NJUSA, IEEE Press*, 2008

[12] Sivadon Chaisiri, Bu-Sung Lee, and Dusit Niyato, "Optimal Virtual Machine placement across multiple cloud providers," *editors, APSCC*, pp, 103–110, IEEE, 2009.

[13] C. Panarello, A. Lombardo, G. Schembra, L. Chiaraviglio and M. Mellia, "Energy saving and network performance: a trade-off approach," *Proc.1st ACMInternational Conference on Energy-Efficient Computing and Networking, e- Energy 2010*," Passau, Germany, pp. 41–50, 2010.

[14] A. Kansal, F. Zhao, J. Liu, N. Kothari and A.A. Bhattacharya, " Virtual Machine power metering and provisioning," *Proc. 1st ACM symposium on Cloud Computing*, 2010.

[15] U. Deshpande, X. Wang, and K. Gopalan, "Live Gang Migration of Virtual Machines," *High-Performance Parallel and Distributed Computing*, June 2011.

[16] Anton Beloglazova, Jemal Abawajyb and Rajkumar Buyyaa, "Energy-Aware Resource Allocation Heuristics for Efficient Management of Data Centers for Cloud Computing," *Future Generation Computer Systems*, pp. 755-768, 2012