Context Aware VM Placement Optimization with Minimal Cost Based Resource Provisioning and Scheduling

S. Kayalvili[1], P. Poovizhi[2] and S. Saranyaprabha[3]

[1]Department of Computer Science and Engineering, VCET, Tamilnadu, India. Email: kayalvilis@gmail.com.

[2]Department of Computer Science and Engineering, VCET, Tamilnadu, India. Email: poovizhi0804@gmail.com.

[3]Department of Computer Science and Engineering, VCET, Tamilnadu, India. Email: s.saranyabe2016@gmail.com.

Abstract - In the cloud environment, the workflows have been frequently used to model large- scale problems in areas such as bioinformatics, astronomy, physics and arithmetic process. Such a resource obtains a task from the cloud providers that have ever-growing data and computing requirements and therefore demands a high-performance computing environment in order to be executed in a reasonable amount of time. This work proposes a context-aware heuristic-based solution for VM placement optimization in cloud data centers. The proposed technique takes into account physical machine characteristics and load (peak and non-peak) conditions in the data centers to save power and also improve performance efficiency for data center owners. This work proposes a resource provisioning and scheduling strategy for scientific workflows on Infrastructure as a Service (IaaS) and Platform as services clouds (PaaS). This work minimizes the overall workflow execution cost using the Superior Element Multitude Optimization (SEMO) algorithm. The main scope of the work is used to analyse the best available resources in the cloud environment depend upon the total execution time and cost which is compared between one process to another. If the provider satisfies the least time, then the process terminates.

Keywords - Virtual machine placement optimization, resource provisioning and scheduling.

I. INTRODUCTION

Cloud computing is being adopted rapidly by all businesses because of its profitable benefits. It helps businesses avoid higher investment for IT infrastructure at the beginning and reduces the hardware maintenance costs, administration costs, and worries from their users. Cloud computing services are offered by multiple vendors in distinct models such as Infrastructure-as-a-Service, Platform-as-a-Service, and Software-as-a-Service to its users based on their business needs. Cloud data centers are a farm of heterogeneous computing servers that are provisioned dynamically to user applications. The virtualization technology enables data centers to maximize utilization by sharing physical computing nodes between users/ applications. If cloud computing has to succeed, it has to be cost-effective and efficient for bothcloud service providers and its users.

A Variety of services is being offered in the cloud environment at an ever-increasing pace, and the users across the globe consume its services. The cloud service providers are building their data centers at geo-distributed locations to cater to users located at different geo-regions to improve performance, fault tolerance, and also to provide reliable services round the clock. The cloud service providers make a significant investment at the beginning to set up data centers for IT infrastructure and other logistics, and later they incur significant data center management costs to keep their data centers running. The data center management costs include power/electricity costs, hardware & software maintenance costs, and other logistics costs. The data center management costs vary greatly based on the total power usage, electricity cost at that location, and space renting costs. As per a published study, the electricity/ power costs contribute to around 13% of the overall cost for data center management, which is a significant share contributing to the cost of data center owners. It is vital to reduce the power consumption of the data center whenever and wherever there is scope without affecting the cloud application performance to reduce operations cost for data center owners.

The data center is a server farm consisting of a large number of heterogeneous physical machines connected by a high speed shared network. These physical machines often vary in terms of their computing capacity, composition, and also in their power consumption characteristics at different load conditions. Such heterogeneity in the composition of physical machines results in some of these machines being more power-efficient than others. It is essential to optimize power consumption in the data center by scheduling VMs on more power and performance efficient physical machines. It also identifies and switch off other physical machines with lower power efficiency and having lower utilization during non-peak hours. In this paper, the problem of optimizing power consumption by efficient utilization of heterogeneous physical machines in a data center having an inherent variability in power consumption and performance metrics is evaluated. Optimizing overheads of load balancing algorithms considering the data center load parameter is investigated. Heterogeneity in physical machine's power and performance characteristics along with data center load conditions are denoted in our proposed work as data center context parameters. This work presents a context-aware VM placement optimization technique to reduce the cost of data center management by optimizing power consumption and enhancing performance without affecting the response times of applications. The figure shows the proposed solution.

II. LITERATURE REVIEW

P. Mell, T. Grance, Cloud computing is an evolving technology. The NIST definition characterizes the foremost important aspects of cloud computing. It is also the simplest way for comparisons of cloud services and deployment strategies and is used to providea baseline for discussion of what is cloud computing and also the thanks to best use cloud computing. The cloud services and deployment models defined form a straightforward taxonomy that's not intended to prescribe or constrain any particular method of service delivery, deployment, or business operation. The audience of this document is program managers, system planners, technologists, adopting cloud computing as consumers or providers of cloud services. Cloud computing could even be a model for enabling ubiquitous, on-demand network access to a shared pool of computing resources like networks, servers, storage, applications, and services that will be provisioned and released with minimal management effort or service provider interaction. This cloud model consists of three service models, five essential characteristics, and 4 deployment models. Y. Fukuyama and Y. Nakanishi, Considering voltage stability, a particle swarm optimization for reactive power and voltage control. A control strategy with continuous and discrete control variables such as AVR operating values, OLTC tap positions, and the amount of reactive power compensation equipment is determined by the proposed system. Using a continuation power flow technique the method considers the voltage stability. The feasibility of the proposed method is demonstrated on model power systems with promising results [12].

A. Top of Form

The static scheduling algorithms produce a good schedule given the current state of Grid resources and does not take into account changes in resource availability. In multi-processor systems, critical path heuristics have been used extensively for scheduling interdependent tasks. It determines the longest of all execution paths from the beginning to the end in a task graph and schedules them earliest to minimize the entire graph's execution time. The critical path is dynamically determined after each task is scheduled in the Dynamic Critical Path (DCP) algorithm. For mapping tasks on to homogeneous processors, this algorithm is designed, and is static, in the sense that the schedule is only computed once for a task graph. The DCP algorithm is designed to map and schedule tasks in a workflow on heterogeneous resources in a dynamic Grid environment in this project. It has extensively compared the performance of the algorithm, called DCP-G (Dynamic Critical Path for Grids), against well-known Grid workflow algorithms.

Reactive power and voltage Control (Volt/VarControl: VVC) determines an online control strategy for keeping voltages of target power systems considering varying loads in each load point and reactive power balance in target power systems. Considering execution time and available data from the actual target power system VVC is usually realized based on power flow sensitivity analysis of the operation point. Recently, the voltage stability problem has been dominating. VVC problem has required for the consideration of stability. Continuation power flow (CPFLOW) is suitable for the calculation since the fast computation of voltage stability is required for VVC. The authors have been developed a practical CPFLOW and verified it with an actual power system. With continuous state variables such as AVR operating values and discrete state variables such as OLTC tap positions, VVC can be formulated as a mixed-integer nonlinear optimization problem and the amount of reactive power compensation equipment. According to the power system condition, the objective function can be varied. For example, the function can be a loss minimization of the target power system for the normal operating condition. Conventionally, the methods for the VVC problem have been developed using various methods. However, a practical method for a VVC problem formulated as a mixed-integer nonlinear optimization problem has been eagerly awaited. Particle swarm optimization (PSO) is one of the Evolutionary Computation (EC) techniques. The original method is able to handle continuous state variables easily and search for a solution in a solution space efficiently.

However, the method can be expanded to treat both continuous and discrete variables. Therefore, the method can be applicable to a VVC problem. This paper presents a PSO for a VVC problem formulated as a mixed-integer nonlinear optimization problem considering voltage stability. Voltage stability is considered using a continuation power flow. The feasibility of the proposed method for VVC is demonstrated on a simple power system and IEEE 14 bus system with promising results. M. Rahman, S. Venugopal, and R. Buyya, For the execution of performance-driven Grid applications effective scheduling, is a key concern. The efficient mapping of tasks by calculating the critical path in the workflow task graph at every step is determined by Dynamic Critical Path (DCP) based workflow scheduling algorithm. The algorithm assigns priority to a task in the critical path which is estimated to complete earlier. Using simulation, It has compared the performance of the proposed approach with other existing heuristic and meta-heuristic based scheduling strategies for different types and sizes of workflows. Based on the DCP approach a better schedule is generated for most of the type of workflows irrespective of their size particularly when resource availability changes frequently.

On present-day many of the large-scale scientific applications, executed Grids are expressed as complex e-Science workflows. A workflow could be a set of ordered tasks that are linked by data dependencies. A Workflow Management System (WMS) is usually employed to define, manage and execute these workflow applications on Grid resources. For mapping, the tasks in a very workflow a WMS may use a particular scheduling strategy to appropriate Grid resources so as to satisfy user requirements. Within the literature, numerous workflow scheduling strategies are proposed for various objective functions.

B. Bottom of Form

J. Yu and R Buyya, Grid technologies have progressed towards a service-oriented paradigm based on utility computing models that enable a new way of service provisioning over the last few years. Based on their QoS (Quality of Service) requirements users consume these services. In such "pay-per-use" Grids, during scheduling based on users ‟ QoS constraints workflow execution cost must be considered. In this paper, they propose a budget constraint-based scheduling, which minimizes execution time while meeting a specified budget for delivering results. A new type of genetic algorithm is developed to solve the scheduling optimization problem and test the scheduling algorithm in a simulated Grid testbed. Utility computing has emerged as a new service provisioning model and is capable of supporting diverse computing services such as servers, storage, network and applications for e-Business and e-Science over a global network. Users consume the services when they need to, and pay only for what they use for utility computing-based services. With the economic incentive, utility computing encourages organizations to offer their specialized applications and other computing utilities as services so that other individuals/organizations can access these resources remotely.

To develop their own core activities without maintaining and developing fundamental infrastructure it facilitates individuals/ organizations. Service-oriented Grid computing creates an infrastructure for enabling users to consume services transparently over a secure, shared, scalable, sustainable and standard worldwide network environment. It reinforced providing utility computing services in the recent past. M. Malawski, G. Juve, E. Deelman, and J. Nabrzyski, Inter-related workflows is a group of Large-scale applications expressed as scientific workflows. On Infrastructure-as-a-Service (IaaS) clouds address a new and important problem concerning the efficient management of such ensembles under budget and deadline constraints. To discuss, develop, and assess algorithms based on static and dynamic strategies for both task scheduling and resource provisioning. A set of the scientific workflows used to perform the evaluation via simulation ensembles with a broad range of budget and deadline parameters, taking into account uncertainties in task runtime estimations, provisioning delays, and failures. The ability to decide which workflows in an ensemble to admit or reject for execution is the key factor determining the performance of an algorithm. Based on workflow structure admission procedure estimates of task runtimes can significantly improve the quality of solutions. Scientific workflows, usually represented as Directed Acyclic Graphs (DAGs), are an important class of applications that lead to challenging problems in resource management on grid and utility computing systems. Workflows for large computational problems are often composed of several inter-related workflows grouped into ensembles. Workflows in an ensemble typically have a similar structure, but they differ in their input data, number of tasks, and individual tasksizes.

Scientific workflow ensembles are required by many applications. For example, CyberShake uses ensembles to generate seismic hazard maps. A hazard curve for a particular geographic location is generated by each workflow in a CyberShake ensemble, and several hazard curves are combined to create a hazard map. In 2009 using CyberShake a map is generated that requires an ensemble of 239 workflows. Users of Montage often need several workflows with different parameters is used to generate a set of image mosaics that can be combined into a single, large mosaic. The Galactic Plane ensemble, which generates several mosaics of the entire sky in different wavelengths, consists of 17 workflows, each of which contains 900 sub-workflows. Another ensemble example is the Periodograms application, which searches for extrasolar planets by detecting periodic dips in the light intensity of their host star. Due to the large scale of the input data, this application is often split up into multiple batches processed by different workflows. Additional workflows are created to run the analysis using different parameters. A recent analysis of Kepler satellite data required three ensembles of 15 workflows. Workflows in an ensemble may differ not only in their parameters, but also in their priority. For example, in Cyber-Shake some sites may be in heavily populated areas or in strategic locations such as power plants, while others may be less important. Scientists typically prioritize the workflows in such an ensemble so that important workflows are finished first. This enables them to see critical results early, and helps them to choose the most important workflows when the time and financial resources available for computing are limited. Infrastructure-as-a-Service (IaaS) clouds offer the ability to provision resources on-demand according to a pay-per-use model. These systems are regarded by the scientific community as a potentially attractive source of low-cost computing resources. In contrast to clusters and grids, which typically offer best-effort quality of service, clouds give more flexibility in creating a controlled and managed computing environment. Clouds provide the ability to adjust resource capacity according to the changing demands of the application, often called auto-scaling. However, giving users more control also requires the development of new methods for task scheduling and resource provisioning. Resource management decisions required in cloud scenarios not only have to take into account performance-related metrics such as workflow make span or resource utilization, but must also consider bud-get constraints, since the resources from commercial clouds usually have monetary costs associated with them.

In this paper, it aims to gain insight into resource management challenges when executing scientific workflow ensembles on clouds. It address a new and important problem of maximizing the number of completed workflows from an ensemble under both budget and deadline constraints. Y. Fukuyama and Y. Nakanishi, Cloud computing environments facilitate applications by providing virtualized resources that can be provisioned dynamically. However, users are charged on a pay-per-use basis. Large data retrieval and execution costs incur user applications when they are scheduled taking into account only the „execution time‟. The cost arising from data transfers between re-sources as well as execution costs must be taken into account while optimizing execution time. Particle Swarm Optimization (PSO) based heuristic is presented to schedule applications to cloud resources. It takes into account both computation cost and data transmission cost. In the experiment, with workflow application by varying its computation and communication costs. To

compare the cost savings when using PSO and existing "Best Resource Selection" (BRS) algorithm. PSO can achieve: a) as much as 3 times cost savings as compared to BRS, and b) good distribution of workload on to resources".

III. EXISTING METHODS

The existing systems implement a) Proposed VM Placement Optimization algorithm, b) Power and performance aware Best Fit Decreasing algorithm, c) Under loaded Host Detection Algorithm and d) DC Load Context Detection Algorithm to achieve Context-Aware VM Placement Optimization Technique for Heterogeneous IaaS Cloud platform. The existing system develops a static cost-minimization, a deadline-constrained heuristic for scheduling a workflow application in a cloud environment. This approach considers the features of IaaS providers such as the dynamic provisioning and heterogeneity of computing resources. To achieve this, both resource provisioning and scheduling are merged and modeled as an optimization problem. PSO is then used to solve such a problem and produce a schedule defining the number of nodes that should be assigned. The process referred to in the single cloud provider which is used to compute the consumption time and execution cost for running the process in the environment. The scheduling process is done on the basis of a set of resources, the number of tasks that are defined to that resource in the environment. Here to compute the result of total consumption cost and total execution time using PSO logic. The existing system has following disadvantages,

- Adaptable only in situations where same initial set of resource availability.
- Suitable for single cloud service provider environment only.
- Data transfer cost is not considered between different cloud data centers.

IV. PROPOSED METHOD

The dissertation presented the algorithm named SEMO (Superior Element Multitude Optimization) which compares the entire execution time and total execution cost between one process to a different process. It extends the resource model to think about the information transfer cost between data within the cloud environment in order that nodes are often deployed in several regions. It assigns different options for the choice of the initial resource pool. For the given task, the various set of initial resource requirements is assigned. The information transfer costs between the information environments also are calculated so on minimize the price of execution during a multi-cloud service provider environment. The proposed system has following advantages,

- Adaptable in situations where multiple initial set of resource availability.
- Suitable for multiple cloud service
- Data transfer cost is reduced between different cloud area.
- It reduces management costs for data center owners
- It provides efficient context-aware heuristic-based solution for the VM placement optimization in the heterogeneous cloud data centers
- It improves performance efficiency for data center owners.

V. METHODOLOGIES

This module generates the transfer time matrix in which a number of taken are taken as columns and rows (the square matrix is prepared) and the time a task transfers the data to other tasks is stored as values. The diagonal elements are always zero since the same task has no data transfer operation.

A. Schedule Generation

Initially, the set of resources to lease R and the set of tasks to resource mappings M are empty and the total execution cost TEC and time TET are set to zero. After this, the algorithm estimates the execution time of each workflow task on every resource ri initial. This is expressed as a matrix in which the rows represent the tasks, the columns represent the resources and the entry ExeTimei, j represent the time it takes to run task ti on resource rj. This time is calculated using Figure 1 (a). The next step is the calculation of the data transfer time matrix. Such a matrix is represented as a weighted adjacency matrix of the workflow DAG (Directed acyclic graph) where the entry Transfer Timei, j contains the time it takes to transfer the output data of task ti to task tj. This value is taken from the database and is zero whenever ij or there is no directed edge connecting ti and tj. An example of these matrices is shown in Figure 1 (a) and 1 (b).

$$exeTime = \begin{array}{c} \\ t_1 \\ t_2 \\ t_3 \\ t_4 \\ t_5 \\ t_6 \\ t_7 \\ t_8 \\ t_9 \end{array} \begin{array}{c} \begin{array}{ccc} r_1 & r_2 & r_3 \end{array} \\ \begin{bmatrix} 2 & 1 & 4 \\ 4 & 3 & 6 \\ 10 & 6 & 15 \\ 7 & 4 & 12 \\ 8 & 4 & 10 \\ 3 & 2 & 7 \\ 12 & 7 & 18 \\ 9 & 5 & 20 \\ 13 & 8 & 19 \end{bmatrix} \end{array}$$

Fig 1 (a) Matrix representation of execution time

$$transferTime = \begin{array}{c} \\ t_1 \\ t_2 \\ t_3 \\ t_4 \\ t_5 \\ t_6 \\ t_7 \\ t_8 \\ t_9 \end{array} \begin{array}{c} \begin{array}{ccccccccc} t_1 & t_2 & t_3 & t_4 & t_5 & t_6 & t_7 & t_8 & t_9 \end{array} \\ \begin{bmatrix} 0 & 9 & 9 & 9 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 5 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 7 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 10 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

Fig 1 (b) Matrix representation of transfer time

VI. CONCLUSION

The work presented the SEMO (Superior Element Multitude Optimization) algorithm which is employed to predict the smallest amount time computation within the cloud provider area. Additionally, the thesis compared the time evaluation work between one dynamic resource flows to a different process within the cloud environment. Additionally, it extends the resource model to think about the information transfer cost between data centers in order that node is deployed on different regions. Extending the algorithm to incorporate heuristics that ensure a task is assigned to a node with sufficient memory to execute it'll be included within the algorithm. Also, it assigns different options for the choice of the initial resource pool. As an example, for the given task, the various set of initial resource requirements is assigned. Additionally, data transfer cost between data centers also are calculated so on minimize the value of execution in multi-cloud service provider environment. The main contribution of thesis, the following problem solve in the existing system, they contribution are

- Adaptable in situations where multiple initial set of resource availability.
- Suitable for multiple cloud service provider environments.
- Data transfer cost is reduced

The system is extremely flexible and user-friendly; that the maintenance supported the changing environment and requirements is incorporated easily. Any changes that are likely to cause failures are prevented with security and preventive measures may well be taken. The coding is finished in understandable and versatile method program which helps easy changing. Since MS-SQL Server and Java are very flexible tools, user can easily incorporate any modular program within the application.

REFERENCES

[1] G. Thickins, “Utility Computing: The Next New IT Model”, Darwin Magazine, April 2003.

[2] H. D. Chiang, et al., "CPFLOW: A Practical Tool for Tracing Power System Steady-State Stationary Behavior Due to Load and Generation Variations", IEEE Trans. on Power Systems, Vol. 10, No. 2, May 1995.

[3] H. Yoshida, Y. Fukuyama, et al., "Practical Continuation Power Flow for Large-Scale Power System Analysis", Proc. of IEE of Japan Annual Convention Record, No. 1313, 1998 (in Japanese).

[4] J. Kennedy and R. Eberhart, "Particle Swarm Optimization", Proc. of IEEE International Conference on Neural Networks, Vol. IV, pp.1942-1948, Perth, Australia, 1995.

[5] J. Yu and R Buyya, “A budget constrained scheduling of workflow applications on utility grids using genetic algorithms,” in Proc. 1st Workshop Workflows Support Large-Scale Sci., pp. 1–10, 2006.

[6] J. Yu and R. Buyya, “Taxonomy of Workflow Management Systems for Grid Computing”, Journal of Grid Computing, 3(3-4): 171-200, Springer, New York, USA, Sept. 2005.

[7] J. Yu and R. Buyya, “Workflow Scheduling Algorithms for Grid Computing”, Tech. Rep., GRIDS-TR-2007-10, University of Melbourne, Australia.

[8] M. Rahman, S. Venugopal, and R. Buyya, “A dynamic critical path algorithm for scheduling scientific workflow applications on global grids,” in Proc. 3rd IEEE Int. Conf. e-Sci. Grid Comput., pp. 35–42, 2007.

[9] P. Mell, T. Grance, “The NIST definition of cloud computing- recommendations of the National Institute of Standards and Technology” Special Publication 800-145, NIST, Gaithersburg, 2011.

[10] S. Kim and J. Browne, “A General Approach to Mapping of Parallel Computation upon Multiprocessor Architectures”, Proceedings of IEEE International Conference on Parallel Processing, IEEE press, 1988.

[11] T. Eilam et al., “A utility computing framework to develop utility systems”, IBM System Journal, vol. 43, no. 1, pp. 97-120, 2020.

[12] Y. Fukuyama and Y. Nakanishi, “A particle swarm optimization for reactive power and voltage control considering voltage stability,” in Proc. 11th IEEE Int. Conf. Intell. Syst. Appl. Power Syst., pp. 117–121, 1999.