Available online at www.worldscientificnews.com

World Scientific News

An International Scientific Journal

WSN 41 (2016) 55-61 EISSN 2392-2192

A Survey on Resource Provisioning Heuristics

M. Gowthami*, V. Suganya

Department of Computer Science and Engineering, Avinashilingam Institute of Home Science and Higher Education for Women, Coimbatore, India

*E-mail address: gowthamimaruthachalam@gmail.com

ABSTRACT

Cloud Computing allow the users to dynamically provide computing resource to meet their information technology needs. Cloud providers are able to rent resources from cloud for many computational purposes using any one of the provisioning model static/dynamic the company can pay bills as per the model. One of the major drawbacks in cloud computing is linked to step up the dataflow model with better resources. Resource allocation is performed with the aim of minimizing the costs and to get high throughput. The added challenges in resource allocation is satisfying the customer demands and application necessities. In this paper, we have presented an extensive survey on various resource allocation strategies for various dataflow model and their challenges are discussed in detail.

***Keywords*:** cloud computing, information technology, resource allocation

1. INTRODUCTION

Cloud computing is a computer paradigm where data and services reside in massive scalable data centers in the cloud and can be used with any connected devices over the internet. It is a way of providing various services on virtual machine allocated on a larger physical machine which resides in the cloud. Cloud Computing enables users to acquire resources dynamically and elastically. One of the major challenges in resource provisioning technique is to determine the right amount of resources required for the executing the task in order to reduce the financial cost from the user's point of view and to maximize the resource utilization from

(Received 01 February 2016; Accepted 10 February 2016; Date of Publication 11 February 2016)

the service providers side So, Cloud computing is one of the preferred options in today's venture.

Resource provisioning is the process of selecting, deploying, and run-time management of software (e.g., database management servers, load balancers) and hardware resources such as CPU, storage, and network for ensuring the guaranteed performance. This resource provisioning considers the Service Level Agreement (SLA) for providing services to the cloud users. This is like an initial agreement between the cloud users and cloud service providers to ensure the Quality of Service (QoS) parameters like performance, availability, power consumption, reliability, response time, etc. Based on the use and application needs Static Provisioning/Dynamic Provisioning and Static/Dynamic Allocation of resources have to be made in order to efficiently make use of the resources without violating Service Level Agreement and meeting these QoS parameters. Over provisioning/under provisioning of resources must be avoided. Another important constraint is power consumption. It is necessary to reduce power consumption, power dissipation and also on VM placement. Some techniques to avoid excess power consumption are used.

So the main goal of the cloud user is to reduce cost by renting the resources and from the point of the cloud service provider's to increase the profit by efficiently allocating the resources. In order to achieve the main goal of the cloud user has to ask for cloud service provider in advance to make a provision for the resources by statically or dynamically so that the cloud service provider will know what and how much resources are required for a given application. By provisioning the resources, the QoS parameters like availability, throughput, security, response time, performance, reliability must be achieved without violating SLA.

The development of the Internet of Things, has increased the rate and quantity of data being generated endlessly hence there is a need to analyze and manage such high velocity data in real-time forms. For this high dataflow a proper resource need to be provided for execution. The various processing techniques are discussed below:

2. LITERATURE REVIEW

J. Dean and S. Ghemawat [1] proposes map reduce encoding model for processing and generating large data sets which automatically execute parallel on a large cluster of commodity machines. The run-time system takes care of the details of partitions in the input data, scheduling the program's execution across a set of machines, handling machine failures, and managing the required inter-machine communication.

M. Zaharia, T. Das, et al. [2] explain discretized streams (D-Streams), a stream programming model for large clusters that provides high consistency and powerful integration with batch systems. The spark streaming is to make streaming as a series of short batch jobs, and bring down the latency of these jobs as much as possible. This new parallel recovery technique that makes the system to be cost-efficient for stream processing in large cluster.

L. Neumeyer, B. Robbins et al., [3] proposes the S4 (Simple Scalable Streaming System) architecture engine to solve problems in the circumstance of exploring application that use data mining and machine learning algorithms. The S4 design Provide a plain Programming Interface for processing data streams by using local memory in each processing node and avoiding disk Input/output bottlenecks also making cluster with high availability that can scale using commodity hardware.

B. Satzger, W. Hummer et al., [4] presented a new stream computing platform known as elastic stream cloud (esc) an easy programming model for programmers based on DAGs defines the data flow of a program, whose vertices represents a operations apply to the data. The data that are streaming through the graph are expressed as key/value pairs. This system hides complexity near-term along with distribution and fault tolerance. By making the system to dynamically adapt to changing workloads based on a high frequency trading scenario.

M. Maheswaran, S. Ali et al., [5] proposes Dynamic mapping (matching and scheduling) heuristics for a class of autonomous tasks using heterogeneous distributed computing systems using Five immediate mode heuristics (HiHi, KPB, MCT, OLB and the MET) and three batch mode heuristics (Sufferage, Min-min, HiHi) shows OLB is slightly better than other immediate modes and Sufferage is better than others

Z. Wu, X. Liu et al., [6] presented a market-oriented hierarchical scheduling tactic in cloud workflow systems using the candidate service-level scheduling algorithm which utilize three meta-heuristic based scheduling algorithms as well as genetic algorithm (GA), ant colony optimization and particle swarm optimization are adapted and the results show ACO performs better the other meta-heuristics.

F. Zamfirache, M. Frincu et al., [7] presented a simple population based method for task scheduling in heterogeneous distributed systems using hybrid perturbation operator which combines greedy and random strategies in order to make sure local improvement of the schedules and compares against sps (Simple Population Scheduler) algorithm shows hybrid perturbation works better in task scheduling.

R. Castro Fernandez, M. Migliavacca et al., [8] shows SPS (stream processing system) approach which integrates scaling in which SPS partitions operators on demand in response to bottleneck operators and in failure recovery the sps maintains two versions of operator o's state, which could be partitioned for scale out: the current state, maintained by o, and a recent state checkpoint stored by operator backup(o)through explicit state management of stateful operators. This approach treats operator state as an independent entity such as checkpoint, backup, restore besides these shows sps works well in scaling out.

R. Tolosana-Calasanz, J. Angel Banares et al., [9] utilize the token bucket model that stores data elements from a stream and then forwards them to the computational phase of a work flow stage at a predefined rate over a shared Cloud infrastructure while providing each workflow with a particular QoS requirement thereby adding control strategy at each workflow stage to dynamically adjust Token Bucket parameters to adapt the resources available so that variations in data size (data inflation/deflation) between stages can be self configured by the application

Andres Quiroz, Hyunjoo Kim et al., [10] work on a decentralized healthy online clustering approach used to detect patterns and trends in distributed environment and used it to optimize provisioning of virtual (VM) resources it also presents a model-based approach to estimate application service time using the long-term application performance monitor to provide feedback about the suitability of requested resources as well as the system ability to meet Quality of service constraints and Service Level Agreement.

Ye Hu, Johnny Wong et al., [11] proposes heuristic algorithm to find out a resource allocation strategy (Shared or Dedicated) which consider the processing of interactive jobs only where global arbiter makes resource allocation decision, information on the number of servers that should be allocated to each Application Environment would be very helpful making it to use the smallest number of servers required to meet the Service Level Agreement of both classes

and then did a comparative evaluation of first come first serve (FCFS), head-of-the-line priority (HOL) and a new scheduling regulation called probability dependent priority (PDP).

Saeid Abrishami, Mahmoud Naghibzadeh et al., [12] utilizes Partial Critical Paths (PCP), which aims to reduce the cost of workflow execution with a user defined deadline by mapping each task to an appropriate resource and of ordering the tasks on each resource to satisfy some performance criterion and it consists of two phases one-phase algorithm which is called Infrastructure as a service Cloud Partial Critical Paths (IC-PCP), and a two-phase algorithm which is called Infrastructure as a service Cloud Partial Critical Paths with Deadline Distribution (IC-PCPD2) have a polynomial time complexity which makes them suitable options for scheduling large workflows and the result shows IC-PCP performing better.

W. Dawoud, I. Takouna et al., [13] Microeconomic Inspired Approach to determine the number of Virtual Machine allotted to each user according to user financial capacity there by mechanically adjusting to the ever changing equilibrium point caused by active workloads and ensures that resources are shared proportionally by continuously monitoring the response time of user application. Ming Mao, Marty Humphrey [14] proposes auto-scaling mechanism which finishes all jobs by user specified deadlines in a cost-effective way based on a monitor-control loop that adapts to dynamic changes such as the workload bursting and delayed instance acquisitions there by eliminating user instances. The major drawback is it does not task with increased size.

According to K. Tsakalozos, H. Kllapi et al., [15] Genetic Algorithm (GA) which is designed and implemented to compute the optimized system state, i.e., VM-to-node mapping and the resource capacity allocated to each VM, so as to optimize resource consumptions but it doesn't provide optimized state when the node increases.

The technique proposed by R. Jeyarani, N. Nagaveni [16] makes use of the Provisioner called adaptive power-aware virtual machine provisioning (APA-VMP) where the resources are provisioned varying from a large resource pool. This is from Iaas provider point of view where the traditional Virtual achines (VM) are launched in appropriate server in a data center. The cloud data center considered here is heterogeneous and large scale in nature. The proposed meta scheduler maps efficiently a set of Virtual Machine instances onto a set of servers from a highly dynamic resource pool by fulfilling resource requirements of maximum number of workloads.

3. TECHNIQUES USED

The various techniques and algorithms were used for effective handling of resource provisioning for dynamic flow of data. These methods were used for widely used for searching solution space and outperforms well in provisioning resources.

3. 1. Partial Critical Path

In the PCP [12] scheduling algorithm, the critical path and partial critical paths of the whole workflow is to be found. In order to find these, some idealized, notion of the start time of each work flow task are needed before scheduling of the tasks is done. This means that two notions of the start times of tasks are available, the earliest start time computed before scheduling the work flow, and the actual start time computed by the scheduling algorithm. For each unscheduled task the Earliest Start Time (EST) is found as the earliest time can start its

computation despite of the actual service that will process the task which is determined during scheduling Since grid is a heterogeneous environment and the computation time of tasks varies from service to service the EST cannot be found exactly.

3. 2. Genetic Algorithm

Genetic Algorithm [15] finds the survival of the fittest among individuals over successive generation for solving a problem. Each individual indicate a point in a search space and a possible solution. The individuals in the population are then performs a process of evolution. Genetic Algorithm is based on an analogy with the genetic structure and behavior of chromosomes within a population of individuals using the following foundations:

- Individuals in a population compete for resources and mates.
- Those most successful individuals in each competition will produce more offspring than those individuals that perform badly.
- Genes from good individuals propagate throughout the population so that two high-quality parents will sometimes create offspring that are better than either parent.
- Thus, each successive generation will become more suited to their environment.

3. 3. MicroEconomic-Inspired Approach

In infrastructure as a Service (IaaS) how many virtual machines (VMs) a user should request from an IaaS Cloud specified that users have an inadequate budget and that there are speed-up barriers set by the available physical resources. Follow a microeconomic-inspired approach [13] to determine the number of VMs allotted to each user according to user financial capacity. Since the core physical resources are shared among all cloud tenants, the performance the users get out of the cloud may significantly vary over time. Therefore, this approach continuously monitors the response time of user applications and adjusts the amount of resources accordingly. At its equilibrium point, the suggested approach maximize profit. From the provider's point of view this profit corresponds to financial benefit whereas from the consumer's point of view, the same profit corresponds to quality of service received.

3. 4. Particle Swarm Optimization (PSO)

Particle swarm optimization (PSO) [17] is a computational method that optimizes a given problem by repeatly trying to enhance the candidate solution in terms of quality. PSO optimizes a problem by having a population of candidate solutions (agents) and moving these particles around in the search-space according to simple mathematical formula over the agents position and velocity. Each agent movement is influenced by its local best known position but, is also guided to the best recognized positions in the search-space, the better positions are found by other particles are updated. This is expected to move the swarm toward the best solutions.

3. 5. Tabu Search

Tabu Search [18] is a meta-heuristic that guides a local heuristic search procedure to explore the solution space outside local optimality. One of the major components of Tabu Search is its use of adaptive memory, which creates a more elastic search actions. Memory-based strategies is most important of tabu search approaches, founded on a quest for integrating principles by which alternative forms of memory are appropriately combined with effective

strategies for exploiting them. To address the problem related with training multilayer feed forward neural networks. These networks have been broadly used for prediction as well as classification in many different areas.

4. CONCLUSION

Cloud computing has been the playing a major role in a distributed computing system due to the way the resource provisioning and charging. Providing and Managing Resource is a crucial task with various innovative technology for small to large needs. Many researchers have put forward their ideas for new and innovative solutions for handling such an imperative area. In this paper, we have carried out a decisive review of the most recent work carried out in this area.

References

[1] J. Dean and S. Ghemawat, MapReduce: Simplified data processing on large clusters. *ACM Commun.* Vol. 51, no. 1, pp. 107-113, 2008

[2] M. Zaharia, T. Das, H. Li, S. Shenker, and I. Stoica, Discretized streams: an efficient and fault-tolerant model for stream processing on large clusters, in Proc. *USENIX Conf. Hot Topics Cloud Comput.* 2012, p. 10.

[3] L. Neumeyer, B. Robbins, A. Nair, and A. Kesari, S4: Distributed stream computing platform, in Proc. *IEEE Int. Conf. Data Min. Workshops*, 2010, pp. 170-177

[4] B. Satzger, W. Hummer, P. Leitner, and S. Dustdar, Esc: Towards an elastic stream computing platform for the cloud, in Proc. *IEEE Int. Conf. Cloud Comput.*, Jul. 2011, pp. 348-355

[5] M. Maheswaran, S. Ali, H. J. Siegel, D. Hensgen, and R. F. Freund, Dynamic mapping of a class of independent tasks onto heterogeneous computing systems, in *J. Parallel Distrib. Comput.* Vol. 59, no. 2, pp. 107-131, 1999

[6] Z. Wu, X. Liu, Z. Ni, D. Yuan, and Y. Yang, A market-oriented hierarchical scheduling strategy in cloud workflow systems, *J. Supercomput.* Vol. 63, no. 1, pp. 256-293, 2013

[7] F. Zamfirache, M. Frincu, and D. Zaharie, Population-based metaheuristics for tasks scheduling in heterogeneous distributed systems, in Proc. *7th Int. Conf. Numerical Methods Appl.* 2011, Vol. 6046, pp. 321-328

[8] R. Castro Fernandez, M. Migliavacca, E. Kalyvianaki, and P. Pietzuch,Integrating scale out and fault tolerance in stream processing using operator state management, in Proc. *ACM SIGMOD Int. Conf. Manage.* Data, 2013, pp. 725-736

[9] R. Tolosana-Calasanz, J. Angel Ba~nares, C. Pham, and O. Rana, End-to-end qos on shared clouds for highly dynamic, large-scalesensing data streams, in Proc. *IEEE/ACM Int. Symp. Cluster, Cloud Grid Comput* 2012, pp. 904-911

[10] A. Quiroz, H. Kim, M. Parashar, N. Gnanasambandam and N. Sharma, Towards autonomic workload provisioning for enterprise Grids and clouds, *2009 10th IEEE/ACM International Conference on Grid Computing*, 2009, pp. 50-57, doi: 10.1109/GRID.2009.5353066

[11] Ye Hu, Johnny Wong, Gabriel Iszlai and Marin Litoiu, Resource Provisioning for Cloud Computing, Proceedings of the 2009 *Conference of the Center for Advanced Studies on Collaborative Research*, Pages 101-111, 2009

[12] Saeid Abrishami, Mahmoud Naghibzadeh, Dick H.J. Epema, Deadline-constrained workflow scheduling algorithms for Infrastructure as a Service Clouds, *Future Generation Computer Systems*, Volume 29, Issue 1, January 2013, Pages 158-169

[13] W. Dawoud, I. Takouna, and C. Meinel, Infrastructure as a Service Security: Challenges and Solutions, in Proc the *7th International Conference on Informatics and Systems 2010 (INFOS'10)*, Cairo, March 2010, pp. 1-8

[14] M. Mao and M. Humphrey, Auto-scaling to minimize cost and meet application deadlines in cloud workflows, *SC '11: Proceedings of 2011 International Conference for High Performance Computing, Networking, Storage and Analysis*, 2011, pp. 1-12

[15] K. Tsakalozos, H. Kllapi, E. Sitaridi, M. Roussopoulous, D. Paparas, and A. Delis, Flexible Use of Cloud Resources through Profit Maximization and Price Discrimination, in Proc of the *27th IEEE International Conference on Data Engineering (ICDE 2011),* April 2011, pp.75-86.

[16] R. Jeyarani, N. Nagaveni, R. Vasanth Ramc, Design and implementation of adaptive power-aware virtual machine provisioner (APA- VMP) using swarm intelligence, *Journal of Future Generation Computer Systems*, Volume 28, Issue 5, May 2012, pp. 811-821, DOI:/10.1016/j.future.2011.06.002

[17] J. Kennedy and R. Eberhart, Particle swarm optimization, in Proc. *IEEE Int. Conf. Neural Netw.* 1995, Vol. 4, pp. 1942–1948.

[18] I. De Falco, R. Del Balio, E. Tarantino, and R. Vaccaro, Improving search by incorporating evolution principles in parallel tabu search, in *1994 IEEE Conference on Evolutionary Computation*, Vol. 2, pp. 823-828, 1994