Framework for Efficient Job Scheduling in Cloud Environment

S.Venkadalakshmi[1] Dr.K.Ruba Soundar[2]
[1]PG Student [2]Professor & HOD
[1, 2] CSE Dept.
[1, 2] P. S. R Engineering College, Sivakasi, India

Abstract---Cloud computing is known as digital service delivery over the Internet by several applications which are carried out by computer systems in distributed datacenters. It supplies a high performance computing based on protocols which allow shared computation and storage over long distances. Managing the allocation of cloud virtual machines at physical resources is a key requirement for the success of clouds. It aims at spreading the workloads among the processing resources in an optimal fashion to reduce the total execution time of jobs and then, to improve the effectiveness of the whole cloud computing services. A new Bee Swarm optimization algorithm called Bees Life Algorithm (BLA) is applied to efficiently schedule computation jobs among processing resources onto the cloud datacenters. The algorithm mimics the food foraging behavior of swarms of honey bees. The algorithm performs a kind of neighborhood search combined with random search and optimization problems.

Key words: Data center, job scheduling, bee's life algorithm

I. INTRODUCTION

The cloud computing is a recent field in the computational intelligence techniques which aims at surmounting the computational complexity and provides dynamically services using very large scalable and virtualized resources over the Internet. It is defined as a distributed system containing a collection of computing and communication resources located in distributed datacenters which are shared by several end-users. There are two kinds of the cloud; the former is the public cloud in which services may be sold to anyone on the Internet. Here, Amazon Elastic Compute Cloud (EC2), Google App Engine is large public cloud providers. The second type of the cloud is the private cloud. It is a proprietary network or a datacenter that supplies hosted services to a limited number of clients (end-users).

The cloud computing model allows access to information and computer resources from anywhere that a network connection is available. Cloud computing provides a shared pool of resources, including data storage space, networks, computer processing power, and specialized corporate and user applications.This cloud model promotes availability and is composed of five essential characteristics:

1) On-demand self-service - Users can order and manage services without human interaction with the service provider, using, for example, a Web portal and management interface. Provisioning and de-provisioning of services and associated resources occur automatically at the provider.
2) Ubiquitous network access - Cloud services are accessed via the network (usually the Internet), using standard mechanisms and protocols.
3) Resource pooling - Computing resources used to provide the cloud services are realized using a homogeneous infrastructure that's shared between all service users.
4) Rapid elasticity- Resources can be scaled up and down rapidly and elastically.
5) Measured service - Resource/service usage is constantly metered, supporting optimization of resource usage, usage reporting to the customer, and pay-as-you-go business models.

Cloud computing can offer an organization several benefits but as with most things, there are also disadvantages to consider when making decisions about using cloud services or choosing a cloud provider. Due diligence is required in each situation involving a cloud computing decision to ensure benefits are maximized and the risks of the associated disadvantages are mitigated. Some of the disadvantages can be addressed as legal contract issues.

Cloud computing comes in three different services models, Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS). Each model has advantages and disadvantages and no one model can serve all the needs of Johnson County. Each service model offers differing levels of control for the organization. The service models are: SaaS – Software as a Service, PaaS – Platform as a Service, IaaS – Infrastructure as a Service.

Cloud services can be implemented in a multitude of different configurations; private cloud, community cloud, public cloud, and hybrid cloud. These are called cloud deployment models and an organization can utilize more than one model.

Any organization preparing to use a cloud computing resource has many things to consider and decisions to make. While actually purchasing these services is in many cases extremely easy, answers to the question of who owns the data, what kind of security provisions are in place, and how is the service accessed are just the beginning of several issues to be addressed before using a cloud solution. Other issues include cultural factors, staff skills, cost, and vendor experience, impact to the existing technology environment, legal issues, and even the use of personal cloud services by individual employees.

The use of cloud services and technologies is an issue that every organization must address. This issue will be forced by vendors that only offer their products as cloud services and by the increasing use of personal devices by employees. Organizations may choose to limit their use of cloud technologies, but will miss out on the many benefits these services have to offer.

A well thought out strategy for using cloud resources is the best plan for moving forward and exploiting all this emerging technology has to offer. Organizations that take this approach will be in the best position to take advantage of this shift in computing technology. However, time is of the essence. Vendors and employees will not wait for an organization to make decisions about cloud computing. These two groups will move to fill the vacuum created by any lack of direction or corporate strategy in this area, perhaps with negative consequences for the organization.

II. OPEN STACK

The central component that manages the allocation of virtual resources of a cloud infrastructure's physical resources is known as the cloud scheduler. A cloud scheduler should consider the application's performance requirements and user's security and privacy requirements. We introduce a trustworthy cloud scheduler whom we call Access-Control-as-a Service (ACaaS).A novel cloud scheduler which considers both user requirements and infrastructure properties. We focus on assuring users that their virtual resources are hosted using physical resources that match their requirements without getting users involved with understanding the details of the cloud infrastructure. As a proof-of-concept, we present our prototype which is built on OpenStack. OpenStack refers to its cloud scheduler component using the name "nova-scheduler" .It identifies the scheduler as the most complex component to develop and states that significant effort still remains to have an appropriate cloud scheduler. OpenStack is an open source tool for managing the cloud infrastructure which is under continuous development, (Access Control as a Service), which performs the following when allocating a physical resource to host a virtual resource: i) Considers the discussed cloud taxonomy, ii) Selects a physical resource which has properties that can best match the requested user requirements, iii) Ensures that the user requirements are continually maintained.

A. *Open STAC Prototype Implementation*

A mechanism for a trustworthy collection of resources' RCoT, and calculates for each group of resources their DCoT and CDCoT. It then uses the ACaaS scheduler to match user properties with infrastructure properties.The trust measurements performed by the DC-C identifies the building up of a resource's RCoT and its integrity measurements. It involves three stages: 1) Trust Attestation via the DC-C, 2) Trust Management by the DC-S,3) Preliminary Performance Evaluation.

B. *Acaas Scheduler*

We introduce a trust-worthy cloud scheduler which we call Access-Control-as-a-Service (ACaaS). ACaaS is a novel cloud scheduler which considers both user requirements and infrastructure properties. It focuses on assuring users that their virtual resources are hosted using physical re-sources that match their proper-ties without getting users involved with understanding the details of the complex cloud infrastructure.

A disadvantage is that it does not provide flexibility. A certain entity is bound to the access provided by the role they are in. More often than not there are exceptions in the access needs of an entity. It would be rare that very large groups of entities would all need the exact same access.

III. RELATED WORKS

R. Chow, P. Golle, M. Jakobsson, E. Shi, J. Staddon, R. Masuoka, and J. Molina [1] Proposed an Cloud computing is clearly one of today's most enticing technology areas due, at least in part, to its cost-efficiency and flexibility. However, despite the surge in activity and interest, there are significant, persistent concerns about cloud computing that are impeding momentum and will eventually compromise the vision of cloud computing as a new IT procurement model. In this work, characterize the problems and their impact on adoption. In addition, and equally importantly, we describe how the combination of existing research thrusts has the potential to alleviate many of the concerns impeding adoption. In particular, we argue that with continued research advances in trusted computing and computation-supporting encryption, life in the cloud can be advantageous from a business intelligence standpoint over the isolated alternative that is more common today. Research in trusted computing and computations supporting encryption in the cloud. New problem areas in security that arise from cloud computing. Cheap data and data analysis. The rise of cloud computing has created enormous data sets that can be monetized by applications such as advertising.

S. Joshua, M. Thomas, J. Trent, and M. Patrick [2] Proposed an Cloud verifier service that generates integrity proofs for customers to verify the integrity and access control enforcement abilities of the cloud platform that protect the integrity of customer's application VMs in IaaS clouds. A significant system bottleneck, we demonstrate that aggregating proofs enables significant overhead reductions. A cloud- wide verifies sets vice could present a significant system bottleneck; we demonstrate that aggregating proofs enables significant overhead reductions. As a result, transparency of data security protection can be verified at cloud-scale.Using an attested time server to provide nonces. Handle over 7,000 requests per second. Current integrity measurements approaches are very system configuration specific are difficult to assess arbitrary data and custom code.

P. Bryan, M. M. Jonathan, and P. Adrian [3] Trusting a computer for a security-sensitive task requires the user to know something about the computer's state. We examine research on securely capturing a computer's state, and consider the utility of this information both for improving security on the local computer (e.g., to convince the user that her computer is not infected with malware) and for communicating remote computer's state (e.g., to enable the user to check that a web server will adequately protect her data). Although there cent "Trusted Computing" initiative has drawn both positive and negative attention to this area, we consider the older and broader topic of bootstrapping trust in a computer. We cover issues ranging from the wide collection of secure hardware that can serve as a foundation for trust, to the usability issues that arise when trying to convey computer state information to humans. This approach unifies disparate research efforts and highlights opportunities for additional work that can guide real-world improvements in computer security**.**

S. Bleikertz, and K.Eriksson [4] the highly dynamic nature of the cloud environment leads to a time-varying resource utilization and the cloud provider can potentially accommodate secondary jobs with the remaining resource. To better implement the idea of resource reutilization in the cloud environment, the problem of secondary job scheduling with deadlines under time-varying resource capacity is considered in this paper. A transformation is proposed to reduce the offline problem with time varying processor capacity to that with constant capacity.

For online scheduling of under loaded system, it is shown that the earliest deadline first (EDF) scheduling algorithm achieves competitive ratio For the overloaded system, an online scheduling algorithm V-Dover is proposed with asymptotically optimal competitive ratio when a certain admissibility condition holds .It is further shown that, in the absence of the admissibility condition, no online scheduling algorithm exists with a positive competitive ratio. Simulation results are presented to illustrate the performance advantage of the proposed V-Dover algorithm.

Online algorithm exists with positive competitive ratio if the admissibility condition is relaxed. The gain is not significant for small λ since there are not many supplement jobs scheduled by V-Dover; for large the gain is not significant either, since only a small portion of the supplement jobs scheduled by V-Dover is finished.

IV. Bees Life Algorithm

In this paper, we propose to First, the life of bees in nature is overviewed as an inspiration source of this algorithm. The bees share a communication language of extreme precision, based on two kinds of dances: the round dance when food is very close. They are carried out when bees search food. The bees' reproduction is guaranteed by the queen. It will mate with several males in full flight, until her spermatheca is full. The unfertilized egg will give rise to a drone, while, the fertilized egg gives rise to worker or queen depending on food quality.

Bees in Nature

Bees Life Algorithm starts with bee population initialization step which contains N bees (individuals) chosen randomly in the search space. The population fitness is evaluated in the second step. A bee population contains *'1'* queen, *'D'* drones and *'W'* workers in which the fittest bee represents the queen, the *'D'* fittest following bees represent the drones and the remaining bees are the workers. Consequently, the sum of the different bee individuals *(1, D and W)* equal to the population size *(N)*. *'D'* and *'W'* are considered as a two user-defined parameters. Each cycle of a bee population life consists of two bee behaviors: reproduction and food foraging respectively. In reproduction behavior, the queen starts mating in the space by mating-flight with the drones using crossover and mutation operators. Next, queen starts breeding *'N'* broods in step 4. Then, the evaluation of the brood fitness is performed (steps 5). If the fittest brood is fitter than the queen, it will be considered as the new queen for the next population. Moreover, *'D'* best bee individuals are chosen among the *'D'* fittest following broods and the drones of the current population to form the drones of the next population.

After that, *'W'* best bee individuals are chosen among the *'W'* fittest remaining broods and the workers of the current population in order to ensure the food foraging (steps 6 to 8). In step 9, the *'W'* workers search food source in *'W'* regions of flowers. We consider that each worker represents one region and there are other bees for each region recruited and employed to search the best food source among the different food sources in the region (step 10). The recruited bees represent neighbor solutions in the search space used to ensure neighborhood search. BLA uses more recruited bees for the *'B'* best regions among *'W'* regions. *'B'* is user-defined parameter. For each region in step 11, only the bee with the highest fitness will be selected to form the next bee population. The evaluation of the new population fitness is executed in step 12. If the stopping criterion is not satisfied, a new bees' life cycle is performed, and then we rerun the third step and so on.

1) Initialize population (*N* bees) at random
2) Evaluate fitness of population (fittest bee is the queen, *D* fittest following bees are drones, *W* fittest remaining bees are workers)
3) *While* stopping criteria are not satisfied (Forming new population)
   a. /* reproduction behavior */
4) Generate *N* broods by crossover and mutation
5) Evaluate fitness of broods
6) If the fittest brood is fitter than the queen then replace the queen for the next generation
7) Choose *D* best bees among *D* fittest following broods and drones of current population (Forming next generation drones)
8) Choose *W* best bees among *W* fittest remaining broods and workers of current population (to ensure food foraging)
   a. /* food foraging behavior */
9) Search of food source in *W* regions by *W* workers
10) Recruit bees for each region for neighborhood search (more bees *(FBest)* for the best *B* regions and *(FOther)* for remaining regions)
11) Select the fittest bee from each region
12) Evaluate fitness of population (fittest bee is the queen, *D* fittest following bees are drones, *W* fittest remaining bees are workers)
13) *End while*

V. Simulation Results

The main appeal of the scheduling a job is an optimal allocation for each user. And provide datacenter to availability of the user running their jobs. Optimization is to provide a best solution for the users.The user select a job and to allocate their jobs randomly allocating in datacenters. The cloud server is using many third party servers. Allocating jobs to the datacenter to fit their jobs in a particular datacenter. But the jobs are fit or unfit in their datacenters.

A fitness evaluation unit and genetic operators for reproduction; crossover and mutation operations. Associated with each string is a fitness value computed by the evaluation unit. A fitness value is a measure of the goodness of the solution that it represents. The aim of the genetic operators is to transform this set of strings into sets with higher fitness values. The minimum resource requirement job which acts as a scout bee are identified and sent to the cluster where in the jobs identify the instances present.. A scout job identifies the site by using a fitness function which runs that job in a particular instance and if a progress is made it determines that instance specification as in it is either memory oriented or processor oriented.

Conceptually, fitness refers to how much progress each job is making with assigned resources compared to the same job running on the entire cluster. Bees Algorithm starts with a number of scout bees being placed randomly in the search space. The best sites visited in point of view fitness are chosen as selected bees which will be foraged their close sites to carry out neighborhood search. And only the bee with the highest fitness will be selected to form the next bee population.

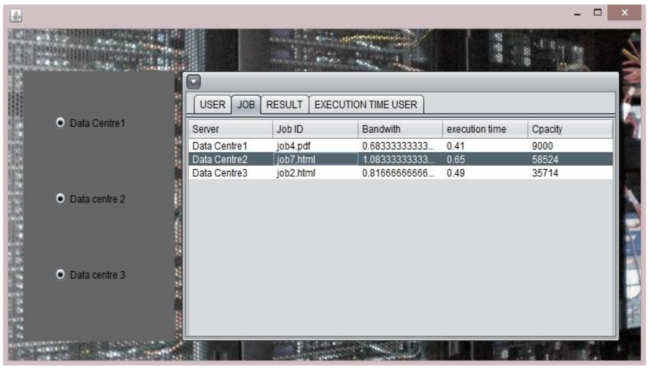

Fig. 1: Job details in data centers

The proposed scheme for a data centers to execute their jobs and the fitness evaluation is calculated in their related execution time from figure 2.

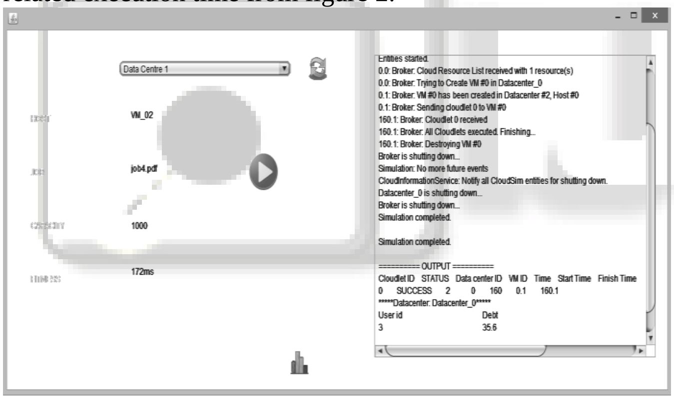

Fig. 2: Calculate fitness value in data centers

The proposed scheme for an increase in the number of data centers to execute the jobs and the execution time to the users from figure .3. It is evident that a minimum time and user execute their jobs in different data centers.

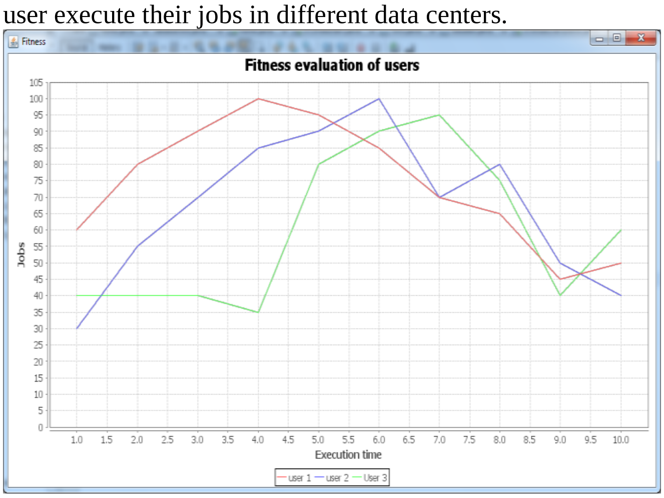

Fig. 3: Fitness Calculation

VI. Conclusions And Further Work

Scheduling is one of the most important jobs in cloud computing environment. In this paper we have analyze Bees life algorithm scheduling. We have noticed that disk space management is critical issue in virtual environment. Existing scheduling algorithm gives high throughput and cost effective but they do not consider reliability and availability. So we need algorithm that improves availability and reliability in cloud computing environment. The presented work is about to perform the scheduling and the allocation of the processes to the clouds in case of fit or unfit conditions. The Future enhancement of the work is possible in the following directions optimal way of scheduling for users and using a reproduction and food foraging behavior in bee's life algorithm.

References

[1] R. Chow, P. Golle, M. Jakobsson, E. Shi, J. Staddon, R. Masuoka, And J. Molina, "Controlling Data In The Cloud: Outsourcing Computation Without Outsourcing Control," In Proc. 2009 Acm Workshop On Cloud Comput. Security (Ccsw '09), New York, Ny, Usa, 2009, Pp.85–90, Acm.

[2] S. Joshua, M. Thomas, J. Trent, and M. Patrick, "Seeding Clouds with Trust Anchors," In Acm Workshop on Cloud Comput. Security in Proc. 2010 Acm Workshop on Cloud Comput. Security.

[3] S. Joshua, M. Thomas, J. Trent, and M. Patrick, "Seeding Clouds with Trust Anchors," In Acm Workshop on Cloud Comput. Security in Proc. 2010 Acm Workshop on Cloud Comput. Security.

[4] S. Bleakest, And Eriksson, "Secondary Job Scheduling In the Cloud with Deadlines," In Proc. 2010 Acm Workshop on Cloud Comput. Security Workshop, 2010, Pp. 93–102, Acm.

[5] I. M. Abad, "Middleware Services at Cloud Virtual Layer," In Proc. 2nd Int. Workshop on Dependable Service-Oriented and Cloud Comput. (Dsoc 2011), Aug. 2011, IEEE Comput. Soc...

[6] J. Abawajy, "Determining Service Trustworthiness in Intercloud Computing Environments," In Proc. 2009 10th Int. Symp. Pervasive Syst., Algorithms, and Networks (Ispan), Dec. 2009, Pp. 784–788.

[7] S. Frederic, F.Andreas, K. Stefan, And E. Claudia, "Improving The Scalability Of Platform Attestation," In Proc. 3rd Acm Workshop On Scalable Trusted Comput In 2008.

[8] M. A. Imad, "Toward Trustworthy Clouds' Internet Scale Critical Infrastructure," In Proc. 7th Inform. Security Practice and Experience Conf. Lncs (Ispec '11), Berlin, Germany, Jun. 2011, Vol. 6672, Pp. 73–84, Springer-Verlag.

.