Application of Lotka-Volterra model to analyse Cloud behavior and optimise resource allocation on Cloud

Chethan Babu V

Developer, SAP Labs, Bangalore, India

DOI: 10.29322/IJSRP.8.5.2018.p7712
http://dx.doi.org/10.29322/IJSRP.8.5.2018.p7712

Abstract- Cloud is a complex distributed environment which has occupied the center stage in the modern-day service computing; allowing permissive resource provisioning with minimalistic conflict and enabling on-demand, pay-per-use benefits. Provisioning of resources in a dynamic environment, such that none are under-provisioned or over-provisioned is a primal challenge. The problem is analysed, and an optimal resource allocation strategy is formulated by the quantitative analysis of a biologically inspired model called Lotka-Volterra.

Index Terms- Lotka-Volterra model, Cloud computing, Optimisation, CloudSim

I. Introduction

Traditionally, in a client-server architecture, each server in a datacenter was capable of having one operating system in it, which was very viable to be overutilized even with a limited number of user requests. This dilemma is resolved by the introduction of the disruptive Cloud architecture. Cloud computing provides IaaS, by integrating multiple operating system instances on a single host (server). The end user has the impression that he is accessing an independent server, which is, in turn, a layer of abstraction over the physical server by the virtual machines. Each of the virtual machines has a software component and the relevant virtual hardware, allocated from the physical servers. Cloud computing enables concurrent access to a shared pool of computing resources, which can be rapidly provisioned and released with minimized managerial effort. On-demand leasing of computational resources, either by pay-per-use criteria or by subscription is one of the privileges of a cloud-subscriber. Cloud computing allows companies to avoid the lump sum upfront infrastructure costs. Furthermore, the operating costs per server are comparatively lower in larger farms than small datacenters. Cloud provides virtual machines which accept the user requests and allocates the available physical resources accordingly. Cloud service provider acts as a broker between user requests and the cloud. The major challenge that confronts these cloud service providers is the provisioning of cloud resources in a dynamic environment without compromising the quality of service within limited cost margins.

Several implementations of biological concepts have shown good results in the computational platform. This paper is intended to implement an optimisation model for the cloud so that no resources are under-provisioned or over-provisioned. When the demand for cloud resources is highly volatile, the chances of the resources being under-provisioned or over-provisioned is inflated. We use a biological model called Lotka-Volterra (Predator-Prey model) to control the cloud system for stabilising the optimal provisioning of resources. Lotka-Volterra is one of the application areas of Biology which portrays the correspondence of the population interdependence of a predator and prey species. Here, we have adopted a similar kind of interdependency in the cloud ecosystem, with the predators being the user requests and the prey being the limited cloud resources. The parameters of the Lotka-Volterra model is altered to optimise the resource allocation and control the system breakdown. Lotka-Volterra model also provides us with a mathematical property known as limit cycles which are described in contour portraits also known as phase portraits of the system. Limit cycle describes a qualitative limit for the stability of a system. A system whose parameters can be differed in such a way that the system grows out of balance. The difference acquired by the configuration is measured to tell the domain of stability. Aforementioned has direct application in understanding the performance of a web server with incoming requests. Limit cycle of a system with the rate of incoming requests can help us understand the bound of the system. Dynamic acquisition or release of resources have to be implemented on the cloud system for improved scalability and streamlined performance.

As the paper is based on the predator-prey interaction dynamically, agent technology concept is implemented, so that the volatile and heterogeneous demands of the users are controlled in a cloud-like environment. In an unpredictable, dynamic environment, computer systems called agents, are proficient in extensible autonomous behavior. Single-agent systems are difficult to control and maintain, hence comes multi-agent technology. Replicating user tasks on his behalf can be performed by the programs embedded in the multi-agent systems. The entire architecture is optimized by implementing the multi-agent system, in [4] by the author. In our proposed approach, some degree of autonomy is obliged to permit the elements to counter to dynamically shifting conditions. Our paper addresses the problem mentioned above and focused on solving the allocation problem using biological

II. Related Work

Resource optimisation in the Cloud platform is one of the most significant challenges for a cloud provider. Numerous research groups have stated diverse solutions for it. On-demand cloud resource allocation plan is costlier than reservation plan.

The author architected an algorithm for optimal cloud resource provisioning in the paper [1], and he implemented stochastic programming model to over- come that problem. In the paper [2] and [3] agent technology is used to control dynamic environment like the cloud. In paper [5] the author has proposed two models for cloud resources allocation and management. Static model deals with the distribution of re- sources in advance. Dynamic model uses Predator-Prey dynamics to control the cloud environment abstractly. This paper illustrates the same with a simulation model called CloudSim.

III. OUR CONTRIBUTION

Cloud is a complex ecosystem where the allocation of any resource can happen sporadically. We use Predator-Prey dynamics to control the cloud environment. Implementation of the multi-agent technology is a nuclear part of the project. An agent is an autonomous pro- gram that performs specific tasks on behalf of a user. These agents can control the behaviour of the dynamic cloud environment. We have used CloudSim for simulating the client-server Cloud system. User requests and cloud resources were represented by Cloudlets and Virtual machines respectively.

1.1 Predator-Prey Model and Population Dynamics: The Classical Lotka Volterra

Lotka-Volterra is a biological model for the growth of two inter- dependent populations. The models deal with any two species of animals and check the interdependencies for survival. The interdependence might arise because one species serves as food for the other species. Consider an example of Rabbits (Prey) and Foxes (Predators). These are two interdependent species where Rabbits are the food source for Foxes. The population change through time according to the pair of equations

$$dP/dt = \alpha P - \beta PQ \qquad (1)$$

$$dQ/dt = \delta PQ - \gamma Q \qquad (2)$$

Where,
P is the number of Rabbits (Prey)
Q is the number of Foxes (Predators)
α is natural reproduction rate of rabbits in the absence of predation
β is the death rate per encounter of rabbits due to predation
γ is natural death rate of foxes in the absence of food
δ is reproducing rate of foxes
dP/dt and dQ/dt represent the growth rates of two populations over time

Predicting how would the model behave when the population is stable. When population is stable, $\frac{dP}{dt} = \frac{dQ}{dt} = 0$
$\alpha - \beta Q = 0, \ \delta P - \gamma = 0$ (3)

Equation (3) evaluates to a stationary point ($\gamma/\delta, \alpha/\beta$). Plotting a phase-portrait for the above gives Fig1. Notice from Fig1 that all variations of population encircle around a stationary point.

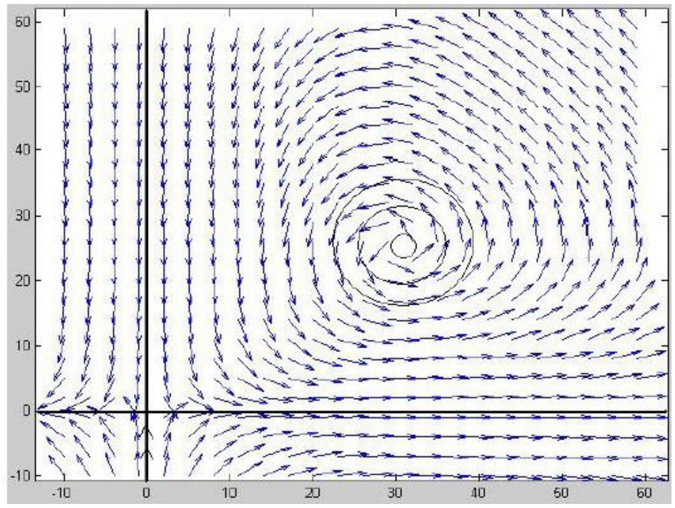

Fig. 1. Phase Portrait

1.2 The modelling approach in CloudSim

CloudSim is a framework used for simulating and modelling cloud computing foundations and architectures. Recently Cloud computing emerged as leading technology for providing reliable, secure, sustainable and scalable computing services. These extensive ecosystems of the cloud environment, coupled with the growing market for energy-efficient computational technologies, necessitate appropriate, oscillatory, and controllable techniques for processing of algorithms, frameworks, and policies prior to the actual construction of the cloud products. Since the application of the actual testbeds limit the test experiments to the order of the testbed and makes the replication of results a complicated undertaking, substitutional approaches for testing and analysis leverage the development of innovative cloud technologies. An appropriate choice is the application of simulations tools, which presents the probability of assessing the condition before software development in an environment wherein the user can reproduce test cases. Here in this paper, we use CloudSim to simulate in two different ways and compare the experimental results.

CloudSim is a java based framework, equipped with abun- dant customisation features for replicating the cloud environment. The system simulates user requests as cloudlets, and they get executed in the virtual machines, allocated on the physical hosts in the datacenters. The commencement of the lifecycle of the simulation is by the creation of the datacenters, followed by the generation of brokers, virtual machines and cloudlets. Moreover, CloudSim has the attributes to configure hardware requirements like the amount of RAM needed in a particular VM.

The initial approach is to allocate all the resources statically at the beginning of a simulation. When the resources are allocated statically, it results in over/under utilisation and/or over/under provisioning of resources. Over-provisioning of resources occurs when the user requests gets surplus resources than the demanded ones. Under-provisioning of resources happens when the user requests are addressed with less number of resources than the demand. Both over-provisioning and under-provisioning of resources do not result in optimised resource allocation.

The next approach is to add the resources on-demand dynamically. Adding resources dynamically into the system avoids over or under-provisioning of resources. Here the dynamic simulation model is compared with a biological model called Lotka-Volterra. The resources on CloudSim compared with Lotka-Volterra model as,
P is the number of Virtual Machines (Preys)
Q is the number of Cloudlets (Predators) where Cloudlet specifies the user request
α is birth rate of Virtual machines in the absence of predation by Cloudlets
β is death rate of Virtual machines due to predation
γ is natural death rate of Cloudlets in the absence of Virtual Machines
δ is reproducing rate of Cloudlets
The simulation model is used to compute the parameters of Lotka-Volterra model. These parameters are used to control the system.

IV. EXPERIMENTAL RESULTS

When the model is simulated with stable population of 1000 Cloudlets and 200 VMs, Fig.2 is obtained which shows the dynamic behavior of Cloudlets. The same model is simulated multiple times with 250, 500, 750 and 1000 Cloudlets keeping the VM count 200 and a phase plot is obtained as Fig3. It can be intuited from Fig.3 that the resources are over-utilized.

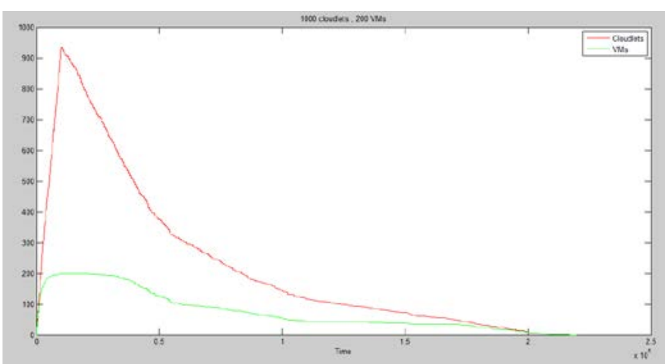

Fig. 2. Variation of Cloudlets and VMs wrt time (Static allocation)

The above graph can be differentiate among several phases and based on the above graph some cases can be analysed

Time stamp 0.0 This phase signifies the ratio of Vm vs Cloudlets as 200/ 950 i.e 0.2105. This implies every Cloudlets can consume the 21.05 percent of its allocated Vm.

Time stamp 0.5 This phase signifies the ration of vm vs cloudlets as 120/375 i.e 0.32. This implies every cloudlets can consume the 32 percent of its allocated vm.

Time stamp 1.0 This phase signifies the ratio of vm vs cloudlets as 50/150 i.e 0.333. This implies every cloudlets can consume the 33.3 percent of its allocated vm.

Time stamp 1.5 This phase signifies the ratio of vm vs cloudlets as 30/80 i.e 0.375. This implies every cloudlets can consume 37.5 percent of its allocated vm.

Time stamp 2.0 This phase signifies the ratio of vm vs cloudlets as 10/10 i.e 1. this implies a single clouldlets is consuming the complete Vm resource.

The overall experimental result implies timestamp 0.0 is the phase of system dynamics as overload. Whereas time stamp 0.5, 1.0 and1.5 are overall equilibrium state with average cloudlet allocation 3 per vm. The time stamp 2.0 is the phase of underloaded system. To maintain the equilibrium, at timestamp 0.0 the additional vm need to be released so that the ratio come around 0.3. At timestamp 2.0 phase every Vm is underloaded, so it is better to kill some vm so that the ratio reaches as 0.3 avera**ge.**

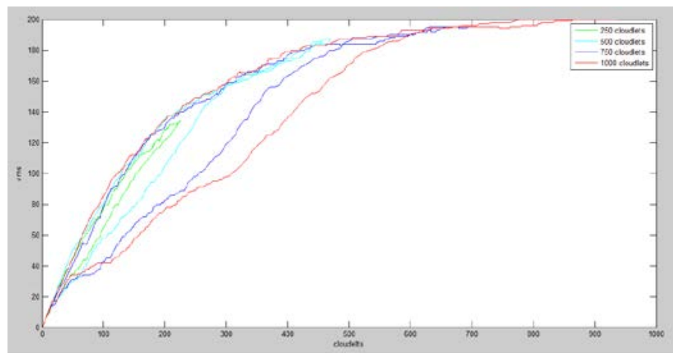

Fig. 3. Phase-Portrait obtained for Static VM and Cloudlet Generation

The above result explains the simulation scenario of cloud allocation dynamics with a finite Vm and fixed cloudlets. The static model results explains that the relation of predator(cloudlets) and prey (Vm) maintain a typical relation of nonlinear form of $y = x^{1/m}$ while m is a positive real number. This graph provides a clear idea to predict the number of Predator over a particulate value of prey with correct estimation of m. This hint us how many number of Vm should be there is the system dynamics as per the needed load factor.

Dynamic simulation of Cloudlets and VMs following the Lotka-Volterra model results in the computation of parameters which can be used to control the entire system. The values of the parameters computed from the Cloudsim for one simulation with 1000 Cloudlets and 200 VMs, where 50 VMs are generated dynamically is
$\alpha = 0.005$
$\beta = 0.0001$
$\gamma = 0.04$
$\delta = 0.001$
Initial Number of Cloudlets, Q= 30 Initial Number of VMs, P = 150

The whole system can be controlled by varying this parameters. This indicates that complex distributed dynamic environment like Cloud follows lower order non-linear dynamics.

The above graph is the result of system dynamics for dynamic model. Comparing with Fig.2 which indicates the static model, the decision can be obtained. The output signifies that both of them follow almost same nature of predator prey ratio with typical three phase of overload, balanced load and under load.

Fig.5 explain the relation of predator-prey for dynamic allocation of Vm. Again as a comparison with static scenario this can be concluded both the case is able to provide a hint

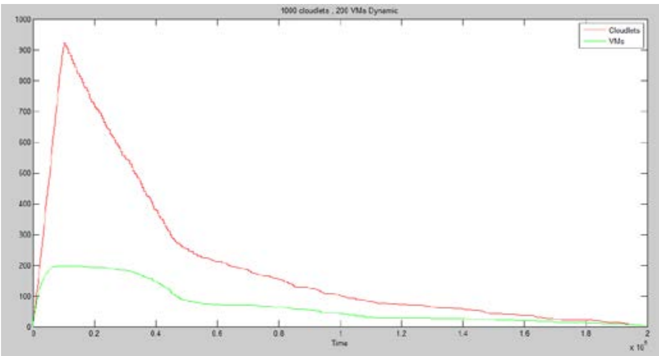

Fig. 4. Variation of Cloudlets and VMs w.r.t time (Dynamic allocation)

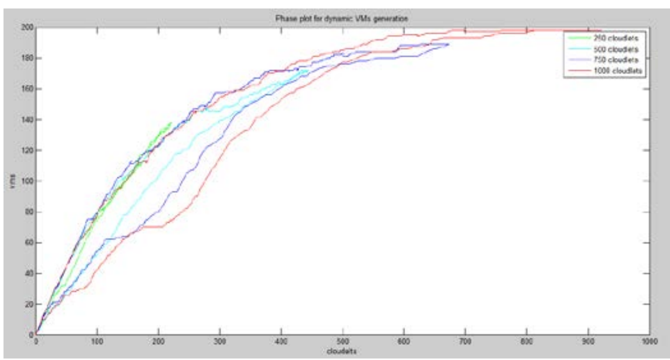

Fig. 5. Phase-Portrait obtained for Dynamic VM and Cloudlet Generation

of availability of Vm vs the existing cloutlets on a given point of time.

Fig.7 is a classical Lotka-Volterra model obtained by plotting mathematically using the parameter values computed from a simulation model. Comparing Fig.4 and Fig.7, it can be stated that complex distributed environment like Cloud follows lower order non-linear dynamics.

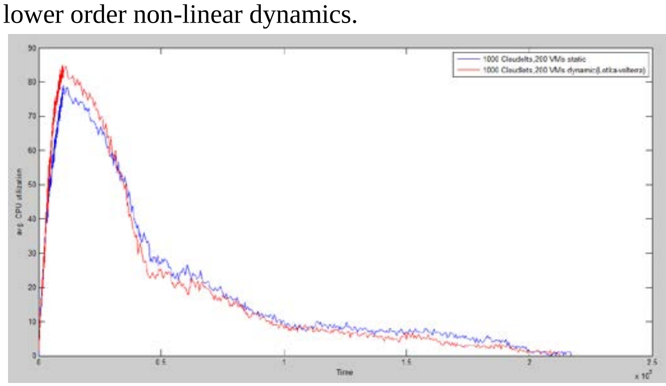

Fig. 6. Average CPU Utilization

From Fig.6 it can be intuited that; the over utilization of CPU is relatively decreased. This is because over-provisioning of VMs is reduced as VMs are added dynamically on-demand into the system.

The above picture highlights the CPU consumption by VM on static model as well as in dynamic model. The both the scenario implies the ratio of Vm to cloudlets maintain 0.33 ratio as an average apart from the scenario of under load and overload situation.

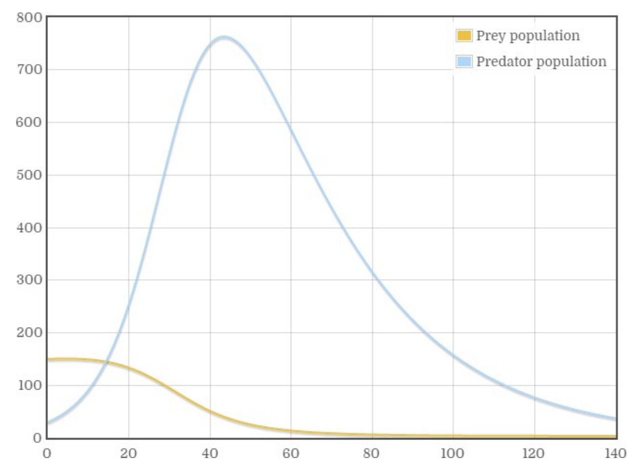

Fig. 7. Classical Lotka-Volterra plot for parameters computed

V. CONCLUSION

It can be intuited from the experimental results that, a dynamic environment like cloud follows lower dimensional chaos (Non-linear dynamics). The motive of the project was to bring about the cloud configurations under the different situation and model them using non-linear dynamics. The parameters were calculated at the boundary conditions using a java based simulation platform called CloudSim. The dynamic creation of cloudlets allowed us to vary the number of cloudlets and the resources of the cloud to create a system which was able to showcase all possible scenarios. Lotka- Volterra model suited as the best model to describe the cloud parameters with reasonable accuracy. A phase portrait is used to determine over-utilisation of resources, leading the whole system into instability. A phase portrait is repeatedly plotted by using the cloud data. When there is a sign of uncertainty in the system, more re- sources can be added to bring the system into an equilibrium state.

VI. FUTURE WORK

The idea projected is going to be tested under different traffic load on a separate cloud server. The project has been analysed under various conditions, in a simulation-based environment called CloudSim. The real intricacies of the parameters measured would show up only when the model is subjected to actual traffic on a cloud server optimisation-threshold. Agent technology can be employed to control the whole system. It keeps monitoring the entire system and can be used to control dips and rises in the number of resources. Other non-linear models such as Lorentz model would be tested for better stability. The model used has mostly been representative of the vast majority of non-linear models available. Other non-linear models might have a better accuracy at predicting the balance of the system.

REFERENCES

[1] Sivadon Chaisiri, Bu-Sung Lee and Dusit Niyato, : Optimization of Resource Provisioning Cost in Cloud Computing, IEEE TRANSACTIONS ON SERVICES COMPUTING, VOL. 5, NO. 2, APRIL-JUNE 2012

[2] M. Luck., McBurney P. and Preist. C and the AgentLink Community, Agent Technology: Enabling Next Generation Computing,

Agent Technology, a roadmap, page 94- 1: 94Agentlink

[3] M. Kang*, L. Wang and K. Taguchi: Modeling Mobile Agent Applications in UML2.0 Activity Diagrams," Third International Workshop on Software Engineering for Large-Scale Multi-Agent Systems (SELMAS'04)" W16L Workshop - 26th International Conference on Software Engineering(2004/916)

[4] Andrs Pantoja and Nicanor Quijano: A Population Dynamics Approach for the Dispatch of Distributed Generators, IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, VOL. 58,

NO. 10, Page 4559-4567 @2011

[5] Bidisha Goswami and Snehanshu Saha: Resource Allocation in Abstraction using Predator-Prey Dynamics: A Qualitative Analysis, Volume 61 No.6, January 2013

AUTHORS

First Author – Chethan Babu V, Bachelors of Engineering, PESIT South Campus, email: chethan1512@gmail.com, Phone: +91 9620422684