

Performance Analysis of Cloud Computing using Multistage Ant System

K. Mukherjee

Computer science & Engineering,
Birla Institute of Technology,
Ranchi , India

G.Sahoo

Information Technology,
Birla Institute of Technology
Ranchi, India .

ABSTRACT

Utilization of resources is difficult in a heterogeneous, dynamic environment, like Cloud computing, where resources are accessed and analyzed in real time. Sometimes, it is also needed to adapt to the changing resource usage scenario in order to maintain the desired QoS. We observe that the existing infrastructure of modern civilization is based on accessing and utilizing those resources which are pay per user in nature. As for example, the utility services like water, gas, electricity etc are chargeable per user. Similarly, the agenda of cloud computing is to provide on demand IT resources on pay per user basis. These IT resources consist of different web services. Accessing and scheduling these web services is always a challenging job. This paper proposes a multistage mathematical model based on ant system, for proper utilization of these web services. Results of the implementation are presented in order to demonstrate the effectiveness of the mathematical model.

Keywords: Cloud computing, Web services, Ant System

1. INTRODUCTION

Performance monitoring of any application in cloud computing is always complex, different and challenging. In the absence of precise knowledge about the availability of resources at any point of time and due to the dynamic resource usage scenario, prediction-based analysis is not possible. Thus, performance analysis in cloud computing must be characterized by dynamic data collection (as performance problems must be identified during run-time), data reduction (as the amount of monitoring data is large), low-cost data capturing (as overhead due to instrumentation and profiling may contribute to the application performance), and adaptability to heterogeneous environment. Just like the day to day utility services such as water, gas, electricity etc. are essential for the smooth running of our daily lives; the trend of cloud computing[21] is also to provide different IT resources on demand on a pay per user basis. The IT resources include different computational web services of different nature like tax calculation web service, weather information web service, shipping status web service etc. Cloud computing aims to be global and to provide such computational web services to the masses, ranging from the end user that hosts its personal documents on the internet to enterprises outsourcing their entire IT infrastructure to external data centers. In fact, cloud has given a new approach to make IT a real utility which is global and complete for the end users. The agenda of computational web service of cloud is to offer simple web methods that computational client can call to perform application specific computation on their own data. Web services are programmable and reusable. They are available

anywhere via the internet. Programs built using this model will run across multiple websites extracting information from each of them and combining and delivering it in a customized form to any device anywhere in the world. The potential of web services is unlimited. For example, a software company may provide a web service to calculate income tax. Any company that wants to calculate income tax can subscribe to this web service. The company offering the service can dynamically update it to accommodate new taxation rates. The subscribers need not to do anything to get the new updates. In future a collection of such web services may replace packaged software. As web services break down the distinction between the internet, standalone applications and computing devices of every kind, they enable business to collaborate and offer an unprecedented range of integrated and customized solutions that enable their customers to act on information at any time, any place and on any device. There are different key players to develop web services, as for example, Microsoft has come up with .NET platform. This platform simplifies the process of building these solutions and provides a framework for integration and interoperability. This platform is based on open standards like XML and SOAP so that it can work across all programming languages and operating systems. It helps us to combine the power of PCs and smart devices with the richness of internet. Again SUN has introduced java to develop web services etc. Web services consists of three layers as given below:

Layer1: A Framework to build web services.

Layer2: Development tools and language to make this development easy.

Layer 3: Server infrastructure to deploy and operate the web services.

But these web services are randomly scattered over different web servers and accessing and scheduling them is really a challenging job. For this, we have used multistage Ant System. Ant System is a population based metaheuristic which is capable to find approximate solution of difficult optimization problem. Ant System [24] is best suited in solving problems which run in dynamic and varied environments without the help of any central control. This is why it is very much applicable to distributed problem solving. As cloud computing is dynamic in nature, hence we propose to apply Ant System in cloud computing.

Organization of this paper is as follows: Related work is discussed in Section 2. Proposed Frame Work of Cloud Computing is discussed in Section 3. Proposed multistage Ant System model is discussed in section 4. Section 5 gives the details of the simulation results.

2. RELATED WORK

Most of the existing work in web service performance focuses on latest trend of technologies and standards. Andreozzi et al. [2] present a model for rigorous representation of service characteristics. Gouscos et. al. [3] presents a simple approach to model certain web service management attributes. Thomas et al. [4] represent distributed web service by modeling the flow of messages and methods in a web service transaction. Tue et al. [5] discuss design strategies to improve the performance of web service. Levy et al. [6] present architecture and prototype implementation of performance management system for cluster based web service. Cardellini et al. [19] consider different category of web applications, and evaluates how static, dynamic and secure web service request affect performance and quality of service of distributed web sites. But the entire above discussed web services are either very much server centric or device centric in nature. A properly distributed web service that fits effectively in cloud environment is rarely available. Our web service is computational in nature that hides parallel computing resources behind easy to use web service. We have proposed a mathematical model for better utilization of these web services. We have used multistage Ant System for better performance of cloud computing. Ant System was proposed by Dorigo et al.[25]at the year 1992. In Ant system, Ant build tour based on probabilistic action choice rule, called random proportional rule to decide for the tour of different cities in Traveling salesman problem. In this paper, we have used this nature of ant in cloud computing to solve the problems which are dynamic without the help of any central control.

3. PROPOSED FRAMEWORK OF CLOUD COMPUTING

AS we know that cloud computing is computing over a cloud[23]; where a cloud consists of grids of commodity machines and a software layer (called Hadoop), which is responsible for distributing applications data across the machines, parallelizing and managing application execution across the machines, and detecting and recovering from machine failures. We propose that the Hadoop would consists of four components. Each of components has a specific job. The different component of Hadoop are shown in Figure 1.

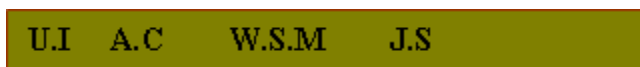


Figure 1. Proposed Components of Hadoop

Where

U.I is User Interface.

A.C is Authentication Check.

W.S.M is Computational Web Service Mapping.

J.S is Job Scheduler.

Whenever the user requests for a web service to Hadoop, the latter checks the authenticity of the user after interfacing with Authentic Server, then Hadoop refers “Web Service Mapping” and maps to web servervices existing at different location and fetches the required webservice from it and submits it to “Job scheduler” of Hadoop which schedules the jobs to the Grid of volunteer commodity hardware. The idea behind volunteer

computing [20] is to allow ordinary users on the internet to volunteer their idle computers’ processing powers (PCs, Clusters, Supercomputers, Mainframes etc., which as a whole make volunteer commodity hardware) towards solving computationally intensive tasks. The scheduler of Hadoop sends the jobs to idle volunteer commodity hardware, the sent jobs are loaded to these idle volunteer commodity in the form of an assembly (i.e a DLL file, in context of .NET platform) or byte code(in context of Java platform); security mechanisms in Microsoft’s .NET framework or JVM of java safely executes these assemblies or byte codes and when jobs get computed successfully they are pulled back by Hadoop and are sent to the required thin clients, Mobile etc. The load balancing of idle volunteer commodity hardware is being done by J.S of Hadoop. Thus Hadoop provides super computing power in the hands of ordinary users. A country like India where 70% of the total population earns less than \$2 per day, can not afford expensive laptops or desktops etc. So, with cloud computing, software applications can be accessed over a network using thin clients or mobiles(which are a lot cheaper).Thus cloud computing can help to make computing ubiquitous and bring it within the reach of the masses, especially the poor. In this paper, we propose a specific frame work of clod computing (shown in figure 2), at which Hadoop is at the top, which is being accessed by thin clients or by commodity hardware(PCs, Clusters, Supercomputers, Mainframes etc).The Commodity hardware again are of two types:

a)Active Commodity hardware: It needs web service from Hadoop.

b) Idle Commodity Hardware: The idle Commodity hardware are used as volunteer computing commodity hardware. They are used for processing the web services and the processed web service are supplied to thin clients, active commodity hardware etc.

Thus Hadoop is connected with thin clients and commodity hardware. At a time all commodity hardware can not be busy, some may be idle and we have to use them in optimized order in order to enhance the cost/benefit analysis of cloud computing. By providing a simple interface to the user, Hadoop makes it possible to achieve super computing power as easily as one can get electric power through a wall socket.

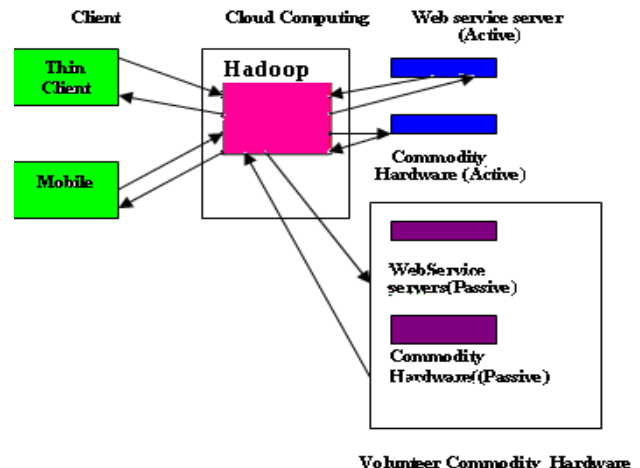


Figure 2

4. PROPOSED MULTISTAGE ANT-SYSTEM.

In ant system, artificial ants are put on randomly chosen nodes. Here nodes represent different stages and each stage represents nectar of different quality. In context of cloud computing, “quality of nectar” implies the different web services. In contrast with the traditional ant system, our proposed ant system in addition to the other parameters of ant system, has a special parameter $r_{ji}(d)$ (i.e. immediate return from stage j to stage i due to decision d). In cloud computing, users need different types of web services (i.e. different quality of nectar in context of Ant system). In general, we can say that k^{th} ant will immediately return back after getting a specific quality of nectar (A specific web service) from j^{th} stage to i^{th} stage; whereas $(k+1)^{th}$ ant will return from h^{th} stage to m^{th} stage and so on. In order to move from one stage to other, ant uses random proportional rule.

Let p_{ij}^k = Probability with which ant k , currently at stage i

$$\begin{aligned} & \text{chooses to go to stage } j \\ & p_{ij}^k = \frac{[\tau_{ij}]^\alpha [\eta_{ij}]^\beta r_{ji}(d)}{\sum_{l \in N_i^k} [\tau_{il}]^\alpha [\eta_{il}]^\beta}, \text{ if } j \in N_i^k \\ & = 0, \text{ otherwise.} \end{aligned}$$

Where

η_{ij} = Heuristic value = $1/D_{ij}$

D_{ij} = Euclidian distance.

τ_{ij} = Pheromone Trials.

α = Parameter which determines the relative influence of the pheromone trial.

β = Parameter which determines the relative influence of the pheromone trial

N_i^k = Feasible neighborhood of ant k when it is at web server i .

$r_{ji}(d)$ = Immediate return of ant from j^{th} stage to i^{th} stage due to decision d .

We define $r_{ji}(d)$ as follows:

$$\begin{aligned} r_{ji}(d) &= +1, \text{ when } d=1 \\ &= -1, \text{ when } d=0; \end{aligned}$$

At +1 case, the ant moves forward from its current position. Here we apply forward recursive approach.

At -1 case, the ant moves backward from its current position. Here we apply backward recursive approach.

The proposed algorithm is given below:

Algorithm 1

Forward_Recursive

/* Initialization*/

Step 1. For every stage (i,j) do

$$\tau_{ij}(0) = \tau_0$$

$$\eta_{ij} = \eta_0$$

$$d = 1$$

End for

Step 2. For $k=1$ to m do

Place ant k on a randomly chosen web server.

End for.

/* Main Loop */

Step 3. For $t = 1$ to t_{max} do

For $k = 1$ to m do

Build tour T^k by using random proportional rule to move to next web server j

$$\begin{aligned} p_{ij}^k &= \frac{[\tau_{ij}]^\alpha [\eta_{ij}]^\beta r_{ji}(d)}{\sum_{l \in N_i^k} [\tau_{il}]^\alpha [\eta_{il}]^\beta}, \text{ if } j \in N_i^k \\ &= 0, \text{ otherwise.} \end{aligned}$$

Where i is the current web server.

If $d = 1$, $r_{ji}(d) = 1$

Store stage j to $tabu_k$.

End For.

For $k = 1$ to m do

Compute the length L_k of every ant and update the shortest tour found.

For every stage (i, j) do

Update pheromone trails by using following:

$$\tau_{ij} = (1 - \rho) \cdot \tau_{ij}(t) + \sum_{k=1}^m \Delta \tau_{ij}^k$$

Where $\Delta \tau_{ij}^k = 1/C^k$, if stage (i, j) $\in T^k$

$$= 0, \text{ otherwise}$$

C^k is the sum of lengths of arc belonging to T^k

ρ is the pheromone evaporation rate.

If the required web service is found at stage j then

do

set $d = 0$

```

call Procedure Backward Recursive( tabuk, d )
Else Continue
End For
End

```

Algorithm 2

Backward Recursive (tabu_k, d)

```

Step 1. Repeat until d == 0
do
For every stage (i, j) do
If i is in tabuk then move from current stage j to
stage i
if the current position is the starting stage then do
set d=1
Step 2. End.

```

5. EXPERIMENTAL SET UP

A local Cloud test bed has been set up that consists of heterogeneous nodes running in Linux. The computational nodes of the test bed include HP Net Server LH 6000, HP ProLiant ML570 G4 with 4 Intel core2 Duo processors, Intel core2 Duo PCs and Intel Pentium-4 PCs. The nodes communicate through a fast local area network. The Cloud test bed is built with Globus Toolkit 4.0 (GT4) [12]. This paper demonstrates the results of local accessing of multiple web services running on the web Servers. Exclusively one agent is responsible for analyzing the performance of multiple jobs submitted to the same node and running simultaneously. Experiments have been carried out on a HP ProLiant ML570 G4 with 4 Intel core2 Duo processors (referred as HPserver) in the Grid environment. As test codes, we have used Java codes with JOMP(JOMP implements an OpenMP-like set of directives and library routines for shared memory parallel programming in Java) directives and executed in Jade framework. It has already been mentioned that the current work presents the results related to analysis and tuning of Insufficient Parallelism problem observed in the test codes. When identified by the agent, tuning agent decides that additional processors are required to overcome this performance problem. For example, if the job initially executes on p (≥ 1) processors, performance enhancement may be achieved by providing more processors ($q > p$) at run-time. Tuning Agent decides whether the job will continue either with p processors or with additional processors q ($> p$) and resumes job after allocating additional resources to the job. The framework supports multiple jobs running on the same node. Thus, multiple parallel Java codes (with JOMP directives) start executing using single processor on the HPserver and an agent monitors performances of all of them by profiling and analyzing the performance data. We use simple timer routines for profiling. After completing a part of the computation, the agent analyses the data, identifies that performance needs to be

improved and recognizes that performance can be improved by adding more processors to the execution environment.

6. RESULTS

We have experimented with two different web services. However, in order to demonstrate the effectiveness of our system, we submitted multiple images of the same webservices as separate jobs. The jobs have been initiated at different times. The comparison between forward recursive ant system and the backward ant system is shown below:

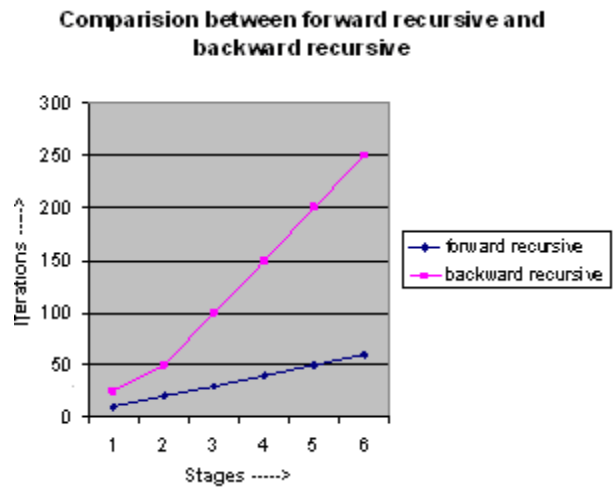


Figure 3

From figure3, we conclude that due to tabu_k backward recursive is more effective than forward recursive. In forward recursive we are using probabilistic approach where as in backward recursive we are using deterministic approach. Thus our proposed algorithms provide a better utilization of web services in cloud computing environment. Here the ants proceed for a specific web service and search at different web servers and once they get it, return immediately due to decision d ; for this old ants die and new ants are born, which use backward recursive approach from that very nectar (i.e. web service server) to nest(i.e Hadoop) using tabu_k. Due to tabu_k, backward recursive is more effective, as figure3 reflects.

7. COCLUSION AND FUTURE WORK

In this paper, we have proposed a new framework of cloud computing. Here we have focused only on accessing different web services. Scheduling of these web services to different volunteer nodes has not been discussed, which we intend to do in our forthcoming endeavor.

8. REFERENCES

- [1] DeRose, L., Y. Zhang, D.A. Reed, "SvPablo: A Multi-Language Performance Analysis system", in Proceedings of 10th Internal Conference on Computer performance, pp. 352-355, Spain, 1998
- [2] S.Adreozzi, P. Ciancarini, D. Montesi, R. Moretti, "Towards a model for quality of web and grid service" In Proc 13th IEEE international Workshops on Enabling

- Technologies: Infrastructure for Collaborative Enterprises(WET ICE'04) page 271-276,2004
- [3] D. Gouscos, M. Kalikakis, and P. Georgiadis. "An approach to modeling Web service QoS and provision price", in Proceeding 3rd International Conference on Web Information Systems Engineering Workshops, pages 121-130, 2003
- [4] J.P Thomas, M.Thomas and G.Ghinea "Modeling of Web service flow", in Proceeding IEEE International Conference on E-Commerce (CEC 03), pages 391-398, 2003
- [5] S.Tu, M.Flanagin, Y. Wu, M. Abdelguerfi, E. Normand, V. Mahadevan, "Design Strategies to improve performance of GISWeb services", in Proceeding International Conference on Information Technology : Coding and Computing(ITCC04),pages 444-448,2004
- [6] R.Levy, J Nagarajaro, G. Pacifici, M. Spreitzer, A. Tantawi, and A. Youssef, "Performance management for cluster based Web services", in Proceeding IFIP /IEEE 8th International Symposium on Integrated Network Management, pages 247-261, 2003
- [7] Fahringer, T., Gerndt, M., Riley, G. D., and Traiff, J. L., "Formalizing OpenMP Performance Properties with ASL", In Proceedings of the Third International Symposium on High Performance Computing (October 16-18, 2000), ISHPC, pp. 428-439
- [8] Foster I., C. Kesselman, J. Nick, and S. Tuecke, "Grid Services for Distributed System Integration", IEEE Computer, pp. 37-46, June 2002
- [9] Furlinger K., "Scalable Automated Online Performance Analysis of Applications using Performance Properties", Ph.D. Thesis, 2006 in Technical University of Munich, Germany
- [10] Furlinger K., M. Gerndt, "A Lightweight Dynamic Application Monitor for SMP Clusters", HLRB and KONWIHR joint Reviewing Workshop 2004, March 2004.
- [11] Furmento N., A. Mayer, S. McGough, S. Newhouse, T. field, and J. Darlington, "ICENI: Optimization of Component Applications within a Grid environment", Parallel Computing, 28(12): 1753-1772, 2002
- [12] Globus Toolkit 4.0 – available in www.globus.org/toolkit
- [13] GrADS: Grid Application Development Software Project, <http://www.hipersoft.rice.edu/grads/>.
- [14] Kennedy K., et al, "Toward a Framework for Preparing and Executing Adaptive Grid Programs", Proceedings of the International Parallel and Distributed Processing Symposium Workshop (IPDPS NGS), IEEE Computer Society Press, April 2002.
- [15] Kesler J. Charles, "Overview of Grid Computing", MCNC, April 2003.
- [16] Miller, B.P., M. D. Callaghan, J.M. Cargille, J.K. Hollingsworth, R.B. Irvin, K. L. Karavanic, K. Kunchithapadam, T. Newhall, "The Paradyn Parallel Performance Measurement tools", IEEE Computer 28 (11), pp. 37-46, 1995
- [17] Nudd, g.R., Papaefstathiou, E., et. Al., "A layered Approach to the Characterization of Parallel Systems for Performance Prediction", in Proceedings of Performance Evaluation of Parallel Systems, pp. 26-34, Warwick, 1993
- [18] Ribler R.L, H. Sinitchi, D. A. Reed, "The Autopilot Performance-Directed Adaptive Control system", Future Generation Computer Systems 18(1), pp. 175-187,
- [19] V. Cardellini, E. Casalicchio, and M. Colajanni," A performance study of distributed architectures for the quality of Web services", in Proceeding 34th Annual Hawaii International Conference on System Sciences,2001. September 2001
- [20] L.F.G Sarmenta, "Volunteer Computing", Ph.D. Thesis in pages 137-150. "Massachusetts Institute of Technology" March 2001.
- [21] L.Vaquero, L.Rodero-Marino,J.Caceres, M.Linder, "A break in the clouds: towards a cloud definition" SIGCOMM Computer Communication Review,2009.
- [22] T.D. Seely, "The Wisdom of the Hive", Harvard University Press, 1995, pages 295-296.
- [23] M.Armbrust, A.Fox, R.Griffith, A.D. Joseph, R. Katz, A.Konwinski, G.Lee, D. Patterson, A. Rabkin, I. Stoika, M. Zaharia, "Above the Clouds: a Berkeley view of cloud computing", Technical report, available at <http://abovetheclouds.cs.berkeley.edu>
- [24] Eric Bonabeau, Marco Dorigo, and Guy Theraulaz, Swarm Intelligence: From Natural to Artificial Systems, Oxford University Press, 1999
- [25] Marco Dorigo and Thomas Stutzle "Ant Colony Optimization", Prentice –Hall of India, 2001