Front and Back Mechanism based Task Scheduling in Cloud

Naziya Kouser
MTech 4th sem, Department of CSE,
Don Bosco Institute of Technology
Bangalore, India

Mr. Bhaskar R,
Assistant Professor, Department of CSE
Don Bosco Institute of Technology
Bangalore, India

Abstract: **The number of real time applications such as signal processing and weather forecasting running has increased in the cloud. Scheduling for real time task plays an important role to maintain the QoS. The collaborative process consist of two phase i.e. Basic matching phase, Bidirectional announcement bidding phase. Bidirectional announcement bidding phase is further classified into forward announcement bidding phase and backward announcement bidding phase. Bidirectional announcement bidding mechanism is used to enhance the system's performance. These work proposes the calculation of bidding values in both forward and backward announcement bidding phases. On the basis of these bidirectional announcement bidding mechanism an agent based dynamic scheduling algorithm is proposed for real time, independent and aperiodic tasks in clouds.**

Keywords- Agent based scheduling, real-time, bidirectional announcement-bidding mechanism.

I. INTRODUCTION

Cloud computing is an efficient paradigm to offer computational capabilities as services on a "pay-per-use" basis.
Meantime, the virtualization technology is commonly employed in clouds, e.g., the Amazon's elastic compute cloud (EC2), to provide flexible and scalable services, which gives users the illusion of infinite resources. Running applications on virtual machines (VMs) has become an effective solution. Multiple virtual machines can be run by a single host simultaneously. The VMs can be dynamically relocated by live VM operations (e.g., creation, migration and deletion) to achieve fine-grained optimization of computing resources.
From distributed artificial intelligence (DAI) domain the agent based technology is derived. It is required to provide real time guarantees for some applications. An important scheduling technology, i.e., agent-based scheduling technology that shows great advantages in dealing with task allocation issue in distributed systems. These Agent based scheduling technology is not sufficiently considered on the emerging clouds. For example, weather forecasting and medical simulations have strict deadlines which, once broken, make the result useless. Therefore, to obtain the guaranteed computing services within the time constraint is critical for the deadline constrained applications. It has great strength in open, complex, dynamic, and distributed environment due to the inherent nature that agents make decisions based on local interactions, and good agent based scheduling algorithms allow them to adapt, and enable them to coordinate through self-organization.

According to Wooldridge and Jennings, an agent is "a self-contained program capable of controlling its own decision making and acting, based on its perception of its environment, in pursuit of one or more objectives" [2], which provides a new way to allocate tasks in clouds. Coordination, negotiation among all the agents is important to complete the goal of the system. In the scheduling process based on agent technique, multiple individual agents are capable of composing a multi-agent system, in which these agents interact with each other to accomplish the goal of the system. Meanwhile, interaction technology among agents is of great importance, and the corresponding rules in the interactions can be designed by users, which makes agent-based technology flexible enough to meet totally different requirements while scheduling. To facilitate interactions, the ability to cooperate, coordinate, and negotiate with each other is required. Agent-based scheduling is to employ these operations so as to finish task allocation on cloud resources.

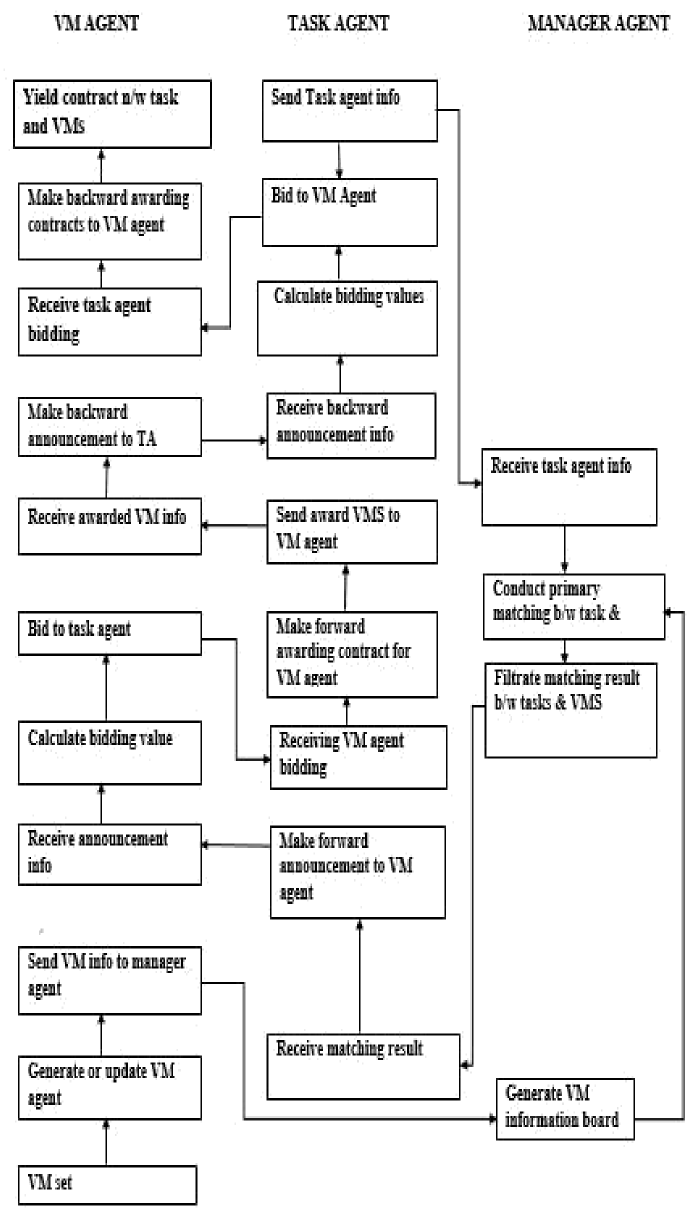

Fig 1: System Architecture

II. RELATED WORK

Scheduling algorithms can be either static (i.e., off-line) or dynamic (i.e., on-line). Assignment of tasks and the time at which the tasks start to execute are determined by a priori in static scheduling algorithm. For periodic tasks they are developed. Tasks will be scheduled and the arrival time of aperiodic tasks is not known in dynamic scheduling strategies. Kong et al. concentrated on the uncertainties of both the availability of virtualized servers and workloads, and utilized the type-I and type-II fuzzy logic systems to predict the resource availability and workloads to enhance the system's availability and responsiveness performance [3].

Malawski et al. presented several static and dynamic scheduling algorithms to enhance the guarantee ratio of real-time tasks while meeting QoS constraints such as budget and deadline [4]. The agent-based scheduling algorithms can be classified into two categories, i.e., threshold-based algorithms and market-based algorithms. The scheduling algorithms are developed from the threshold model in threshold based algorithm. For example, Price evaluated the adaptive nature inspired task allocation against decentralized multi-agent strategies. However, the large-scale dynamic scheduling issue cannot be efficiently address in market based algorithm. Both tasks and resources will be dynamically varies in cloud. Resources changes with the variation of system workload and most of tasks arrive in an aperiodic mode. Thus to enhance the system's schedulability and utilization in dynamic cloud environment the scheduling algorithms that are used to allocate tasks and adjust resources are very essential. Agent based scheduling algorithms have the ability to allocate tasks through negotiation, which brings great advantages for dealing with dynamically arrived tasks in distributed systems (e.g., the cloud computing systems).

III. PROBLEM FORMULATION

A. *Scheduling Objective*

The main objective is it lacks in schedulability. Scheduling algorithm tries to finish task before the deadline occurs. In the system overloaded condition these scheduling algorithm tries to finish the task first that is with the higher priority. Two main scheduling objectives are used: Task guarantee ratio and priority guarantee ratio.

(1) Task Guarantee Ratio:

$$TGR = \max\left\{\sum_{k=1}^{|H_a|}\sum_{j=1}^{|V_k|}\sum_{i=1}^{|T|} x_{ijk} / |T|\right\} \qquad (1)$$

Task Guarantee Ratio can be calculated with the maximum active number of host that are present, the number of VMs that are available and with the number of Task.

(2) Priority Guarantee Ratio:

$$PGR = \max\left\{\sum_{k=1}^{|H_a|}\sum_{j=1}^{|V_k|}\sum_{i=1}^{|T|} x_{ijk} \cdot p_i / \sum_{i=1}^{|T|} p_i\right\} \text{ ------------- } (2)$$

Priority Guarantee Ratio can be

calculated with max number of active host, the number of VMs and Task and finally selected that has the Propability of doing task.

Where Ha is the active number of host.

V k is the number of VM.

T is the number of Task.

P represents the Propability.

Notation	Definition
t_i	The ith task in the task set $T = \{t_1, t_2, \ldots\}$
a_i, l_i, d_i, p_i	t_i's arrival time, length/size, deadline, and priority
h_k	The kth host in the host set $H = \{h_1, h_2, \ldots\}$
H_a	Active host set, $H_a \subseteq H$
v_{jk}	The jth VM on host h_k
r_{jk}, c_{jk}	v_{jk}'s ready time and creating time
$s_{ijk}, e_{ijk}, f_{ijk}$	The start time, execution time, and finish time of t_i on v_{jk}
x_{ijk}	x_{ijk} is "1" if t_i is assigned to v_{jk}; otherwise, x_{ijk} is "0"
t_i^A	The ith task agent in the task agent set $T^A = \{t_i^A, i = 1, 2, \ldots, \lvert T \rvert\}$
v_{jk}^A	The jth VM agent in the VM agent set $V^A = \{v_{jk}^A, j = 1, 2, \ldots, \lvert V_k \rvert, k = 1, 2, \ldots, \lvert H_a \rvert\}$
m^A	The manager agent
fb_{ijk}, bb_{ijk}	The bidding value in forward bidding and backward bidding, respectively
f_{pjk}	The finish time of t_i's preceding task t_p on the same VM v_{jk}
an_i	The ith announcer in the announcer set $AN = \{an_i, i = 1, 2, \ldots, n\}$
bi_{ij}	The jth bidder for an_i in the bidder set $BI_i = \{bi_{ij}, j = 1, 2, \ldots, m\}$
bv_{ij}	The bidding value of bi_{ij} for an_i

Table 1: Definitions of main notations

IV. PROPOSED TECHNIQUE

For running real time applications in clouds it is required to obtain high performance. Scheduling plays an essential role for running real time applications, in which real time tasks in these applications are mapped to machines such that deadlines and response time (RT) requirements are satisfied. Efficient scheduling algorithms are able to facilitate the resources to contribute the whole system, and thus can significantly boost the system's service capability. To date, a handful of scheduling strategies on clouds have been investigated. Unfortunately, an important scheduling technology, i.e., agent-based scheduling technology that shows great advantages in dealing with task allocation issue in distributed systems is not sufficiently considered on the emerging clouds. The agent-based technology is derived from distributed artificial intelligence (DAI) domain. These agent based scheduling mechanism is used for following purpose:

- To improve contract net protocol (CNP), a bidirectional announcement bidding mechanism is proposed.
- An agent based scheduling algorithm is developed for independent, real-time tasks.
- MAX strategy and P strategy are used to determine the contractors.
- It is used to improve the system performance.

V. PERFORMANCE EVALUTATION

The performance metrics by which we evaluate the system performance include: Task guarantee ratio defined as:

- TGR = Total count of tasks guaranteed to meet their deadlines/Total count of tasks;

Using parameter base Deadline to control, a task's deadline that can be calculated as Di= ai + base Deadline; Where parameter ai denote the arrival time of task, and base Deadline is in uniform distribution U (base Time, a*base Time) and set a = 8;

- Priority Guarantee Ratio:

PGR = Sum of priorities of tasks that are finished before their deadlines / Sum of priorities of all tasks.

B. *Simulation Steps*

The CloudSim toolkit is chosen as a simulation platform.

The detailed setting and parameters are given as follows:

- Each host can have one CPU core and the CPU performance is based on MIPS.
- Each VM requires one CPU core that has MIPS, RAM and storage capacity.
- The start-up time of a host and the creation time of a VM is considered.
- The required number of VMs are created and the number of task are assigned.
- Primary matching between the tasks and VMs is done.
- Forward announcement contract and backward announcement contracts are determined.
- Then bidding values in both Forward announcement mechanism and backward announcement mechanism is done.
- The priority of each task is a randomly assigned.
- The high priority tasks get completed first than the low priority task.

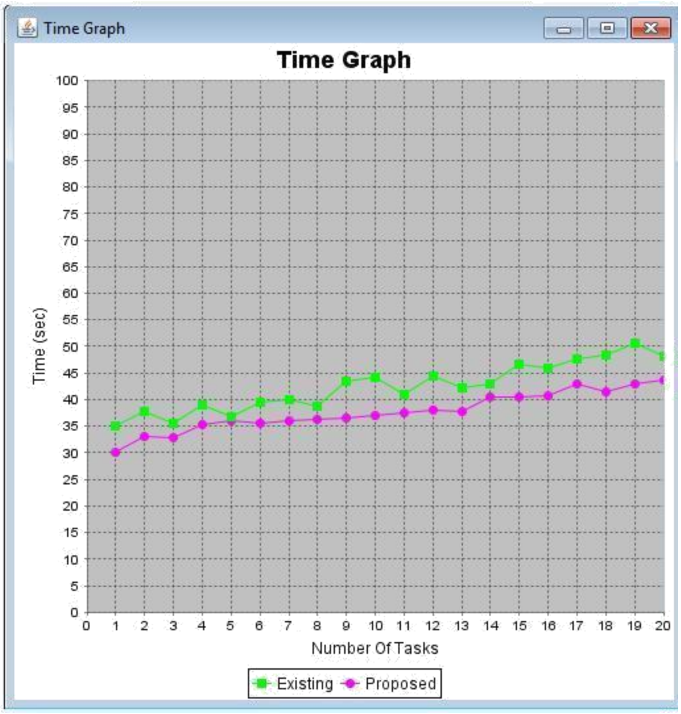

Fig 2: Time graph

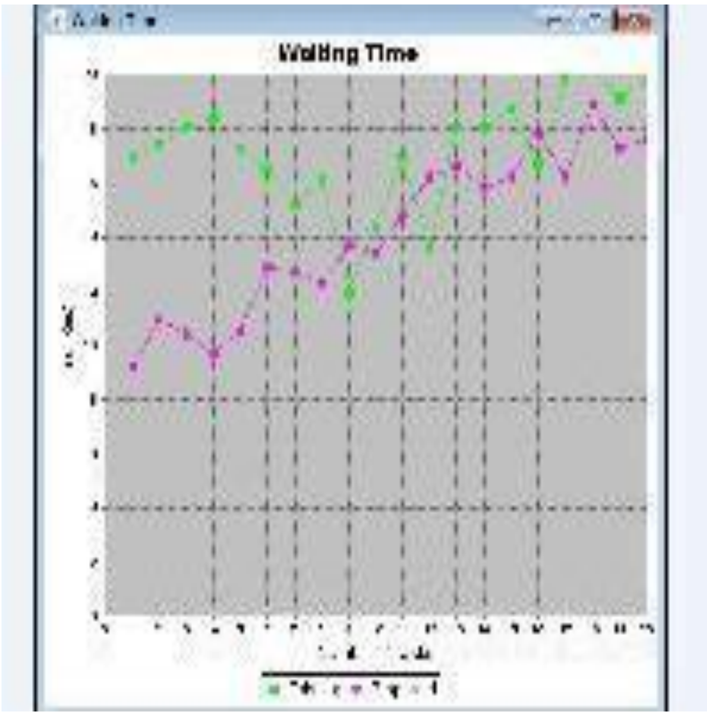

Fig 3: Waiting time graph

VI. CONCLUSION AND FUTURE WORK

In this method, the problem of agent based scheduling is used for aperiodic, independent real-time tasks in virtualized cloud. The agent based scheduling mechanism employs a new bidirectional announcement-bidding mechanism, in which the contributions include designing the basic matching policy, forward announcement bidding and backward announcement bidding, as well as their process flows. Besides, the Calculation of bidding values in forward biding and backward bidding is done. MAX strategy and P strategy are used to determine the contractors. Again, sufficiently considered the elasticity of clouds and proposed a scaling up policy to dynamically add VMs so as to enhance the system schedulability.

FUTURE ENHANCEMENT

In future the following three issues are being addressed: Firstly, the communication and dispatching times are taken into account by implementing a new scheduling mechanism. Secondly to improve the resource utilization by scaling down policy. Finally, to run the Agent based scheduling mechanism.

REFERENCES

[1] Xiaomin Zhu, Chao Chen, Laurence T. Yang and Yang Xiang, “ANGEL: Agent-Based Scheduling for Real-Time task in virtualized cloud.
[2] M. Wooldrige and N. R. Jannings, “Intelligent agents: Theory and practice,” Knowl. Eng. Rev., vol. 10, pp. 115–152, 1995.
[3] X. Kong, C. Lin, Y. Jiang, W. Yan, and X. Chu, “Efficient dynamic task scheduling in virtualized data centers with fuzzy prediction,” J. Netw. Comput. Appl., vol. 34, no. 4, pp. 1068–1077, 2011
[4] M. Malawski, G. Juve, E. Deelman, and J. Nabrzyski, “Cost-and deadline-constrained provisioning for scientific Workflow ensembles in IaaS clouds,” in Proc. Int. Conf. High Perform. Comput, Netwrage Anal., 2012, pp. 1–11.
[5] R. Price, “Evaluation of adaptive nature inspired task allocation against alternate decentralised multiagent strategies,” BSc dissertation, University of Birmingham, Birmingham, U.K., 2004.
[6] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, and I. Bran dic, “Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering Tasks in Virtualized Clouds” IEEE TRANSACTIONS ON COMPUTERS, VOL. 64, NO. 12, DECEMBER 2015 computing as the 5th utility Future Generation Comput. Syst., vol. 57, no. 3, pp. 599– 616, 2009.
[7] M. Mishra, A. Das, P. Kulkarni, and Sahoo, “Dynamic resource management using virtual machine migrations,” IEEE Commun. Mag., vol.50, no. 9, pp. 34–40, Sep. 2010.
[8] M. Owliya, M. Saadat, R. Anane, and M. Goharian, “A new agents-based model for dynamic job allocation in manufacturing shopfloors,” IEEE Syst. J., vol. 6, no. 2, pp. 353–361, Jun. 2012.
[9] G. Lange and F. Lin, “Modeling well scheduling as a virtual enterprise with intelligent agents,” in Proc. IEEE 17th Int. Conf. Comput. Sci. Eng., 2014, pp. 89–96.