A Comparative Study of Simulation Tools for Distributed Environment

Ms. Rupal D Bhatt, Dr. D.B.Choksi

Abstract— **Clusters, grids, and peer-to-peer networks have emerged as popular paradigms for next generation parallel and distributed computing systems. They enable usage of distributed resources for solving large-scale problems in science, engineering, commerce and other domains. The common approach adopted by these systems is to employ scheduling heuristics that lead to an optimal schedule.**

Scheduling diverse applications and resources in distributed environment is challenging because target resources may be heterogeneous and their load and availability varies dynamically. Moreover, independent tasks may share common data files. Unfortunately, it is often impossible to obtain analytical results to compare the efficacy of these heuristics. Given the nature of most scheduling problems one must resort to simulation to effectively evaluate and compare their efficacy over a wide range of scenarios.

This work describes the Grid and cluster scheduling simulators for study, testing and evaluation of various job scheduling techniques. We present a comparative study to highlight the usefulness of various simulators like GridSim, ClusterSim, SimGrid, Alea 2, BeoSim and Bricks for conducting scheduling research. An attempt is made in this paper to provide the basis for appropriate selection and possibility of designing new scheduling algorithms offering better performance and throughput using existing Grid and cluster scheduling simulators aimed at improvement in the performance of distributed systems.

***Index Terms*— cluster scheduling, cluster computing, distributed computing, event-based simulators.**

I. INTRODUCTION

Clustering is used to support many classes of resource-intensive applications, from database servers to numerical analysis programs. Clustered computing in its simplest form consists of a number of workstations linked via control software and a high-speed Local Area Network [1]. Evaluation of scheduling algorithms for distributed environments is a complex task that involves simulation of computing resources, interconnection network, users and workload [2].

Manuscript received Oct 15, 2014

Rupal D. Bhatt, Computer Science Department, Anand Mercantile College of Science, Management and Computer Technology, Anand , Gujarat, India

Dr. D.B.Choksi, Department Of Computer Science, Sardar Patel University, V.V.Nagar, Anand, Gujarat, India

Due to many reasons, such as the cost of resources, the reliability, the varying background load or the dynamic behavior of components, experimental evaluation cannot be mostly performed in the real systems. For this reason many simulators have been developed. Simulation is a very powerful aid for the research of scheduli ng policies and performance evaluation. There are several general Grid and cluster simulators [3] allowing to simulate various scenarios and problems.

Cluster simulations are usually supported by more complex grid simulators. Most of them like SimGrid [5], GridSim[6], Alea 3, Bricks[7], ClusterSim and BeoSim are focused on data replication and process migration among clusters. However, detailed support for scheduling policies within the cluster is not provided[4].

In this paper, we present various simulators like GridSim, ClusterSim, SimGrid, Alea 2, BeoSim and Bricks designed to perform study of scheduling algorithms for distributed application.

Section-I explains the role of scheduling simulator in distributed computing. Section II describes related work in the area of cluster scheduling simulators. In Section III we present salient features and architectures of the existing cluster and grid simulation tools. Comparison of different scheduling simulators and their results are shown in Section IV. Concluding remarks and guidelines for the future work are included in Section-V.

II. RELATED WORK

In the last few years, simulation has been used as a powerful technique for performance analysis of Computer systems[8][9][10].

Casanova, H. in his paper on “Simgrid: a Toolkit for the Simulation of Application Scheduling”, describes how schedulers are managed by iSPD (iconic Simulator for Parallel and Distributed systems)and how users can easily adopt the scheduling policy that improves the performance of a system being simulated[11]. iSPD general architecture, iconic interface, model interpreter, simulation engine and task manager also discussed by the author. The author concluded that the impact of scheduling policies is rather important to achieve optimal performance in distributed systems such as computer grids and their easy modeling should be one of the goals in their simulators.

Satheesh Kumar and Pavan Kumar in their paper “A Scheduling in Heterogeneous computing environment using Alea simulator based on Grid Sim” have focused on the representation of parallel jobs and their execution[12]. They worked on GridSim simulation and ALEA scheduling simulation. This paper provides information regarding the system model, system requirement specifications such as user case & resource case, performance evaluation with results.

They made comparison of various scheduling approaches and evaluated the best performance algorithm for the grid environment since Grid computing is well established and is used for large scale simulations.

Grudenic and N. Bogynovic in their article " Computer Cluster and Grid cluster " addressed the simulation of computer clusters that are the most popular distributed system architectures[13]. They presented the basic building blocks and the architecture of the Grid/cluster simulator. They also discussed the primary features of the available simulators and a new extendable architecture for the new simulator is proposed based on three phase simulation technique.

III. DIFFERENT SIMULATORS

Many systematic investigations have been carried out in the field of resource and job scheduling. There are numerous simulators and toolkits that provide various functionalities for simulations on the Grids, Clusters and Networks. Some popular and most widely used Grid/Cluster simulators are as follows:

A. SimGrid

SimGrid framework enables the simulation of distributed applications in distributed computing environments for the specific purpose of developing and evaluating scheduling algorithms [14]. The Simgrid is a C++-based simulation tool which simulates job scheduling in time-shared architectures, in which a workload can be represented by constants or traces.

In order to run any simulation, SimGrid needs 3 things: something to run (code), a description of the platform on which you want to run your application and finally something regarding to know where to deploy what[15].

Salient features:

SimGrid supports a wide range of features including the following[16][17]:

(1) Support to different network topologies. (2) Resource time sharing and time slicing. (3) Resource models for CPUs and network links. (4) Performance prediction of simulation errors. (5) Arbitrary, dynamic, trace-based resource performance metrics (6) task dependencies. (7) Flexible simulation termination conditions and simple APIs.

Architecture

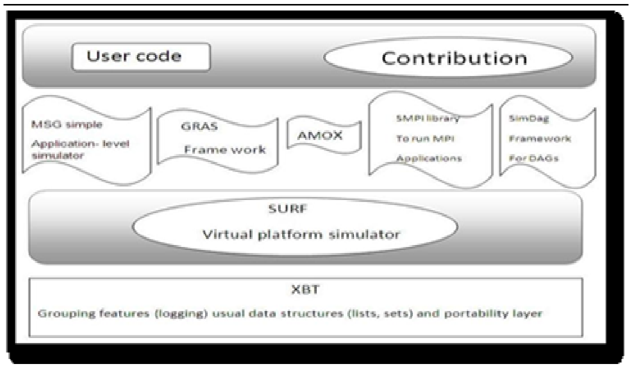

Fig.1 SimGrid's components
(*Source:*http://navet.ics.hawaii.edu/~casanova/corg/simgrid_poster)

SIMGrid V1

This version was simple, and in retrospect a bit naive. However, it was surprisingly useful to study "centralized" scheduling. Although simulating decentralized scheduling with SimGrid v1.0 was actually possible, it was extremely cumbersome and limited in scope. SimGrid v1 was too limiting in terms of scalability and performance[4].

SIMGrid V2

Meta-SimGrid (MSG) added threads and introduced the concept of independently running simulated processes that performed computations and communication tasks in possibly asynchronous fashion. MSG was described towards realistic scheduling simulation of distributed applications[4].
SimGrid v2 also enables the simulation of distributed scheduling agents, which has become critical for current scheduling research in large-scale platforms.

SIMGrid V3

SIMGrid V3 [18] has centralized scheduling without all the power of the MSG API. Another development is that of SMPI, a framework to run unmodified MPI applications in either simulation mode or in real-world mode.

B. ClusterSim

A cluster is composed of single or multi-processing nodes, parallel job schedulers, network topologies and technologies. A workload is represented by jobs submitted by users composed of tasks described by probability distributions and their internal structures.

The ClusterSim is a Java-based parallel discrete-event simulation tool and 3-D animation software for cluster tools, including process modules clustered directly to equipment front-end modules (EFEM) for cluster computing. It supports visual modeling and simulation of clusters and their workloads for performance analysis [19].

New version of ClusterSim, includes two new modules: Message-Passing (MP) and Distributed Shared Memory (DSM) [20,21].

Salient features supported salient features are summarized below[16]:

(1)It provides a graphical environment to model clusters and their workloads.(2)It supports parallelism and its source code is available and its classes are extensible.(3)It provides a mechanism to implement new job scheduling algorithms.(4)It enables simulation of diverse network topologies and it supports the modeling of clusters and heterogeneous or homogeneous nodes.(5)simulation entities are independent threads.

C. BeoSim

BeoSim is a discrete-event simulator aimed at supporting schedulers for grids composed of multiple-cluster such as Beowulf clusters, interconnected through a dedicated network. It offers a GUI to do part of the simulation process but is the only simulator that is not open source[22].

Multi-cluster schedulers make use of all available information pertaining to job communication structure as well as network topology and utilization in order to improve job throughput while mitigating any negative impact to job runtime performance due to network congestion[23]. BeoSim[24][25] tool handles cluster simulation to a greater detail, but the source code and the inside specifics are not publicly available.

Architecture

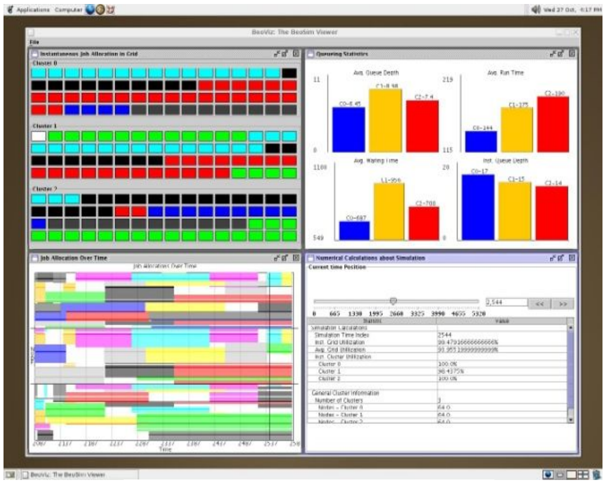

Fig.2 BeoSim's Front-end Visualizer
(Source: http://www.parl.clemson.edu/~wjones/research/)

BeoSim supports the following major features[26]:

(1)It is used for the purpose of studying multi-site parallel job scheduling algorithms in the context of a multicluster computational grids.(2) It enables the valuation of smaller grids under different workloads and scheduling policies.(3)It offers a GUI to do part of the simulation process.

D. GridSim

GridSim is an open source, flexible, modular and universal Grid/Cluster simulation toolkit that will let user perform clustering in distributed computing. It provides a comprehensive facility for simulation of different classes of heterogeneous resources, users, applications, resource brokers, and schedulers. It supports repeatable performance evaluation of scheduling strategies for a range of Grid scenarios[27].

It can be used to simulate application schedulers for single or multiple administrative domains of distributed computing systems such as clusters and grids. Application schedulers in grid environment, called resource brokers, perform resource discovery, selection, and aggregation of a diverse set of distributed resources for an individual user.

It incorporates facilities for identification of failures of Grid resources during runtime. A resource can be a single processor system or multi-processor with shared or distributed memory managed by time or space shared schedulers[28].It incorporates a functionality that reads workload traces taken from supercomputers for simulating a realistic grid environment. The resource brokers use scheduling algorithms or policies for mapping jobs to resources to optimize system or user objectives.

Salient features[29]:

(1)GridSim functionalities include the infrastructure or framework to support advance reservation of a grid system, (2)It permits a data grid extension into GridSim. (3) It supports a network extension into GridSim. (4)Resources and other entities can be linked in a network topology and new allocation policy can be made and integrated into the GridSim Toolkit, by extending the AllocPolicy class.

Architecture

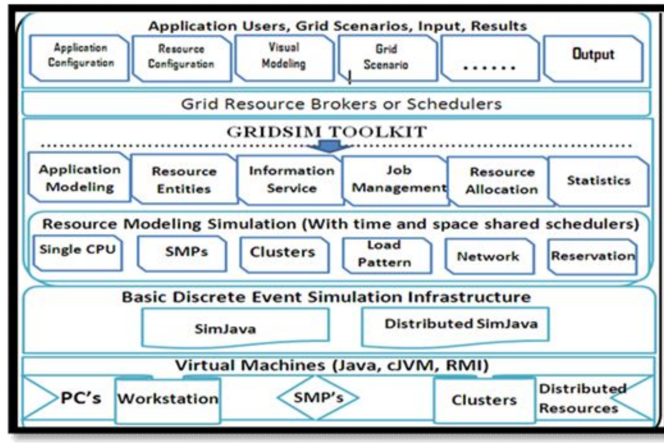

Fig.3 GridSim architecture
(Source:
http://it.wikipedia.org/wiki/File:Architettura_GridSim.JPG)

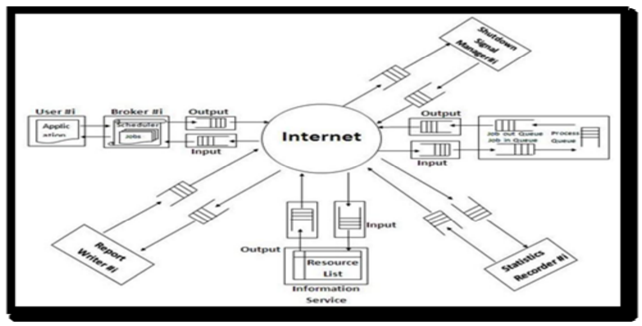

Fig.4 A flow diagram in GridSim based simulator

E. Bricks

Bricks simulate various scheduling schemes on a typical high performance global computing system. Bricks are designed to maximize modularity of restructuring system model based on client-server architecture[30,32]. One may run Bricks to evaluate scheduling heuristics or to evaluate data movement algorithms on grid.

Salient features[31]:

It provides (1)simulation of various behaviors of resource scheduling algorithms, (2)facilitates creation of programming modules for scheduling, (3)enables specification of network topology of clients and servers in global computing systems, and (4)supports various processing schemes for networks and servers.

Architecture

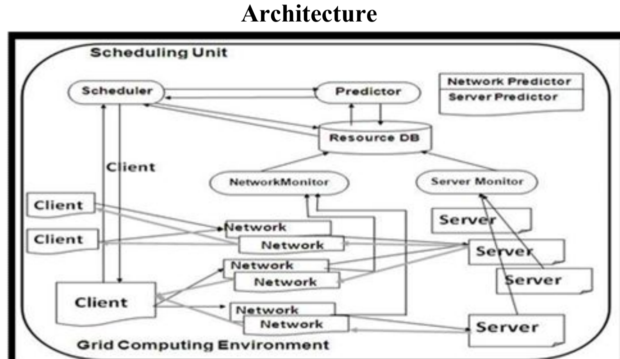

Fig.5 Bricks Simulator
(Source: http://ninf.apgrid.org/bricks/contents/hpdc99.html)

F. Alea 2

The Alea 2 is based on the popular GridSim toolkit and represents a major extension of the Alea simulator, developed in 2007[33]. The extension covers improved design, extended functionality as well as the improved scalability, higher simulation speed and new visualization interface introduced into the simulator.

Alea 2 is an event-based modular simulator, composed of independent entities which implement the desired simulation functionality . AlEA3 is intended to provide a simulation environment that supports simulation of varying Grid/Cluster scheduling problems.

Salient features of Alea 2 are provided in the list given below[34][35]:

(1) It uses advanced scheduling techniques for schedule generation. (2) It supports several new features such as visualization, resource failures and several workload trace formats.(3) The scheduler is capable to handle dynamic situation when jobs appear in the system during simulation.(4) It contains the identification of the message recipient, the type of the event, the time when the event will occur and the message data.

It provides services such as Issue Trackers, Hands-on Builders, Servers and databases and various source data formats like Grid Workload format(GWF), Standard Workload format(SWF), and Metacentric Workload format(MWF).

ALEA3 includes components like Job Loader, Machine Loader, Scheduler, Job Failure and Job Execution with their unique functions[36].

Architecture

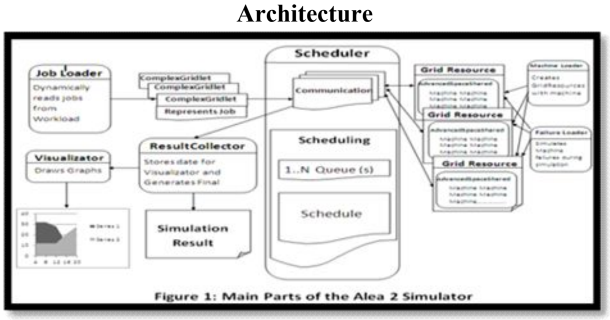

Fig.6 Alea2 Simulator Components (Source: http://www.fi.muni.cz/~xklusac/alea/(Alea))

Components of ALEA2 includeExperimentalSetupclass, MachineLoader, JobLoader,FailureLoader and GridLoader, etc.[37][38]

The *Scheduler* is capable to handle dynamic situation when jobs appear in the system during simulation.

ExperimentSetup class creates instances of the scheduler, the job and machine loader, the failure loader and other entities as required by the standard GridSim.

The MachineLoader entity performs the initialization of the simulated computing environment. It reads the data describing the machines from a file and creates Grid resources accordingly.

The *JobLoader* reads the file containing the job descriptions and creates jobs' instances dynamically over the time.

The *FailureLoader* reads the file containing descriptions of machine failures. Once the simulation time reaches the failure start time, the appropriate machine is set to be failed, killing all jobs being currently executed on that machine.

The resource is represented by the *GridResource* instance and is managed by the local scheduling policy.

Visualizator class generates the simulation's graphical output.

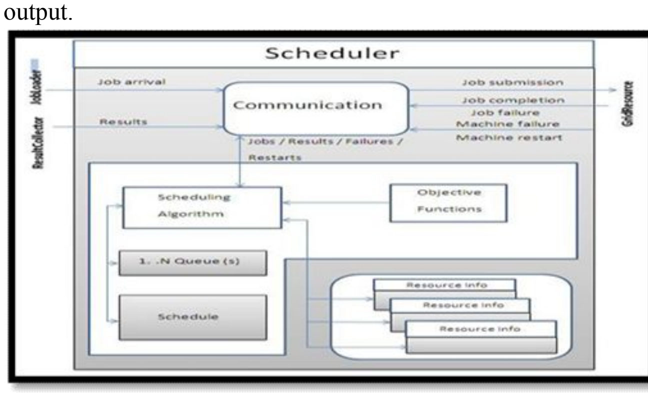

Fig.7 Alea2 Main parts of Scheduler
(Source: http://www.fi.muni.cz/~xklusac/alea /(Alea)

IV. COMPARISON AND ANALYSIS

Comparative evaluation from the viewpoint of general information, capabilities of simulation for different cluster simulators and important features for some popular tools such as SimGrid, ClusterSim, BeoSim, Alea, GridSim and Bricks are presented in this section.

TABLE 1 - PARAMETRIC COMPARISON OF CLUSTER SIMULATION TOOLS

Simulation Tool	Target Applications	Design	Disk I/O overhead	Workload trace format	Modules /projects /entities	Visual -based	Differentiated Network QoS	Lang-uage based
SimGrid	Grid simulation	For the evaluation of scheduling algorithms in distributed applications in heterogeneous, computational Grid environment.	N	Y	PSTSim & DAGSim	N	N	C
ClusterSim	discrete-event simulation	For providing a mechanism to implement new	Y	Y	Message -Passing, Distribut	Y	Y	Java

		job scheduling algorithms.			ed Shared Memory, Equipme nt front-nd			
BeoSim	discrete-eve nt simulation	for studying multi-site parallel job scheduling algorithms in multi-cluster grid.	Y	Y		Y	Y	Java
GridSim	Grid simulation	For designing the network topology, economic analysis of grid infrastructure installation and deployment.	Y	Y	User,bro ker,resou rce,I/o	Y	Y	Java
Bricks	Grid simulation	for studying scheduling algorithms and frame works	Y	Y	Client, Network , Server	Y	Y	Java
Alea	discrete-eve nt simulation	For providing a simulation environment that supports simulation of varying Grid/Cluster scheduling problems	Y	Y	Schedule r, Machine loader, Jobloade r, Failurelo ader	Y	Y	Java

CONCLUDING REMARKS AND FUTURE WORK

Advances in hardware and software technologies have made it possible to deploy distributed and parallel applications over increasingly large sets of distributed resources. In large and distributed environments, it is not possible to evaluate the performance of systems in different scenarios. Simulation appears to be an attractive and feasible way to analyses scheduling algorithms on large-scale distributed systems of heterogeneous resources[39].

It is a very powerful aid for the research on scheduling policies and performance evaluation.

We propose the use of simulators which can give results that are not always experimentally measurable with the current technology. The study supports the belief that simulation does help in the modeling and development of real-world processes, especially in the area of parallel and distributed systems for high performance computing.

This paper provides an overview the cluster simulation tools by outlining some of the tools commonly used in research and conducts a brief survey on these tools[40]. We have highlighted salient features and architecture of various kinds of cluster simulators in this paper.

The work includes comparative study of selected popular cluster simulation tools to help users in making appropriate selection of suitable tools.

An attempt is made in this paper to address the problem of exploring the possibility of designing new algorithms offering better performance and throughput with the help of appropriate simulation tools. Among many possible future works, we are eager to design new scheduling algorithm using simulators for cluster computing.

REFERENCES

[1] http://maggotranch.com/Use%20of%20Cluster%20Computing%20in%20Simulation.html
[2] https://bib.irb.hr/datoteka/402235.Computer_Cluster_and_Grid_Simulator_FINAL.pdf
[3] file:///C:/Documents%20and%20Settings/Administrator.PC69/My%20Documents/Alea%203%20ppr/Meta%20centrum%20Alea%20Simulator.htm
[4] https://bib.irb.hr/datoteka/402235.Computer_Cluster_and_Grid_Simulator_FINAL.pdf
[5] Casanova, "SimGrid: A toolkit for the simulation of application scheduling", Proceedings of the First IEEE/ACM International Symposium on Cluster Computing and the Grid, p. 430-437, 2001.
[6] R. Buyya and M. Murshed, "GridSim: A Toolkit for the Modeling and Simulation of Distributed Resource Management and Scheduling for Grid Computing ",

Journal of Concurrency and Computation: Practice and Experience (CCPE), vol. 14, 2002.

[7] Takefusa, H. Casanova, S. Matsuoka and F. Berman, "A Study of Deadline Scheduling for Client-Server Systems on the Computational Grid" Proc. of 10th IEEE International Symposium on High Performance Distributed Computing, p. 406-415, 2001.

[8] Zhou, B. B., Mackerras, P., Johnson C. W., Walsh, D. "An Efficient Resource Allocation Scheme for Gang Scheduling", 1st IEEE Computer Society International Workshop on Cluster Computing, pp. 187-194, 1999.

[9] Zhang, Y., H. Franke, Moreira, E.J., Sivasubramaniam, A. "Improving Parallel Job Scheduling by Combining Gang Scheduling and Backfilling Techniques", IEEE International Parallel and Distributed Processing Symposium, pp. 133, 2000.

[10] Feitelson, D. G. "Packing Schemes for Gang Scheduling", Workshop on Job Scheduling Strategies for Parallel Processing, pp. 89-110, 1996 Wiley Press, 2004.

[11] Casanova, H., "Simgrid: a Toolkit for the Simulation of Application Scheduling", 3rd IEEE/ACM International Symposium on Cluster Computing and the Grid, Los Angeles, 2001.

[12] Journal of Global Research in Computer Science, Satheesh Kumar N1, Ilayaraja AP2and PNV Pavan3 in their paper "A Scheduling in Heterogeneous computing environment using Alea simulator based on Grid Sim" Volume 3, No. 4, April 2012.

[13] [13]https://bib.irb.hr/datoteka/402235.Computer_Cluster_and_Grid_Simulator_FINAL.pdf

[14] http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=1199362&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxpls%2Fabs_all.jsp%3Farnumber%3D1199362

[15] http://simgrid.gforge.inria.fr/simgrid/3.9/doc/platform.html

[16] http://www.ic.unicamp.br/~luiz.ramos/pdf/iccc04clustersim.pdf (** last site)

[17] http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=1199362&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxpls%2Fabs_all.jsp

[18] http://en.wikipedia.org/wiki/SimGrid

[19] http://electroiq.com/blog/2000/08/3-d-simulation-software-calculates-cluster-tool-throughput/

[20] http://dl.ifip.org/index.php/AICT/article/view/32154

[21] http://cran.r-project.org/web/packages/clusterSim/clusterSim.pdf

[22] http://www.academia.edu/7838046/Scheduler_simulation_using_iSPD_an_iconic-based_computer_grid_simulator

[23] http://www.parl.clemson.edu/~wjones/research/ and http://www.fi.muni.cz/~xklusac/pub/alea2.pdf

[24] W. Jones, L. Pang, D. Stanzione, and W. Ligon III, "Characterization of bandwidth-aware meta schedulers for co-allocating jobs across multiple clusters," J. of Supercomputing , vol. 34, pp. 135–163, 2005.

[25] W. M. Jones, J. T. Daly, , and N. DeBardeleben, "Impact of sub-optimal checkpoint intervals on application efficiency in computational clusters, "in Proceedings of the 19th ACM International Symposium on High-performance Distributed Computing . ACM, pp. 276–279, 2010.

[26] https://bib.irb.hr/datoteka/402235.Computer_Cluster_and_Grid_Simulator_FINAL.pdf

[27] http://arxiv.org/ftp/cs/papers/0203/0203019.pdf

[28] http://www.buyya.com/gridsim/

[29] https://code.google.com/p/gridsim/

[30] http://www.academia.edu/4908816/Fast_simulation_model_for_grid_scheduling_using_HyperSim

[31] http://sameekhan.org/pub/T_K_2013_BC_KZW.pdf

[32] http://books.google.co.in/books?id=DEYIlhwoqrwC&pg=PA205&lpg=PA205&dq=Bricks+cluster+simulator&source=bl&ots=VDsThZzNUw&sig=oHwHeTWF6hsksEQ0Lfifs_w_ieU&hl=en&sa=X&ei=708_VInWLoOQuASTkYKACA&ved=0CFMQ6AEwCA#v=onepage&q=Bricks%20cluster%20simulator&f=false

[33] http://dl.acm.org/citation.cfm?id=1808220

[34] http://www.sitola.cz/wordpress/2009/12/10/alea-2-new-version-of-grid-and-cluster-scheduling-simulator/en/

[35] http://www.fi.muni.cz/~xklusac/alea/

[36] http://www.researchgate.net/publication/234812057_Alea_2_job_scheduling_simulator

[37] https://github.com/aleasimulator/alea

[38] file:///C:/Documents%20and%20Settings/Administrator.PC69/My%20Documents/Alea%203%20ppr/Architecture/Alea%20Simulator.htm

[39] http://www.cloudbus.org/papers/simulationtaxonomy.pdf

[40] http://www.ic.unicamp.br/~luiz.ramos/pdf/iccc04clustersim.pdf

[41] http://www.ic.unicamp.br/~luiz.ramos/pdf/iccc04clustersim.pdf

[42] http://www.fi.muni.cz/~xklusac/alea/

[43] http://www.buyya.com/gridsim/

[44] http://www.ijcsits.org/papers/vol2no62012/33vol2no6.pdf

AUTHORS PROFILE

Ms Rupal Bhatt received her M.Phil degree[Computer Science] in 2010 and pursuing her Ph.D. in Computer Science from Sardar Patel University, V V Nagar, Gujarat.

Dr. D. B. Choksi received his Ph.D.(Computer Science) degree from Sardar Patel University and is working as a a faculty member in the Department of Computer Science of the Sardar Patel University. He received prestigious Young Scientist award from the Indian Science Congress Association in the year 1992. He is a life member of the Indian Science Congress Association (ISCA), Computer Society of India (CSI), Center of the International Cooperation for Computerization (CICC), and Association for Overseas Technical Scholarship (AOTS), Japan. His research interests include system software and computer networks.