Simulation Performance Prediction in Clouds

László Muka and István Derka
Széchenyi István University, Department of Telecommunications, Győr, Hungary
e-mails: muka@sze.hu, steve@sze.hu

***Abstract*—The importance of cloud computing services has been growing rapidly both for users and for service providers of Information and Communications Technology. Together with this process, the cloud computing, as execution environment in parallel and distributed simulation (PADS), has been playing increasing role too. Simulation performance prediction supports efficiently the realization of good performance of PADS. The prediction of behaviour of PADS models in the cloud arises a challenge requiring new approaches in simulation performance prediction. This article discusses how the method used for discrete event simulation performance prediction in heterogeneous network environment can be extended for performance prediction in cloud. An example analysis, showing a potential solution – involving the coupling factor method of performance prediction – is presented for large-scale networks.**

***Keywords*—discrete-event simulation, parallel and distributed simulation, cloud execution environment, coupling factor method, telecommunication systems.**

I. Introduction and Motivation

Over the last few years, the need for the discrete event simulation (DES) of large-scale complex networks and network services has been constantly growing because DES turned to be an efficient tool for the analysis of these systems. The computing capacity requirements large-scale and complex networks can be fulfilled by Parallel and discrete event simulation (PADS) approach [2, 12,13].

According to a common and simple but definition, parallel and distributed simulation (PADS) is any simulation in which more than one processor is used **Error! Reference source not found.**. PADS is the execution of a single discrete event simulation model on a high performance computing platform: on clusters of homogeneous and heterogeneous computers, on WEB, grid and cloud execution environment.

The typical situations of PADS approach applications from the point of view of the runtime performance requirements:

- time critical applications (e.g., real-time decision support, for example in Intelligent Transport System (ITS)) – the time limit is defined and imposed by the modelled system itself, (recently and the decision may require even *extreme fast execution* of prediction
- time consuming applications (e.g., simulation of large and/or complex systems and networks) – the deadline of the task (project) delimits the execution time
    - High Performance Computing (HPC) cloud based application Sim-a-a-S
    - ITS systems cloud application
- on-line applications (e.g., geographically dispersed distant virtual environments for training, playing and WEB applications, ITS applications with good QoE etc.) – the on-line work requirements set the time limits of execution (see later the long-tail problem)

The development and use of systems based on the PDES methods are resource consuming, not easy task even today. The *simulation performance prediction* method is appropriate for support to reach good PADS performance [11].

The question of support may be formulated in the following way: How to build a model with a good parallelization potential and how to execute it with a good runtime performance involving the necessary (available) resources of a parallel and/or distributed environment and how to do it in an efficient way?

The *motivation* of the authors – working on the problems of simulation-based evaluation of communication networks and services – is: the lack of performance prediction methods supporting the planning of cloud simulation execution environment.

The following goals have been set by the authors:
to analyse the influence of latency occurring in cloud execution environment on the performance of the PADS with conservative synchronization method
to establish a method supporting the performance prediction PADS using conservative synchronization protocol.

The main contributions of the paper can be summarized as follows:

- Closed Queuing Network (CQN) Virtual Machine (VM) model of cloud work including modelling of latencies among cloud instances has been defined
- Simulation performance prediction method for the work of conservative synchronization in cloud based on coupling factor approach and on homogeneous network performance prediction has been described

The paper is organized as follows. First, the application PADS introduced, the significance of simulation performance prediction explained, the motivation and goals of authors are described and the contributions of the paper are summarized. Then the issues of execution of PADS in cloud introduced. In the third section, the model of cloud performance of simulation is introduced, the problem of delay in cloud described, the coupling factor concept is shown, the performance prediction for homogeneous cluster – as the tool for prediction support in cloud introduced and the influence of latency on the PADS performance potential in cloud analysed and recommendations are formulated. The fourth section concludes the work.

II. Issues of Simulation Execution in Cloud

A. *Cloud Services*

Cloud computing is a new, evolving approach of computing. The National Institute of Standards and Technology (NIST) defines cloud computing in the following way: ”Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction” [1].

According to definition, the cloud computing has five essential characteristics: on-demand self-service, broad network access, resource pooling, rapid elasticity and measured service.

It has four deployment models private cloud, community cloud, public cloud and hybrid cloud.

Cloud services can be provided according to three service models, e.g., Software as a Service (S-a-a-S), Platform as a Service (P-a-a-S) and Infrastructure as a Service (I-a-a-S).

B. *Parallel and Distributed Simulation in the Cloud*

Cloud can be the execution environment for PADS with huge capacity requirement since it has made accessible high performance computing platforms (HPC) to end users of the cloud. For example, in the Amazon’s very successful Elastic Compute Cloud (EC2) cloud, the Message Passing Interface (MPI) – the standard for parallel programming message communications protocol – is supported by the cloud.

Cloud computing by hiding the problems of parallel and distributed execution and later making its use less risky, can be the method of providing PADS as a service (Sim-a-a-S).

Presently, despite the efforts of cloud service providers – for example, different instances for different use cases in EC2 – to achieve good performance of PADS, careful planning is required concerning the hardware platform on which the PADS designed to be run.

The two main issues that should currently be examined are communications and interference in cloud execution environment.

Communications: Cloud environments are often better at providing high bandwidth communications among applications than in providing low latency [15].

Applications which do not require extensive communications among computing tasks – for example using data processing applications based on MapReduce approach may achieve high performance in cloud.

PADS applications typically work with extensive communications among segments with sending a lot of short messages between the processes. Thus, for PADS good performance quick transport is more important than high bandwidth alone

In the cloud execution environment, for any user, there is no guaranteed access to the resources (resources are virtually assigned to the user) since resources physically are shared among many users.

Problems related with functioning of the cloud, for the PADS applications using optimistic synchronization protocol, may lead to performance degradations [16].

On-line interactive PADS applications have to meet tight time constraints. Even with a good average but high variance in response time would have cause significant decrease of QoE.

For example, the gang scheduling solution – used in the grids – could guarantee the physical resources for a definite user but according to present cloud definition this guarantee may not be given by the cloud provider.

There has been made researches in cloud computing and but less attention have been paid to parallel and distributed simulation and even less to conservative synchronization method.

In the next sections, by outlining a simple performance prediction method for conservative synchronization approach in cloud execution environment, the authors focus will be directed to the conservative synchronization method

III. Simulation Performance in Cloud

A. *General Model of Cloud Performance Simulation*

According to R. Buyya’s definition : “A Cloud is a type of parallel and distributed system consisting of a collection of inter-connected and virtualised computers that are dynamically provisioned and presented as one or more unified computing resources based on service-level agreements established through negotiation between the service provider and consumers.”

In the cloud model the infinite resource pool concept can be taken into account, and the approach of gang scheduling cannot be used.

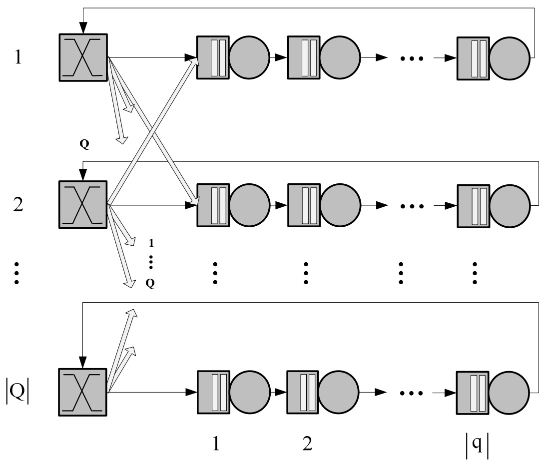


Figure 1. CQN model of PADS execution of simulation

The CQN model with tandems (Q) of simple queues (q) with the modelling of delay in switching between tandems can be used for modelling the PADS performance [3,4,9,10, 14].

B. *About the Probem of Delay in Cloud*

Latency and Jitter in Cloud

In this point, the network performance of EC2 will be discussed, focusing on the measurements of packet delay measurement in spatial experiment published in [5].

In the experiments in [5], the packet round trip delay (RTT) has been measured in EC2, for 750 small instance pairs and 150 medium instance pairs using 5000 ping probes.

For the examined instance pairs, the measured hop count values were within 4 hops.

For the measured instance pairs, the minimum, median, average, maximum RTTs and the RTT standard deviation have been computed.

In Figure 2, the graph (a) shows the cumulative distribution function (CDF) for small instance pairs, graph (b) shows the CDF of RTT for medium instances.

Both graphs show, that the RTT values among the examined instances are not stable.

The RTT values are between 0.2 and 0,3ms for most of the small and medium instance pairs.

- However, on 55% of the small instance pairs, the maximum RTTs are higher than 20ms.
- For the delays of medium instances, it is observed that, for 20% medium instance pairs, the maximum RTTs are larger than 10ms.

Long-tail Problem

Latency in clouds (in EC2 too) is an important problem for on-line, interactive applications, especially with extreme round-trip times (RTTs). There can be customer requests will suffer an unacceptable delay with (QoS degradation). In these, network situations the cause is the long-tail behaviour [6].

A rapidly increasing number of Internet-scale applications are deployed and are relying heavily on Amazon's EC2 cloud.

Ref. [6] using controlled experiments analysis the problem, focusing on the tail of round-trip latency due to its disproportionate impact on user experience. There has been determined the impact of bad nodes on the tail completion time of the partition-aggregation model with 10, 20, and 40 nodes involved in the workloads.

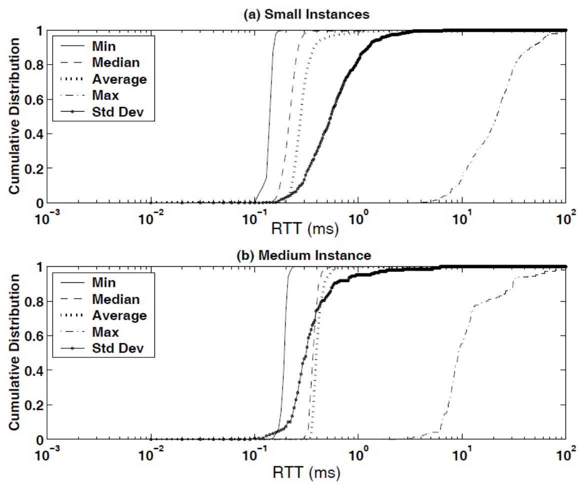


Figure 2. CDF of RTTs for small (a) and medium (b) sized instances in EC2. The x-axis given in log scale. (The figure is taken from [5].)

C. *The Coupling Factor Approach Concept*

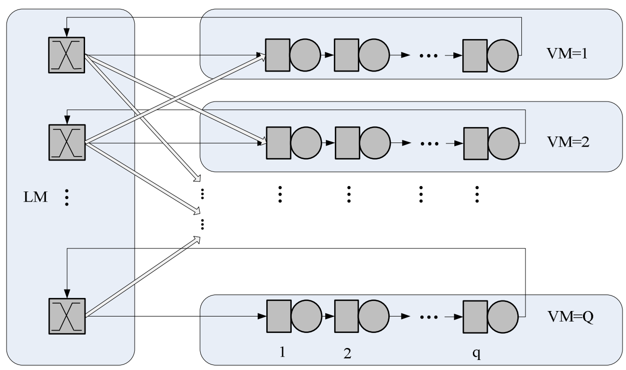


Figure 3. CQN of PADS cloud execution with VMs as and with LM latency modelling switching

The principle of the coupling factor prediction method [7, 9] may be formulated as an inequity:

$$L * E \gg \tau * P$$

where L is the lookahead value characterizing the model (simsec), E is the event density generated by the model (ev/simsec), is the latency of messages between logical process (LPs) of the model (sec), and P is the event processing computation hardware performance (ev/sec).

According to the method, the coupling factor is calculated according to the formula

$$\lambda = \frac{L * E}{\tau * P}$$

TABLE I.
COUPLING FACTOR MEASUREMENT AND CALCULATION

1.	L	[simsec]	0.1	1	10	100	1000
2.	Number of events	[ev]	138122606	138091806	137816386	134885378	102957082
3.	WCT (N=1)	[sec]	524	521	523	516	416
4.	Simulated virtual time	[simsec]	864000	864000	864000	864000	864000
5.	P	[ev/sec]	263502	264868	263465	261132	247653
6.	E	[ev/simsec]	159	159	159	156	119
7.	τ	[sec]	0.000025	0.000025	0.000025	0.000025	0000025
8.	R=P/E	[simsec/sec]	1648	1657	1651	1672	2078
9.	L/ τ	[simsec/sec]	4000	40000	400000	4000000	40000000
10.	λ measured		2.43	24.1	242	2391	19246

The high value of the coupling factor shows the good potential of the simulation model for parallelization. The formula involves only four parameters for the calculation which can be measured in simple sequential simulation runs.

For a separate process, the λ_N parallelization potential of a process is only a part of the whole potential:

$$\lambda_N = \frac{L * E}{\tau * P} * \frac{1}{N_{LP}}$$

where N_{LP} the number of the LPs [8].

D. *Performance Prediction for Homogeneous Cluster*

For the presented method, the results described in [8] will be used in the cloud performance analysis.

The constraints and results of experiments performed in [8] are summarized in the following. (Figure 4. contains the results of the relative speedup analysis too.)

The problem class is communication network modelling, the selected approach is parallel and distributed discrete event simulation, for which the synchronization protocol is conservative synchronization with null message algorithm. The software environment includes: Linux (Debian) operating system, MPI, NFS, Omnet++ network simulator.

The hardware environment is a *homogeneous cluster* of 12 PCs.

The experimentation/evaluation model (Figure 1.) is the CQN model (in OmNet++) with FCFS service discipline. For the experiments in the case, the starting number of jobs is 2 jobs/simple queue with exponential inter-arrival time and with exponential service time distributions (the expected value for arrival and service time is 10sec). The delay on links between simple queues is 1sec. The number of tandem queues is 24 (Q), the number of simple queues/tandem queue is 50 (q). The switching in CQN model is performed by uniform distribution to switch to the next tandem queue, and the delay of switching models lookahead.

The task assignment principles in the execution are the partitioning into LPs and Load Balancing Criterion. The measured value of τ is 0,025ms for the system in the case.

The value of variables that have been measured in sequential simulation runs are shown in Table I.

TABLE II.
COUPLING FACTOR MEASUREMENT AND CALCULATION

1.		RTT(ms)	latency(ms)	τ latency	proportion	λ shift (log10)
2.	CQN model τ		0,025			
3.	EC2 average	0,2	0,1	4		0,60
4.	EC2 average	3	0,15	4,2		0,62
5.	EC2 small instances	10	5	200	55%	
6.	EC2 medium instances	20	10	400	20%	
7.	EC2 small instances weighted τ latency		2,84	113,6		2,06
8.	EC2 medium instances weighted τ latency		2,16	86,4		1,94

E. *Analysis*

In Table II. the published EC2 latency measurements and the calculated λ shifts in predicted relative speedup function are summarized. Figure 4 shows the predicted degradation of the of the PADS performance in loud – predicted according to coupling factor method and validated on CQN model execution – caused by occurring latencies. In Figure 4, there have been presented two examples: the arrows show two decreasing of speedup values to positions denoted with yellow colours for (virtual) processor numbers N=12 and N=2. The value of shift approximately equal for small and medium instances are, 2,06 and 1, 94 respectively.

F. *Recommendations*

The analysis shows that to avoid degradation of relative speedup potential of PADS with conservative synchronization protocol in cloud execution environment, it is recommended:

- -to work with high λ (the decrease of predicted relative speedup is lower for the shift in λ for 19246→2931 than for 242 →2,43). The influence of latency can be compensated by lookahed.
- to limit the number of involved processors (involved resources) and it supports efficiency. (The decrease processor number N=12 is higher than for N=2.)

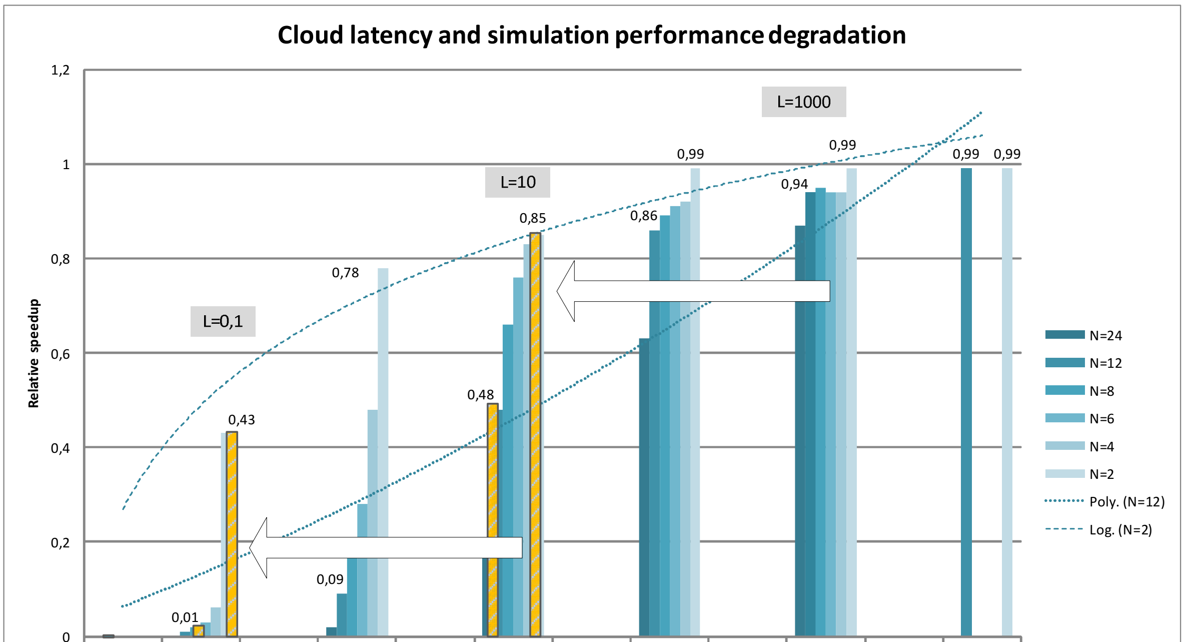
IV. CONCLUSION

We have described the influence of cloud latency on the performance of PADD with conservative synchronization and the long-tail RTT distribution on on-line interactive simulation.

We have described the model of cloud work which models the cloud latencies too.

We have formulated the cloud PADS execution performance prediction based on coupling factor method and on homogeneous network performance model with VMs.

We have introduced an example performance analysis of cloud latencies impacts with different EC2 instances using a scale that has been set up by homogeneous performance model with VMs.



	f(latency)1	λ=2,43	f(latency)2	λ=24,1	f(latency)3	λ=242	f(latency)4	λ=2391	f(latency)5	λ=19246	f(latency)6	λ=192460
N=24		0		0,02		0,18		0,63		0,87		
N=12		0,01		0,09		0,48		0,86		0,94		0,99
N=8		0,02		0,18		0,66		0,89		0,95		
N=6		0,03		0,28		0,76		0,91		0,94		
N=4		0,06		0,48		0,83		0,92		0,94		
N=2		0,43		0,78		0,85		0,99		0,99		0,99

Figure 4. CQN VMs for cloud

We have formulated recommendations how to avoid the influence of cloud latencies.

Our future work may be directed on the analysis performance degradation influence of infinite cloud resource pool, on the detailed research of long-tail caused performance problem.

ACKNOWLEDGEMENT

This research was supported by the TÁMOP-4.2.2.C-11/1/KONV-2012-0012: “Smarter Transport” – IT for co-operative transport system. The Project is supported by the Hungarian Government and co-financed by the European Social Fund and partly supported by the Faculty of Engineering of Széchenyi István University 15-3203-08.

REFERENCES

[1] P. Mell, and T. Grance. The NIST definition of Cloud Computing, National Institute of Standards and Technology (NIST), Special Publication Draft-800-145, 2011. p. 2.

[2] R. M. Fujimoto, “Parallel Discrete Event Simulation”, *Communications of the ACM*, vol. 33, (1990.) no 10, pp. 31-53

[3] Lencse, G., I. Derka and L. Muka. 2013. "Towards the efficient simulation of telecommunication systems in heterogeneous execution environments". submitted to the *36th International Conference on Telecommunications and Signal Processing (TSP), July 2-4, 2013, Rome, Italy*.

[4] R. L. Bagrodia, M. Takai, (2000). “Performance evaluation of conservative algorithms in parallel simulation languages”, *Parallel and Distributed Systems, IEEE Transactions on*, *11*(4), 395-411.

[5] Wang, G., and Ng, T. E. (2010, March). The impact of virtualization on network performance of amazon ec2 data center, In INFOCOM, 2010 Proceedings IEEE, pp. 1-9.

[6] Xu, Y., Musgrave, Z., Noble, B., Bailey, M. (2013, April). Bobtail: Avoiding Long Tails in the Cloud. In NSDI pp. 329-341.

[7] A. Varga, Y. A. Sekercioglu and G. K. Egan. “A practical efficiency criterion for the null message algorithm”, *Proceedings of the European Simulation Symposium (ESS 2003)*, (Oct. 26-29, 2003, Delft, The Netherlands) SCS International, 81-92.

[8] G. Lencse and A. Varga, “Performance Prediction of Conservative Parallel Discrete Event Simulation”, *Proceedings of the 2010 Industrial Simulation Conference (ISC'2010)* (Budapest, Hungary, 2010. June 7-9.) EUROSIS-ETI, 214-219.

[9] A. Varga and R. Hornig, “An overview of the OMNeT++ simulation environment”, Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops, (Marseille, France, March 3-7, 2008) 1-10.

[10] A. Varga and A. Y. Sekercioglu, “Parallel Simulation Made Easy with OMNeT++”, *Proceedings of the 15th European Simulation Symposium (ESS 2003)*, (Oct. 26-29, 2003, Delft, The Netherlands) SCS International, 493-499.

[11] Z. Juhasz, S. Turner, K. Kuntner, M. Gerzson, “A Performance Analyser and Prediction Tool for Parallel Discrete Event Simulation”,*International Journal of Simulation*, vol. 4, no. 1, May 2003. pp. 7-22.

[12] Kunz, G., Tenbusch, S., Gross, J., Wehrle, K. (2011, July). “Predicting Runtime Performance Bounds of Expanded Parallel Discrete Event Simulations. In *Modeling, Analysis & Simulation of Computer and Telecommunication Systems (MASCOTS), 2011 IEEE 19th International Symposium on* (pp. 359-368). IEEE.

[13] R. Ewald, A. M. Uhrmacher, “Automating the runtime performance evaluation of simulation algorithms”, In: *Simulation Conference (WSC), Proceedings of the 2009 Winter*. IEEE, 2009. pp. 1079-1091.

[14] Lencse, G. and I. Derka 2013. "Testing the Speed-up of Parallel Discrete Event Simulation in Heterogeneous Execution

Environments". submitted to the *2010 Industrial Simulation Conference (ISC'2010)* (Ghent, Belgium, May 22-24, 2013.)

[15] Malik, A. W., Park, A. J., Fujimoto, R. M. (2010). An optimistic parallel simulation protocol for cloud computing environments", *SCS Modeling and Simulation Magazine*, 1(4).

[16] Fujimoto, R. M., Malik, A. W., Park, A. (2010). "Parallel and distributed simulation in the cloud*" SCS M&S Magazine,* 3, pp. 1-10.

BIOGRAPHIES

LÁSZLÓ MUKA graduated in electrical engineering at the Technical University of Lvov in 1976. He got his special engineering degree in digital electronics at the Technical University of Budapest in 1981, and became a university doctor in architectures of CAD systems in 1987. Mr. Muka finished an MBA at Brunel University of London in 1996. Since 1996 he has been working in the area of modeling and simulation of infocommunications systems, including human subsystems. Mr. Muka got his PhD at the Budapest University of Technology and Economics in 2011. Dr. Muka has been working as an Associate Professor for the Széchenyi István University in Győr since the February of 2012. He teaches system simulation and network security

ISTVÁN DERKA received his Msc at Faculty of Electrical Engineering and Informatics at the Technical University of Budapest in 1995. He worked for the Department of Informatics from 1999 to 2003 and since then has been working for Department of Telecommunications, Széchenyi István University in Győr. He teaches Programming of communication systems and Interactive TV systems. He is an Assistant Professor. The area of his research includes multicast routing protocols and IPTV services in large scale networks.