Proceedings of the 2016 Winter Simulation Conference
T. M. K. Roeder, P. I. Frazier, R. Szechtman, E. Zhou, T. Huschka, and S. E. Chick, eds.

OPTIMAL EXECUTION OF LARGE SCALE SIMULATIONS IN THE CLOUD. THE CASE OF ROUTE-TO-PA SIM ONLINE PREFERENCE SIMULATION

Przemysław Szufel
Marcin Czupryna
Bogumił Kamiński

Warsaw School of Economics
Al. Niepodległości 162
02-554 Warszawa, Poland

ABSTRACT

Cloud computing enables massive parallelization of execution of large scale simulation experiments but it is complex to do it in a cost-efficient way. We present methodology used to achieve this goal that was devised in the ROUTE-TO-PA project, where we develop a simulator for generalization of the dynamics of preferences observed on the social platform to the entire population. Experimenting with such a complex simulation model over a computing cluster in the cloud requires solving not only technical challenges (solution architecture and management of dynamically changing infrastructure) but also requires optimization of computing cost. In this work we present our approach (ROUTE-TO-PA SIM) to configure and manage such environment in the Amazon Web Services cloud setting.

1 SIMULATION IN THE CLOUD: TYPICAL CHALLENGES

Scientists planning to run large scale simulations in the cloud face a very complex ecosystem with non-trivial architectural decisions to be made. In particular we have identified the following issues while running a large scale social simulation in the cloud: (1) computation cost optimization, (2) computational algorithm parallelization, (3) controlling job execution errors, (4) management of failures of computing nodes, (5) input data processing and storage, (6) output data collection and (7) output data aggregation.

Our architectural decision where mostly determined by the cost factor. The public cloud offering is very vast with several vendors and numerous pricing schema. For most simulation problems a cost effective solution is to purchase computing power on Amazon Web Services EC2 spot market. This choice however, leads to the need for automating spot bidding decisions and management of unexpected shutdowns of computing nodes, see (Kamiński and Szufel 2015).

In this work firstly, in Section 2, we describe technical characteristics of our example simulation model (ROUTE-TO-PA SIM) and secondly, in Section 3, we propose a framework along with a set of architectural decisions and guidelines for running large scale social simulations in the cloud.

2 ROUTE-TO-PA SIM

In the ROUTE-TO-PA SIM model we consider a scenario where a decision maker (public administration) uses an online social platform to collect information on citizens' preferences. However, opinions of the sub-population that uses the online platform might be not representative. Hence, there is a need to develop a preference generalization algorithm in a social network setting. The available data includes population census databases along with full information from the online social platform.

We have implemented an agent-based simulation model and subsequently test various opinion propagation algorithms for different social network topologies and diffusion dynamics scenarios. The simulation

model has been implemented in Java (MASON simulation library and JUNG for network manipulation), the project repository, including the codes described in Section 3, is reachable form the ROUTE-TO-PA project website http://routetopa.eu/.

3 DISTRIBUTED ARCHITECTURE FOR SIMULATING SOCIAL NETWORK IN THE CLOUD

In Section 1 we have identified the problem areas that need to be solved when running large scale simulations in the cloud. These include cost optimization as well as managing distributed computations and data.

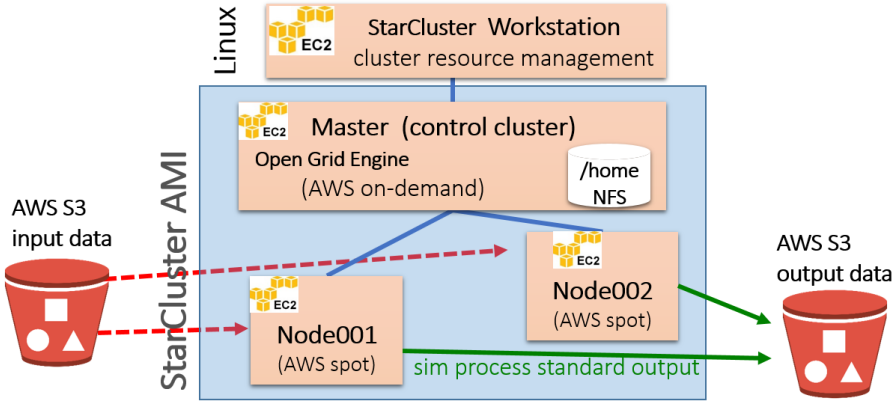

Figure 1: Cloud simulation environment with a master node and two sample worker nodes. AWS S3 data storage is used for distributed provisioning of simulation data as well as collecting process output.

Figure 1 presents architectural layout of our simulation environment deployed using AWS spot instances. We have considered two open source tools for infrastructure management: *StarCluster* and *AWS cfncluster*. The first environment is much more mature and hence more appropriate for deployment on AWS spot instances. Currently, running StarCluster requires updating the stable version with at least instance type list from pre-release available at the Git repository. Additionally we have added support for AWS roles for cluster nodes. Hence, it is not required to store full authentication information on cluster nodes. The nodes can directly read and write to AWS S3.

Running simulations on spot requires to frequent (recommended at least every 3 hours) computational state/results check-pointing. We run each simulation as a separate process (batch job). Code has been developed to catch standard output and automatically compress it and upload to S3 once the process ends. This approach allows to checkpoint the state just after every simulation is complete. When a EC2 spot instance is lost (e.g. due to spot price spike) the computational state is safely stored in a S3 bucket.

Some simulation data is stored in a local database. In order to avoid cross process locks we replicate database by number of cores in a node. We developed locking-prevention solution where logs are being done on directories in a local file system of a node.

The proposed approach enables a robust execution of simulation experiments in a cloud setting and result aggregation and collection.

ACKNOWLEDGEMENTS

This project has received funding from the European Unions Horizon 2020 research and innovation programme under grant agreement No 645860.

REFERENCES

Kamiński, B., and P. Szufel. 2015. “On optimization of simulation execution on Amazon EC2 spot market”. *Simulation Modelling Practice and Theory* 58:172–187.