

# WORKLOAD, RESOURCE, AND PRICE AWARE PROACTIVE AUTO-SCALAR FOR DYNAMICALLY-PRICED VIRTUAL MACHINES



Daham Positha Pathiraja

179340D

M.Sc. in Computer Science

Department of Computer Science and Engineering

University of Moratuwa

Sri Lanka

March 2019

# WORKLOAD, RESOURCE, AND PRICE AWARE PROACTIVE AUTO-SCALAR FOR DYNAMICALLY-PRICED VIRTUAL MACHINES



Daham Positha Pathiraja

179340D

This report is submitted in partial fulfillment of the requirements for the  
Degree of  
Master of Science in Computer Science specializing in Cloud Computing

M.Sc. in Computer Science

Department of Computer Science and Engineering

University of Moratuwa

Sri Lanka

March 2019

## Declaration

I, D.P. Pathiraja, hereby declare that this is my own work and this report does not incorporate without acknowledgement any material previously submitted for the degree or diploma in any other university or institute of higher learning and to the best of my knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made in the text.

Also, I hereby grant University of Moratuwa the non-exclusive right to reproduce and distribute my dissertation, in whole or in part in print, electronic or other medium. I retain the right to use this content in whole or part in future works (such as articles of books).

Signature: .....

Date: .....

Name: D. P. Pathiraja

I certify that the above candidate has carried out research for the Masters thesis under my supervision.

Signature: .....

Date: .....

Name of the supervisor: Dr. H. M. N. Dilum Bandara

## Abstract

Proactive Cloud auto-scalers forecast future conditions and initiate scaling response in advance leading to better service quality and cost savings. Their effectiveness depends on the forecast accuracy and penalty due to miss prediction. However, such solutions assume fixed prices for virtualized Cloud resources to be provisioned. Hence, they are unable to benefit from dynamically-priced resources such as Amazon Spot Instances which are introduced by Cloud providers to deal with fluctuating workloads cost effectively. Moreover, users have the risk of losing resources when the dynamically-adjusted market price of resources exceeds the user-defined maximum bid price. Therefore, proactive auto-scalers should also forecast market price of dynamically-priced resources to minimize the cost further while retraining service quality. However, predicting the market price (to set the maximum bid price) is quite complicated given highly varying workload and resource demands. We present a proactive auto-scaler for dynamically-priced virtual machines by combing the workload and resource prediction capabilities of an existing auto-scaler named IntelliScaler, and a novel technique for forecasting Spot price. We retrieve Spot price history from Amazon and use it to forecast the future prices using Recurrent Neural Networks. Next, we selected the maximum price for a given decision window as the bid value to make Spot request. To demonstrate the utility of the proposed solution, we tested the performance of the enhanced auto-scaler using a synthetic workload generated using the Rain toolkit and the RUBiS auction site prototype. Proposed auto-scaler with dynamically-priced virtual machines reduced the total cost by ~75% compared the same auto-scaler with fixed priced instances. Moreover, no noticeable change in service quality was observed.

## **Acknowledgement**

I owe my deepest gratitude to my supervisor, Dr. Dilum Bandara of Department of Computer Science, Faculty of Engineering, University of Moratuwa, for his invaluable support in providing relevant knowledge, advice and supervision throughout the project. His continuous guidance, inspiring advices and constructive feedback provided this project a great value and an encouraging background throughout the project. This would not have been possible without his expertise and remarkable support.

I take this opportunity to convey our gratitude towards my batch mates, family members and friends who helped me to do my best towards achieving success during the entire project time period.

Finally, I would like to thank my colleagues at Sysco Labs (Pvt.) Ltd for covering my work and helping me to balance the workload. Without them, this project would not have been possible.

Last but not least, I am grateful for all the people who supported me throughout this research in various means.

# Table of Contents

Declaration .....	i
Abstract .....	ii
Acknowledgement.....	iii
Table of Contents .....	iv
List of Figures .....	vii
List of Tables.....	ix
Abbreviations .....	x
1. Introduction.....	1
1.1. Background .....	1
1.2. Motivation .....	3
1.3. Problem Statement .....	5
1.4. Objectives .....	5
1.5. Outline .....	6
2. Literature Review .....	7
2.1. Types of Virtualized Instances .....	7
2.1.1. Reserved Instances .....	7
2.1.2. On-demand Instances .....	7
2.1.3. Dynamically-priced Instances .....	8
2.2. Proactive Auto Scaling .....	11
2.3. Handling Spot Instances .....	14
2.3.1. Predict Biddings .....	14
2.3.2. Bidding Strategies .....	15
2.3.2.1. Always Bidding Maximum Price .....	15
2.3.2.2. AMAZING: Constrained Markov Decision Process (CMDP)- based Dual-Option Bid Strategy .....	16
2.3.3. Checkpointing .....	16
2.3.3.1. Hour-boundary Checkpointing .....	17
2.3.3.2. Rising Edge-driven Checkpointing.....	17
2.3.3.3. Checkpointing with Adaptive Decision .....	18
2.4. Spot Price Prediction Techniques.....	18

2.4.1.	Autoregressive Integrated Moving Average (ARIMA) model .....	19
2.4.2.	Long Short Term Memory (LSTM RNN) model .....	20
2.4.2.1.	Neural Networks .....	20
2.4.2.2.	Recurrent Neural Networks .....	21
2.4.2.3.	Long Short Term Memory (LSTM) Networks .....	22
2.5.	Summary .....	23
3.	Methodology .....	25
3.1.	High-Level Architecture.....	25
3.2.	Workload Predictor .....	27
3.2.1.	InteliScaler Prediction Algorithm.....	27
3.3.	Price Predictor .....	28
3.4.	Prediction Store .....	28
3.5.	Bid Price Selector .....	29
3.6.	Summary .....	29
4.	Proposed Solution .....	30
4.1.	Workload Prediction.....	30
4.1.1.	Determination of Contribution Coefficients .....	32
4.1.2.	Selection of Models.....	32
4.1.3.	Prediction Algorithm.....	33
4.2.	Price Prediction .....	34
4.2.1.	Retrieving Spot Pricing History from AWS .....	34
4.2.2.	Applying Famous Prediction Models .....	36
4.3.	Proposed auto-scaling algorithm .....	39
4.4.	Summary .....	39
5.	Performance Analysis .....	41
5.1.	Implementation.....	41
5.2.	Optimal Bid Selection Procedure .....	43
5.3.	Simulation-Based Evaluation .....	45
5.3.1.	Simulation Setup .....	45
5.3.1.1.	Proactive Auto Scaling .....	45
5.3.2.	Scaling Out with Homogeneous Units.....	47

5.3.3.	Scaling Out with Heterogeneous Units .....	48
5.4.	Empirical Evaluation .....	51
5.4.1.	Experimental Setup .....	51
5.4.2.	Quality of Service Measurements .....	56
6.	Summary .....	61
6.1.	Conclusion .....	61
6.2.	Research Limitations .....	63
6.3.	Future Work .....	63
	References .....	65



# List of Figures

Figure 2.1 : Spot price fluctuations of us-east-1 linux.c1.medium Instance type.....	9
Figure 2.2: How Amazon EC2 charges per-hour price for using a Spot Instance..	4610
Figure 2.3: System Level Architecture of InteliScaler.....	12
Figure 2.4: Hour-boundary checkpointing.....	17
Figure 2.5: Rising-edge driven checkpointing.....	18
Figure 2.6: When user's Spot instance is out of bid .....	19
Figure 2.7: Chunk of a Recurrent Neural Network (RNN).....	21
Figure 2.8: An Unrolled Recurrent Neural Network (RNN). .....	22
Figure 2.9: RNNs perform well when we need more contexts.....	23
Figure 2.10: RNNs do not perform well when we need more contexts.....	23
Figure 2.11: Repeating module in LSTM. ....	23
Figure 3.1: High-level architecture of proposed workload and resource aware auto scaler for Spot instances.....	27
Figure 4.1: Spot instance pricing history. ....	35
Figure 4.2: Sample describe-spot-price-history API response.....	36
Figure 4.3: ARIMA prediction of Spot price for Spot instance type r5.xlarge.....	37
Figure 4.4: LSTM RNN prediction of Spot price for Spot instance type r5.....	38
Figure 5.1: Simulation Setup.....	41
Figure 5.2: Decide the best bid value for m4.4xlarge & m4.2xlarge Spot instances when scaling decision is made. ....	44
Figure 5.3: Penalty factor over violation percentage. ....	46
Figure 5.4: Resource Allocation .....	47
Figure 5.5: Variation of Virtual Machine Count and Load Average Overtime with Novel Approach for Scaling Out with Homogeneous Units for Confident Interval 6 hours.....	47
Figure 5.6: Variation of cost with time for different auto-scaling solutions when Spot confident interval is 6-hours.....	48
Figure 5.7: Variation of Virtual Machine Count and Load Average Overtime with Novel Approach for Scaling Out with Heterogeneous Units for Confident Interval 6 - hours.....	49

Figure 5.8: Variation of cost with time for different auto-scaling solutions when Spot confident interval is 6-hours.....	50
Figure 5.9: Simulation setup that mimic InteliScaler .....	53
Figure 5.10: Simulation setup that mimic novel auto scaler for Spot instances .....	54
Figure 5.11: Average response time of selected endpoints of RUBiS web application .....	60

## List of Tables

Table 5.1: Violation Cost, Total Cost and Violation Percentage after 7-hours with Novel Approach for Scaling Out with Homogeneous Units for Confident Interval 6 - hours.....	47
Table 5.2: Violation Cost, Total Cost and Violation Percentage after 7-hours with Novel Approach for Scaling Out with Heterogeneous Units for Confident Interval 6 - hours.....	51
Table 5.3: Violation Cost, Total Cost and Violation Percentage after 7-hours for different auto-scaling methods.....	51
Table 5.4: Workload profile for simulation set-up.....	56
Table 5.5: Results in the simulation setup for mimic InteliScaler.....	56
Table 5.6: Scaling Decisions – Iteration 1.....	57
Table 5.7: Scaling Decisions – Iteration 2.....	57
Table 5.8: Scaling Decisions – Iteration 3.....	57
Table 5.9: Results of mimic novem price and workload aware auto sacaler for Spot Instances.....	57
Table 5.10: Scaling Decisions – Iteration 1.....	57
Table 5.11: Scaling Decisions – Iteration 2.....	58
Table 5.12: Scaling Decisions – Iteration 3.....	59

## Abbreviations

APE	Absolute Percentage Error
ANN	Artificial Neural Network
ARIMA	Autoregressive Integrated Moving Average
ARMA	Auto Regressive
AWS	Amazon Web Services
CMDP	Constrained Markov Decision Process
EC2	Elastic Compute Cloud
IaaS	Infrastructure as a Service
IOPS	IO Operations per Second
LA	Load Average
MAPE	Mean Absolute Percentage Error
PaaS	Platform as a Service
POC	Proof of Concepts
PTPM	Price Transition Probability Matrix
QoS	Quality of Service
RMSE	Root Mean Square Error
RUBiS	Rice University Bidding System
VM	Virtual Machine