AICT 2021

The Seventeenth Advanced International Conference on Telecommunications

ISBN: 978-1-61208-860-0

978-1-61208-860-0

AICT 2021 Editors

Eugen Borcoci, University "Politehnica" of Bucharest (UPB), Romania

AICT 2021

Foreword

The Seventeenth Advanced International Conference on Telecommunications (AICT 2021), held between May 30 – June 3rd, 2021 covered a variety of challenging telecommunication topics ranging from background fields like signals, traffic, coding, communication basics up to large communication systems and networks, fixed, mobile and integrated, etc. Applications, services, system and network management issues also received significant attention.

The spectrum of 21st Century telecommunications is marked by the arrival of new business models, new platforms, new architectures and new customer profiles. Next generation networks, IP multimedia systems, IPTV, and converging network and services are new telecommunications paradigms. Technology achievements in terms of co-existence of IPv4 and IPv6, multiple access technologies, IP-MPLS network design driven methods, multicast and high speed require innovative approaches to design and develop large scale telecommunications networks.

Mobile and wireless communications add profit to large spectrum of technologies and services. We witness the evolution 2G, 2.5G, 3G and beyond, personal communications, cellular and ad hoc networks, as well as multimedia communications.

Web Services add a new dimension to telecommunications, where aspects of speed, security, trust, performance, resilience, and robustness are particularly salient. This requires new service delivery platforms, intelligent network theory, new telecommunications software tools, new communications protocols and standards.

We are witnessing many technological paradigm shifts imposed by the complexity induced by the notions of fully shared resources, cooperative work, and resource availability. P2P, GRID, Clusters, Web Services, Delay Tolerant Networks, Service/Resource identification and localization illustrate aspects where some components and/or services expose features that are neither stable nor fully guaranteed. Examples of technologies exposing similar behavior are WiFi, WiMax, WideBand, UWB, ZigBee, MBWA and others.

Management aspects related to autonomic and adaptive management includes the entire arsenal of self-ilities. Autonomic Computing, On-Demand Networks and Utility Computing together with Adaptive Management and Self-Management Applications collocating with classical networks management represent other categories of behavior dealing with the paradigm of partial and intermittent resources.

We take here the opportunity to warmly thank all the members of the AICT 2021 Technical Program Committee, as well as the numerous reviewers. The creation of such a broad and high quality conference program would not have been possible without their involvement. We also kindly thank all the authors who dedicated much of their time and efforts to contribute to AICT 2021. We truly believe that, thanks to all these efforts, the final conference program consisted of top quality contributions.

Also, this event could not have been a reality without the support of many individuals, organizations, and sponsors. We are grateful to the members of the AICT 2021 organizing committee for their help in handling the logistics and for their work to make this professional meeting a success.

We hope that AICT 2021 was a successful international forum for the exchange of ideas and results between academia and industry and for the promotion of progress in the field of telecommunications.

AICT 2021 Chairs:

AICT 2021 Steering Committee
Dragana Krstic, University of Niš, Serbia
Kevin Daimi, University of Detroit Mercy, USA
Gautam Srivastava, Brandon University, Canada
Sergei Semenov, HiSilicon, Sweden

AICT 2021 Publicity Chairs
Marta Botella-Campos, Universitat Politecnica de Valencia, Spain
Daniel Basterretxea, Universitat Politecnica de Valencia, Spain

AICT 2021

COMMITTEE

AICT 2021 Steering Committee

Dragana Krstic, University of Niš, Serbia
Kevin Daimi, University of Detroit Mercy, USA
Gautam Srivastava, Brandon University, Canada
Sergei Semenov, HiSilicon, Sweden

AICT 2021 Publicity Chairs

Marta Botella-Campos, Universitat Politecnica de Valencia, Spain
Daniel Basterretxea, Universitat Politecnica de Valencia, Spain

AICT 2021 Technical Program Committee

Ghulam Abbas, GIK Institute, Pakistan
Iwan Adhicandra, University of Sydney, Australia
Abdulazaz Albalawi (Aziz), University of California, Santa Cruz, USA
Michele Albano, Aalborg University, Denmark
Nicolae Dumitru Alexandru, "Gheorghe Asachi" Technical University of Iasi, Romania
Marco Aurélio Spohn, Federal University of Fronteira Sul, Brazil
Ilija Basicevic, University of Novi Sad, Serbia
Oussama Bazzi, Lebanese University, Lebanon
Carlos Becker Westphall, Federal University of Santa Catarina, Brazil
Sara Behjatjamal, Altinbas University, Turkey
Stefano Berretti, University of Firenze, Italy
Robert Bestak, Czech Technical University in Prague, Czech Republic
Antonella Bogoni, Scuola Superiore Sant'Anna-TeCIP Institute, Italy
Eugen Borcoci, University "Politehnica"of Bucharest (UPB), Romania
Christos Bouras, University of Patras, Greece
An Braeken, Vrije Universiteit Brussel, Belgium
Lubomir Brancík, Brno University of Technology, Czech Republic
Peter Brida, University of Zilina, Slovakia
Nicola Calabretta, Eindhoven University of Technology, Netherlands
Vítor Carvalho, 2Ai-EST-IPCA / Algoritmi Research Center - Minho University, Portugal
Fernando Cerdan, Polytechnic University of Cartagena, Spain
Chen Chen, Chongqing University, China
Enrique Chirivella Pérez, Universitat de Valencia, Spain
Tanveer Choudhury, Federation University Australia, Australia
Chi-Wai Chow, National Chiao Tung University, Taiwan
Giampiero Contestabile, Scuola Superiore Sant'Anna, Pisa, Italy
Kevin Daimi, University of Detroit Mercy, USA
Thomas Dreibholz, SimulaMet - Simula Metropolitan Centre for Digital Engineering, Norway
Roman Dunaytsev, Saint-Petersburg State University of Telecommunications, Russia

Copyright Information

For your reference, this is the text governing the copyright release for material published by IARIA.

The copyright release is a transfer of publication rights, which allows IARIA and its partners to drive the dissemination of the published material. This allows IARIA to give articles increased visibility via distribution, inclusion in libraries, and arrangements for submission to indexes.

I, the undersigned, declare that the article is original, and that I represent the authors of this article in the copyright release matters. If this work has been done as work-for-hire, I have obtained all necessary clearances to execute a copyright release. I hereby irrevocably transfer exclusive copyright for this material to IARIA. I give IARIA permission or reproduce the work in any media format such as, but not limited to, print, digital, or electronic. I give IARIA permission to distribute the materials without restriction to any institutions or individuals. I give IARIA permission to submit the work for inclusion in article repositories as IARIA sees fit.

I, the undersigned, declare that to the best of my knowledge, the article is does not contain libelous or otherwise unlawful contents or invading the right of privacy or infringing on a proprietary right.

Following the copyright release, any circulated version of the article must bear the copyright notice and any header and footer information that IARIA applies to the published article.

IARIA grants royalty-free permission to the authors to disseminate the work, under the above provisions, for any academic, commercial, or industrial use. IARIA grants royalty-free permission to any individuals or institutions to make the article available electronically, online, or in print.

IARIA acknowledges that rights to any algorithm, process, procedure, apparatus, or articles of manufacture remain with the authors and their employers.

I, the undersigned, understand that IARIA will not be liable, in contract, tort (including, without limitation, negligence), pre-contract or other representations (other than fraudulent misrepresentations) or otherwise in connection with the publication of my work.

Exception to the above is made for work-for-hire performed while employed by the government. In that case, copyright to the material remains with the said government. The rightful owners (authors and government entity) grant unlimited and unrestricted permission to IARIA, IARIA's contractors, and IARIA's partners to further distribute the work.

Table of Contents

Application of Deep Transfer Learning for Optimal Wireless Beam Selection in a Distributed RAN

Chitwan Arora and Abheek Saha
Hughes Systique Corporation,
Gurgaon, India,
email:*chitwan.arora@hsc.com,abheek.saha@hsc.com*

***Abstract*—This paper continues previous explorations in the area of deep learning applications in the field of cellular wireless networks, specifically the problem of identifying optimal beams in a highly directional urban environment, using topographical data. In our previous work, we have studied the problem and demonstrated how deep-learning can be used on static topographical data for prediction of optimal beams. In this paper, we show a potential architecture for realization of the same for a network of nodes in a given area, taking into account challenges of computational complexity, response time and the inherent architecture of the next generation RAN. This is achieved by using *deep transfer learning* as a way of translating between a global feature space inherent to the coverage area and local variations thereof, specific to the location of each radio-unit.**

Keywords*—*Transfer Learning; Deep Learning; Beam prediction; Distributed/Cloud RAN

I. Introduction

It is well recognized that Deep Learning (DL) is one of the foundational technologies for 5th generation cellular networks, especially in the problem of beam selection and channel estimation in higher frequency bands (mmwave) for urban environments where the radio-environment is highly directional. The problem of urban canyons and shadowing due to buildings is well known [1]. One of the most promising technologies to deal with this problem is the use of machine learning; in this approach, we use Light Detection and Ranging (LIDAR) or Global Positioning System (GPS) maps of a given urban topology to determine the wireless propagation capabilities of the coverage area. It is premised that using deep-learning, we can radically speeden up the process of optimal beam selection for any given User Terminal (UT), if we know its position. To this end, the International Telecommunications Union (ITU) organized a competition in 2020 [2] to explore deep-learning approaches on a multitude of real-world data. The authors participated in this competition and our approach was recognized as achieving 70% accurate prediction of the top-5 beams for a UT in any position in the coverage region. Other competitors showcased solutions, which yielded more than 90% accuracy.

Given that we are already achieving good results using deep learning, it is time to consider the next step of practical deployment of these technologies in the field. It is here that we come up against the biggest engineering challenges. Deep learning algorithms are well known to be prodigious consumers of both computing power and energy; further vast amounts of training data are required to adequately "train" the neural networks (NN). Running a multi-layer neural network in each individual radio unit (RU) for an urban geometry with multiple RHs per sq.km. of coverage area is clearly wasteful (both in terms of computing power as well as energy consumption) and furthermore, very expensive. What is required is to use the combined resources of multiple nodes operating in a common environment, in order to maximally utilize the expensive computing resources in the radio front-end. This is what we shall explore further in this article.

The rest of this paper is organized as follows. In Section II we review the problem in further detail, with a survey of the relevant literature. In Section III, we review the technologies of *transfer learning* and *multiview learning* as modifications introduced in the standard deep-learning methodologies and show how they are relevant to our environment. In Section IV, we present our analysis of the ITU-R dataset and show how it is relevant to the problem at hand. The simulations and corresponding results are work-in-progress and we hope to report our results in a subsequent revision of this paper.

II. Problem Description

In Figure 1, we show the conceptual layout of a 5G cellular network in an urban environment. As we know, the 5G network architecture utilizes the *cloud Radio Access Network (RAN)* concept, where the RAN is disaggregated into the Radio Unit (RU), the Distributed Unit (DU) and the Core Unit (CU). The RUs are placed in diverse locations within the coverage region and are configured to create multiple radio-beams, focusing on specific hotspots. The RUs are connected to a smaller number of DUs, which provide the baseband processing. Finally, the CUs are deployed as a cloud and are designed to provide core signaling and control functions, including the radio-resource management and beam processing functions. ML algorithms can be hosted in various ways within the architecture, most notably within the RAN intelligent controllers (RIC). Some of these schemes have been explored in [3]. There are many possible configurations of this basic architecture, each pertaining to a different use case. A good overview is given in [4].

A. Network Operation

This system works as follows. When a user terminal enters the system, it detects a common signaling channel (low bandwidth, blind detectable) and then signals its position

 ISBN: 978-1-61208-860-0

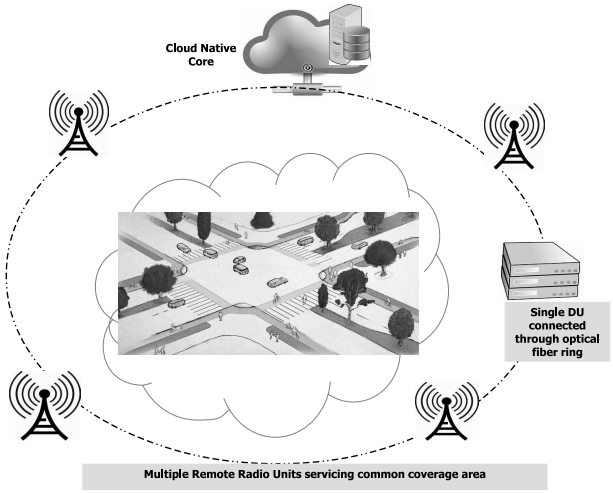

Figure 1. Conceptual View of Distributed RAN covering an urban location

to the network. The network responds to it by identifying a list of predicted top-N beams for it to use. The UT then successively attempts to setup a high-bandwidth data connection with the RU servicing each beam in the list till it achieves success. A beam corresponds to a precoding filter f on the transmitter side and a post-coding vector w on the receiver side. For a given channel matrix $W(p,i)$ corresponding to the channel experienced between the UT at position p and the base-station/RU i the received signal is given by (1).

$$r = \|w^T W(p,i) f\| \tag{1}$$

Obviously, the optimum beam is the one which maximizes the signal strength. We assume a large number of fixed beams, each identified by a tuple of $B \rightarrow \langle b, w, f \rangle$, where b is the beam-id. Each beam is serviced by a given RU (this is invisible to the UT, but important for the beam allocation problem, as we shall see later). The creation and configuration of the individual beams is done externally and available to the network as a database.

Clearly, our algorithm for predicting beams based on UT position has a local (RU specific) as well as a global element to it. Each RU sees an individual view of the environment based on the static topographical features relative to its position, as well as the position of the UT. These static features include high buildings, wide streets, overpasses and other similar features which could potentially either obstruct the signal or provide new reflective paths for it. On the other hand, the system as a whole has to take into account the alignment for all the RUs relative to a given position to determine the optimal beam list.

Matching the tiered nature of the problem, within the network as well, there are tiered layers of control. The *near realtime RAN intelligent controller (rt-RIC)* is typically placed in the DU and the *Non-Realtime RAN Intelligent Controller (nrt-RIC)* is typically placed in the core (Figure 2). The rt-RIC provides closed loop control at very tight latencies, typically focusing on local, high-speed control. The rt-RIC algorithms operate within tight constraints of compute

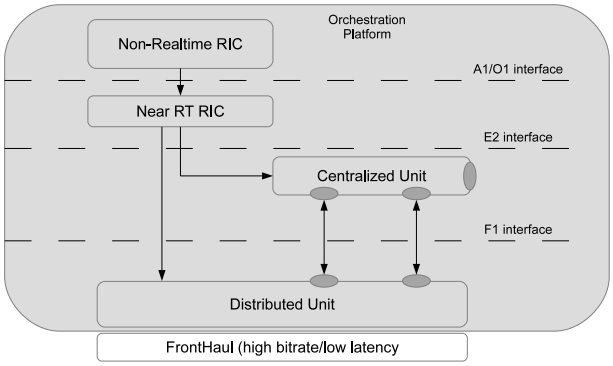

Figure 2. Conceptual View of GnodeB in ORAN

power and latency, in order to fit within the constraints of the DU environment. The nrt-RIC, on the other hand, provides slower control to the DUs using a relatively higher latency link. It has substantially larger compute and memory resources at its disposal, and can afford to take a global view of the network, due to its ability to store and process data from multiple DUs and RUs. This will subsequently play a role in the actual deployment of our ML based beam prediction solution, as we shall discuss in Section IV.

We now consider the ML algorithm. The input to the ML is the topographical information about the coverage region and labelled data corresponding to specific locations within the region and the beam/RU to which it maps. The format of the topographical data can take many forms, such as LIDAR scans [5] from the perspective of individual positions within the coverage area, with the reflections identifying local obstacles, along with GPS topographical data and images taken by wide-angle cameras. In other literature, topographical data is in the form of 3-d maps (for example, as provided by OpenStreetMaps) or in the form of GPS contour data [6][7]. The labelled data comprises of actual measurements from specific UTs at specific positions identifying the UT location and the empirically measured optimal beam id (or top N beams). This will be used to train the DL model.

The problem thus can be summarized as follows. Assuming that we have topographical information for the network coverage area, how do we build an RU specific view, as well as a global view of the propagation characteristics, and subsequently map this to optimal beam positions.

B. Literature Survey

There is a lot of recent literature in beam identification for mmwave communication. In [5], the problem is presented from the perspective of the UT attempting to compute the optimal beam list, based on LIDAR data. In [8], the authors present the problem in a vehicular perspective, using realtime LIDAR measurements to fingerprint a position relative to other vehicles in a given highway. In [9], the authors present a network oriented approach using coordinated beams and a centralized deep-learning model, similar to the problem we are addressing. However, the authors use directly measured signal strengths as the input. Each BS individually *learns* the system and the coordination is purely on the basis of

 ISBN: 978-1-61208-860-0

selection, not in the model itself. In [10], the authors focus on the beam sweeping pattern itself as the output to the ML, as opposed to the beam prediction itself.

For our particular problem, we shall use the technology of *transfer learning* (TL). The area of transfer learning is an active field in DL theory; comprehensive surveys are given in [11][12]. The success of transfer learning is predicated on the ability to extract features in the preliminary part of the DL model; this problem is surveyed in [13][14]. The authors in [15][16] analyze the transferability of the extracted features, by selectively migrating some layers of a pre-trained DL model and comparing it to the performance of the same with randomized starting weights. This is extended in [17] into a concept of a *Joint Adaptation Network*, which will be used in the rest of our paper.

The problem of *multiview learning* is also an area of active research; see the surveys in [18]-[20]. The advantage of multi-view learning is that it enables significant simplification of the input data to be processed at individual nodes, by using commonality to remove redundancies and noise. Multi-view learning seems to be peculiarly applicable to a network node scenario as we have presented in Section II. However, there doesn't seem to be much published research in this domain.

III. Adaptations of Deep Learning to a distributed/hierarchical environment

If we analyze our problem from the TL angle, we see that we have a large number of independently operating nodes, each of which has to learn variations of the same data, i.e., the topography of the coverage region independently. It has been pointed out that we can make substantial savings by coordinating the learning procedure in some way. The two major technologies that we have considered are *transfer learning* and *multiview learning*, which are summarized in the following Subsections.

A. Transfer Learning

TL is a method whereby the information acquired by particular DL model can be *transferred* in suitably adapted form to another DL model. The transfer can be cross-domain or (as in our case), intra-domain. In our particular situation, we can have a central system which *learns* about the topology by processing all the path specific data available to the system and then transfers the learned model to individual RUs for their use. To implement the transfer scheme, we need to decide two things. First is what exactly to transfer and the second is how to accomplish it.

While there are many variants of transfer learning, one of the most appealing is that of *feature based* transfer learning. In this mode, the *features* of the data are extracted and learnt by the main ML and then transferred to the subsequent MLs; these MLs take this feature knowledge and further refine it. Features are fairly intuitive (especially when geographical data is involved) and it is possible to extract them efficiently from raw data. In our case, a feature could be a large building or other artefact that significantly impacts the propagation characterestics within the environment. It is well known that a DL based learning engine learns features in all its layers, starting with the most generic and moving towards the more specific; the problem then becomes selecting the layer within which the features are learnt at the optimal level of specificity. A second problem is the applicability of the features and how to use them in the target inference engine. In our particular environment, it is not just a matter of weighting the feature set, but rather of determining the applicability of a feature and its impact on the inference problem as a whole.

B. Multiview Learning

When we have multiple data sets from a single common environment (for example, RSSI readings for different UT positions from the perspective of multiple base-stations/RUs), a primary problem is the risk of over-fitting, especially if the data is simply concatenated together and fed into a single DL engine. This is the problem that multiview learning tries to avoid. On the other hand, simply separating out the data and treating them completely independent data-sets leads to insufficient training, especially if individual data-sets are small, or uneven. There are many different ways to implement multi-view training, each of which focusses on a different aspect of the problem. Co-training looks at maximizing the agreement between different views, whereas multi-kernel learning and subspace learning operate by implementing a certain structure on the underlying data-space.

IV. Architecture for Deep Adaptation Learning for the RAN beam selection problem

We now come to the realization of the beam-selection algorithm. In our earlier work [21], we described a generic realization as a single centralized inference engine as a Deep Neural Network (DNN) of 11 layers, using UT position as the index, in conjunction with the angles of arrival and departure and signal strength as labels to match optimal beams with UTs in other, unlabelled positions within the coverage area. As shown in the ITU-R challenge referenced above, it is possible to augment the data set with other parametric information. For example, LIDAR/image data is highly perspectival; by providing LIDAR based ranging data from individual BS locations, we can augment the empirical wireless information and get better training of individual inference engines.

In Figure 3, we show conceptually how the beam selection algorithm works. The algorithm is broken up into two tiers. The central algorithm learns the common features of the urban environment and transfers the DNN with pre-trained layers to the RU specific tier. This tier then augments the DNN with local data and computes the final inference engine. For global data, we use the GPS data indexed by position with labelled information about UTs which were able to acquire beams (with associated signal quality). Based on this, we can form a top level view of the predicted coverage for beams which is learned by the engine. In the local tier, we augment this information by using signal strength

 ISBN: 978-1-61208-860-0

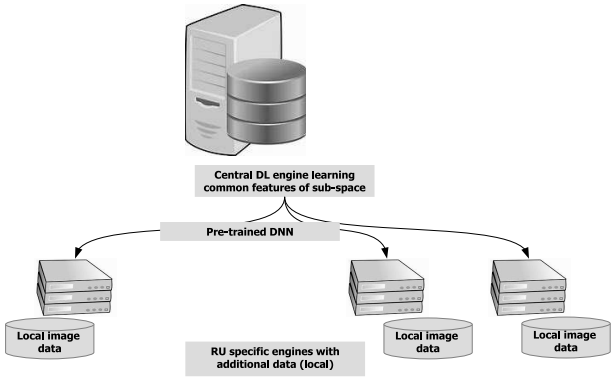

Figure 3. Hierarchical implementation of beam prediction DL engine

measurements (and LOS/NLOS computation) for individual UTs with respect to the position of the associated RU. This allows the RUs to create shortlists of predicted beams, which are then consolidated to form an overall list for advertising to the UTs. To improve the performance of the DNN at the RU, we can augment the central model by using local data specific to the RU. In our case, we use images of the horizon from the RU position. These images can highlight the presence of tall buildings or other obstructions in the surrounding area, which can be utilized to predict the possibility of LOS paths from different UT positions. Using self-supervised auto-encoders, we can identify the key feature-sets of each image and then match the endoded version to beam directions. By adding this information to the feature level data derived from the top level model, we hope to build accurate, but computationally simple local DNNs, which can be implemented relatively cheaply at the RU.

V. Conclusions

We have taken the baseline of the ITU-R data-set as described in [5] as the starting point as one of the few available empirical data-sets available in the field of wireless. The data-set provides GPS, LIDAR and imagery based data. As described above, we must start with the GPS based data as the global data-base. Primary analysis at the global level will be targetted at learning the features of the data-set. Once we have a good understanding of where these features are captured, we will consider the problem of moving the pre-trained DNNs to the RU and adding image data analysis to the same. This shall be explored in the final version of this article.

References

[1] M. K. Samimi and T. S. Rappaport, "3-d millimeter-wave statistical channel model for 5g wireless system design," *IEEE Transactions on Microwave Theory and Techniques*, vol. 64, no. 7, pp. 2207–2225, 2016.

[2] "ITU Artificial Intelligence/Machine Learning in 5G Challenge," https://www.itu.int/en/ITU-T/AI/challenge/2020/Pages/default.aspx last accessed March,2021, 2020.

[3] H. Lee, J. Cha, D. Kwon, M. Jeong, and I. Park, "Hosting AI/ML workflows on O-RAN RIC platform," in *2020 IEEE Globecom Workshops (GC Wkshps)*, 2020, pp. 1–6.

[4] S. K. Singh, R. Singh, and B. Kumbhani, "The evolution of radio access network towards open-ran: Challenges and opportunities," in *2020 IEEE Wireless Communications and Networking Conference Workshops (WCNCW)*, 2020, pp. 1–6.

[5] A. Klautau, N. Gonzalez-Prelcic, and R. W. Heath, "LIDAR Data for Deep Learning based mmwave Beam-Selection," *IEEE Wireless Communications Letters*, vol. 8, no. 3, pp. 909–912, 2019.

[6] A. Klautau, P. Batista, N. Gonzalez-Prelcic, Y. Wang, and R. W. H. Jr, "5G MIMO Data for Machine Learning: Application to Beam-Selection using Deep Learning," in *Proc of the information theory and application workshop, February*, 2018, pp. 1–9.

[7] M. Y. Takeda, A. Klautau, A. Mezghani, and R. W. Heath, "MIMO Channel Estimation with Non-Ideal ADCs: Deep Learning versus GAMP," in *2019 IEEE 29th International Workshop on Machine Learning for Signal Processing (MLSP)*, 2019, pp. 1–6.

[8] Y. Wang, M. Narasimha, and R. W. Heath, "Mmwave beam prediction with situational awareness: A machine learning approach," in *2018 IEEE 19th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, 2018, pp. 1–5.

[9] A. Alkhateeb, S. Alex, P. Varkey, Y. Li, Q. Qu, and D. Tujkovic, "Deep learning coordinated beamforming for highly-mobile millimeter wave systems,," *IEEE Access, , –37 348, June*, vol. 6, pp. 37–328, 2018.

[10] A. Mazin, M. Elkourdi, and R. D. Gitlin, "Accelerating Beam Sweeping in mmwave Standalone 5G New Radios using Recurrent Neural Networks," in *2018 IEEE 88th Vehicular Technology Conference (VTC-Fall)*, 2018, pp. 1–4.

[11] F. Zhuang *et al.*, "A Comprehensive Survey on Transfer Learning," *Proceedings of the IEEE*, vol. 109, no. 1, pp. 43–76, 2021.

[12] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, and C. Liu, "A survey on Deep Transfer Learning," in *International conference on artificial neural networks*. Springer, 2018, pp. 270–279.

[13] S. Dara and P. Tumma, "Feature Extraction by using Deep Learning: A Survey," in *2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA)*, 2018, pp. 1795–1801.

[14] S. Khalid, T. Khalil, and S. Nasreen, "A survey of feature selection and feature extraction techniques in Machine Learning," in *2014 Science and Information Conference*, 2014, pp. 372–378.

[15] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in Deep Neural Networks?" *arXiv preprint*, 2014. Online:https://arxiv.org/abs/1411.1792 last accessed March,2021

[16] M. Long, Y. Cao, J. Wang, and M. Jordan, "Learning Transferable Features with Deep Adaptation Networks," in *Proceedings of the 32nd International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, F. Bach and D. Blei, Eds., vol. 37. Lille, France: PMLR, 07–09 Jul 2015, pp. 97–105.

[17] M. Long, H. Zhu, J. Wang, and M. I. Jordan, "Deep Transfer Learning with Joint Adaptation Networks," in *Proceedings of the 34th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, D. Precup and Y. W. Teh, Eds., vol. 70. International Convention Centre, Sydney, Australia: PMLR, 06–11 Aug 2017, pp. 2208–2217.

[18] C. Xu, D. Tao, and C. Xu, "A survey on multi-view learning," *arXiv preprint*, 2013. Online:https://arxiv.org/abs/1304.5634, last accessed: March,2021

[19] J. Zhao, X. Xie, X. Xu, and S. Sun, "Multi-view learning overview: Recent progress and new challenges," *Information Fusion*, vol. 38, pp. 43–54, 2017.

[20] S. Sun, "A survey of multi-view Machine Learning," *Neural computing and applications*, vol. 23, no. 7, pp. 2031–2038, 2013.

[21] C. Arora and A. Saha, "AI based beam management for 5G (mmwave) at wireless Edge," *Advances in Telecommunication, International Journal on*, vol. 13, no. 1&2, pp. 1 – 9, 2020.

 ISBN: 978-1-61208-860-0

Performance Evaluation of MIMO Detectors over Impulsive Noise

Danilo Pena*, José Juajinoy†, Thais Areias‡, Juliano Bazzo§,

Sidia Institute of Science and Technology, Manaus, Brazil
E-mail: *danilo.pena@sidia.com, †jose.juajinoy@sidia.com, ‡thais.areias@sidia.com, §juliano.bazzo@sidia.com,

***Abstract*—Multiple-Input and Multiple-Output (MIMO) is expected to be one of the most crucial technologies towards the 5G mobile communication systems and beyond. The understanding of the performance and limits of MIMO detectors is essential in order to transmit signals at high rates and with high reliability. In this paper, we present an evaluation of MIMO detectors over non-gaussian impulsive noise. The traditional MIMO detectors are designed assuming noise modeled as gaussian second-order statistics. However, many works have presented non-gaussian impulsive noise in different MIMO scenarios degrading the detector performance. Also, we investigate an alternative to symmetric α-stable distribution to model impulsive noise called the gaussian mixture model. The simulation results show that the Symbol Error Rate (SER) performance depends on not only the quality of the signal but also the impulsiveness level of the noise.**

***Keywords*—Impulsive noise, non-gaussian model, alpha-stable distribution, GMM.**

I. Introduction

Multiple-Input and Multiple-Output (MIMO) technology has been receiving considerable attention recently from the wireless communication field. Nowadays, wireless systems are demanding higher data rates with reliability, being efficient in terms of bandwidth. In this context, MIMO plays a key role in achieving highly efficient spectrum usage with a relatively small number of antennas involving large amounts of data. Thus, MIMO techniques have been investigated by researchers and engineers in several contexts, such as 4G and 5G networks, distributed antennas, heterogeneous network, IEEE 802.11ac and millimeter-wave impacts due to its high frequency [1].

Authors argue that the performance of wireless communication systems is mainly governed by wireless channel characteristics [2]. Measurement and environmental conditions, such as multipath and noise create additional difficulty within already existing detection challenges faced by MIMO systems. Especially for classical MIMO detectors, which rely on second-order statistical noise assumptions, they may suffer severe impact via meaningful degradation in non-gaussian scenarios [3]. Thus, one way to improve the reliability of MIMO systems is by analyzing undesirable effects of channel and noise, thereby evaluating MIMO detectors while considering realistic models. Notably, characteristics of impulsive noise have been modelled accurately by non-gaussian processes [4], demonstrating better fitting than gaussian model in several scenarios due to man-made and electromagnetic interference noises. Also, studies have investigated the presence of non-gaussian noise components in millimeter wave scenarios [5]–[7] at high frequencies.

Several statistical models have been proposed to describe non-gaussian impulsive noise. In particular, stable distributions is one of the most used ones for this purpose [4]. They offer more freedom degrees than the gaussian model by adjusting free distribution parameters, allowing us to describe how impulsive the noise is. This model has been explored in many different communication scenarios, such as acoustic channels [8], wireless communication solutions [9], and satellite communications [10]. Moreover, the α-stable model presents relevant properties for noise modelling such as generalized central limit, stability property, and heavy tails [4].

Additionally, many approaches have been studied modelling impulsive noise by Gaussian Mixture Models (GMM) [8], [11]. They claim that the GMM is capable of representing heavy tailed impulsive noise by an arbitrary additive, independent and identically distributed (i.i.d.), symmetric, non-gaussian GMM noise. Moreover, the expectation-maximization (EM) algorithm for estimating the distribution parameters is a well-known tool based on maximum likelihood. Thus, we purpose the GMM as a beneficial complementary alternative to α-stable distribution to model noise in MIMO systems.

MIMO detectors based on exhaustive searching and channel estimation have been proposed with high performance if compared to traditional detectors in non-gaussian environments [3]. However, those detectors usually have too high computational complexity, making them infeasible in practical scenarios. On the other hand, the classical detectors have unknown performance in non-gaussian noise environments depending on the impulsiveness level. Therefore, the comprehension of the relationship between impulsiveness levels and the performance of detectors is crucial in the making decisions about the choice of methods.

In this article, we examine the performance of MIMO detectors in non-gaussian impulsive noise, highlighting the noise models in such technology and Monte Carlo analysis for relevant distribution parameters. This study also describes an alternative to model impulsive noise, called the gaussian mixture model, and its impact for MIMO detector evaluation. This work uses the Rayleigh fading model, which may represent realistic narrowband mmWave systems [12].

This paper is organized as follows. In Section II, we describe the MIMO system, presenting the channel and noise model. MIMO detectors are presented in Section III. In Section IV, the main results are presented and discussed, comparing the performance of the tradition MIMO detectors in non-gaussian scenarios by simulations. In Section V, we present our final

 ISBN: 978-1-61208-860-0

remarks.

II. MIMO System

Consider a MIMO digital system with N_R antennas at the receiver and N_T antennas at the transmitter. The N_R antennas are spaced, such that the received signals may be considered independent of each other. The k-th symbol received by the m-th antennas is given by:

$$y_m(t) = \sum_{n=1}^{N_T} s_n(t) h_{mn}(t) p(t) + w_m(t), \quad (1)$$

where $s_n(t)$ represents the transmitted symbol from the n-th antenna, originated from a modulation scheme, $h_{mn}(t)$ represents the channel model between the n-th transmitting antenna and m-th receiving antenna, $w_m(t)$ corresponds to the channel noise, and $p(t)$ is a rectangular pulse.

We assume the time-domain channel model coefficients $h_m(t)$ as a Rayleigh distribution, being defined by

$$h_{mn}(t) = h_{mn,r}(t) + j h_{mn,q}(t), \quad (2)$$

where $h_{mn,r}(t)$ and $h_{mn,q}(t)$ are gaussian processes with mean zero and variance equal to $1/2$. We also assume that the differences in propagation times of the signals from the transmitters to the receivers are small relative to the symbol duration.

III. Impulsive Noise Model

The α-stable and gaussian mixture model are the most frequently used distributions to model impulsive noise. Those models have different characteristics presented in this section.

A. Symmetric α-Stable Model

Reasons for statistical modelling using α-stable distributions are based on crucial properties, such as generalized central limit theorem and stability. According to the generalized central limit, if the sum of independent and identically distributed random variables with or without finite variance converge, then the limit distribution must be α-stable. Another relevant property states that the sum of two independent random variables with the same characteristic exponent (α value) is also α-stable, known as stability property. Finally, we consider that the signal exhibits heavy tails and skewness, which is well represented by α-stable model.

There are different parametrizations of α-stable distribution of the characteristic function. We assume the parameters $\boldsymbol{\theta}_\alpha = (\alpha, \beta, \gamma, \delta)$ and the following characteristic function [4]:

$$\varphi(\omega; \boldsymbol{\theta}_\alpha) = \exp(-\gamma^\alpha |\omega|^\alpha [1 - j\Theta(\omega; \alpha, \beta)] + j\delta\omega), \quad (3)$$

with

$$\Theta = \begin{cases} \beta(\tan\frac{\pi\alpha}{2})(\text{sign}\,\omega), & \alpha \neq 1 \\ -\beta\frac{2}{\pi}(\ln|\omega|), & \alpha = 1, \end{cases} \quad (4)$$

where

α is the characteristic exponent such that $0 < \alpha < 2$,
β is the symmetry parameter such that $-1 \leq \beta \leq 1$,
γ is the dispersion parameter such that $\gamma > 0$,
δ is the location parameter such that $-\infty < \delta < \infty$.

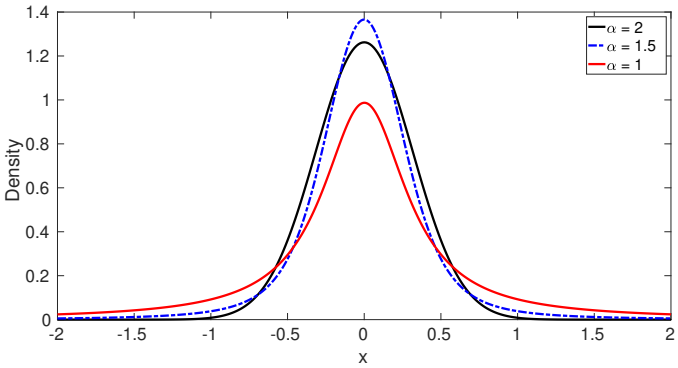

Figure 1. Probability distribution function of symmetrical α-stable with $\beta = \delta = 0$ and $\gamma = 1$.

We also assume a symmetric α-stable (SαS) class because it has proved to be very useful in modelling impulsive noise [13]. For such distribution class, β = 0 and δ = 0 [14]. Figure 1 shows the α value variation representing the impulsiveness level of the distribution, where a low value of α suggests high impulsiveness and a non-gaussian behavior, and a high value of α means that the distribution is close to the gaussian behavior, which $\alpha = 2$ is the gaussian case.

B. Gaussian Mixture Model

The GMM is a linear combination of gaussian functions where the sum of all weight coefficients is equal to one. Thus, a random variable y with GMM distribution is defined by its probability density function as

$$p(y) = \sum_{i=1}^{M} c_i N(x_i|\mu_i, \sigma_i), \text{ with } \sum_{i}^{M} c_i = 1, \quad (5)$$

where c_i is the weight of the i-th Gaussian distribution function, M represents the number of Gaussian distributions in the mixture, and $\mathcal{N}(x_i|\mu_i, \sigma_i)$ is a Gaussian distribution function given by

$$N(x_i|\mu_i, \sigma_i) = \frac{1}{\sqrt{2\pi}\sigma_i} e^{-\frac{(x_i-\mu_i)}{2\sigma_i^2}}, \quad (6)$$

where μ_i and σ_i represents mean and variance, respectively, of the i-th Gaussian. Figure 2 illustrates the gaussian mixture model representing the impulsive noise, which results in a heavy tail distribution.

IV. MIMO Detectors

We consider three different detectors based on frequency nonselective MIMO channel and Rayleigh fading. Those methods are designed for recovering the data symbols with additive gaussian noise assumptions. However, in practical scenarios, those assumptions can mislead the real performance of MIMO systems making them unfeasible depending on channel estimation.

 ISBN: 978-1-61208-860-0

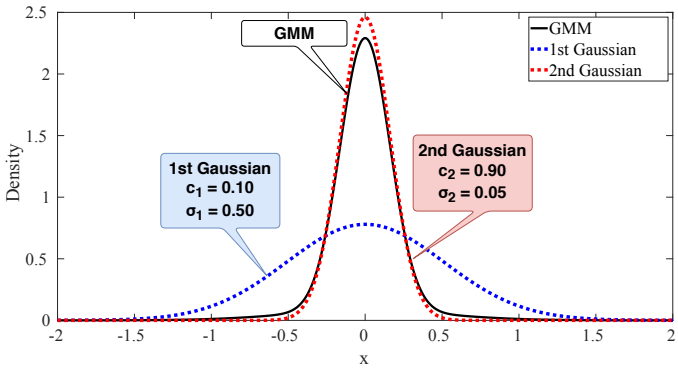

Figure 2. Probability distribution function of gaussian mixture model with two gaussians with parameters $\mu_1 = \mu_2 = 0$, $\sigma_1 = 10$, and $\sigma_2 = 1$.

A. Maximum Likelihood Detector

The maximum likelihood detection is optimum in terms of performance assuming gaussian noise model. This detector minimizes the average error probability by finding the minimum Euclidean distance. This technique requires high computational complexity due to the searching algorithm.

$$\hat{\boldsymbol{s}}_{\mathrm{MLD}} = \arg\min_{\boldsymbol{s}} \left| y_m - \sum_{n=1}^{N_T} h_{mn} s_n \right|^2, \tag{7}$$

where s_n is a symbol among a set of possible constellation symbols used in the transmission.

B. Minimum Mean-Square-Error Detector

The MMSE detector estimates the transmitted symbols based on the linear combination of the received signals. The linear combination is given by

$$\hat{\boldsymbol{s}}_{\mathrm{MMSE}} = \boldsymbol{W}^H \boldsymbol{y}_m, \tag{8}$$

where $\boldsymbol{W}$ is a weighting matrix. In order to minimize the mean square error, the weighting matrix is represented by

$$J(\boldsymbol{W}) = E\left[||\boldsymbol{s}_{\mathrm{MMSE}} - \boldsymbol{W}^H \boldsymbol{y}_m||^2\right]. \tag{9}$$

The weight vectors inside the matrix can be obtained by

$$\boldsymbol{w}_n = \boldsymbol{R}^{-1} \boldsymbol{r}_{s_n y}, \tag{10}$$

where $\boldsymbol{R}$ is the autocorrelation matrix of the received signal y_m, and $\boldsymbol{r}_{s_n y} = E[s_n \boldsymbol{y}_m]$.

C. Inverse Channel Detector

The ICD detector is similar to MMSE, where the estimation is designed using a linear combination of the received signal. However, in ICD the interchannel interference is eliminated due to the weighting matrix with $N_R = N_T$. Therefore, the estimation is given by

$$\hat{\boldsymbol{s}}_{\mathrm{ICD}} = \boldsymbol{H}^{-1} \boldsymbol{y}_m, \tag{11}$$

where $\boldsymbol{W}^H = \boldsymbol{H}^{-1}$.

V. Results

This section presents computer simulation results for the performance evaluation of MIMO detectors. We examined the Symbol Error Rates (SER) for different levels of impulsiveness and quality of signal considering 2x2 MIMO systems. The simulations assessed the error rate performance based on the Monte Carlo method where each point of the SER curves employed at least 50 errors in the estimation. All simulations were performed considering baseband with BPSK modulated signal and unity energy, being the antennas statistically independent of one each other. In addition, Rayleigh flat fading was assumed as the multipath propagation model in the wireless channel.

The SER metric is usually computed versus the signal-to-noise ratio (SNR). However, the infinite variance of non-Gaussian SαS processes prevents to compute the signal-to-noise ratio as a measurement of signal quality. In this work, we use the geometric signal-to-noise ratio (GSNR) [R] instead of the SNR. The GSNR is given by

$$\mathrm{GSNR} = \frac{1}{2C_g} \left(\frac{A}{S_0} \right)^2, \tag{12}$$

where the normalization constant $C_g = e^{C_e} \approx 1.78$ is the exponential of the Euler constant (C_e), used to ensure that GSNR corresponds to SNR when the channel is Gaussian ($\alpha = 2$); S_0 is the geometric power of a SαS random variable; and A is the root-mean-square value of the signal.

For the GMM, we use two gaussians, i.e., $M = 2$, where one gaussian has much higher variance than the other one in order to represent the impulsiveness of the noise. Thus, the variances are given by

$$\sigma_1^2 = \xi \cdot \sigma_2^2. \tag{13}$$

where σ_i are the variances of the i-th gaussian, and ξ is the relationship between them, describing how different they are. We assume that the first variance has higher value than the second one, i.e., $\xi > 1$, and their occurrences are described by $c_1 = 0.1$ and $c_2 = 0.9$. In this case, the total variance is given by the weighted sum of the variances as

$$\sigma_T = c1 \cdot \sigma_1 + c2 \cdot \sigma_2. \tag{14}$$

A. Noise Model Analysis

First, we show in Figure 3 the MIMO performance in the gaussian scenario as a reference scenario indicating no presence of impulsiveness. This behavior of the detectors is expected in environments where the gaussian model describes well the noise model.

Figure 4 presents the MLD, MMSE, and ICD detectors over SαS noise with parameter $\alpha = 1.9$, a low impulsive noise scenario. The SER of detectors are clearly higher than in the gaussian case, since the impulsiveness degrades them. However, the ML detector has low SER values at high GSNR.

 ISBN: 978-1-61208-860-0

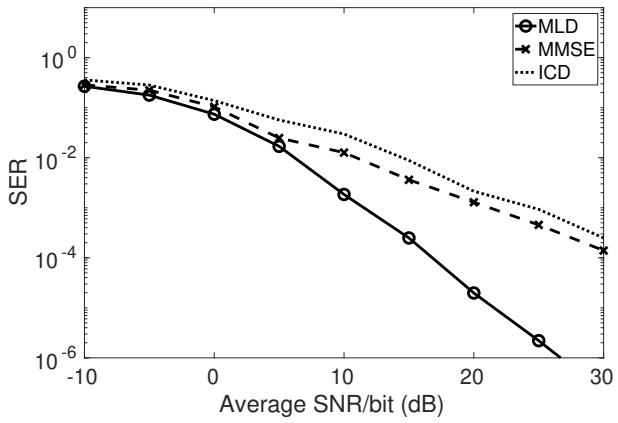

Figure 3. MIMO 2x2 over gaussian noise.

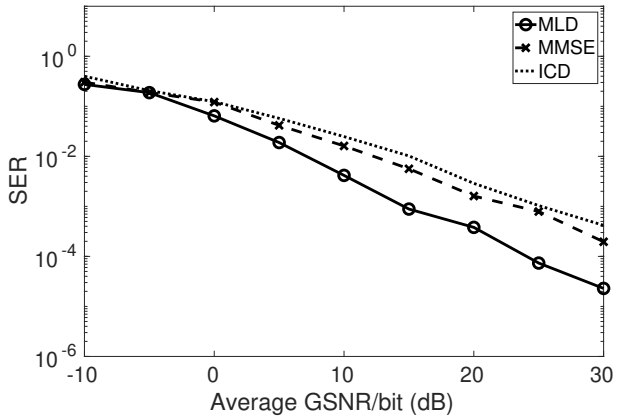

Figure 4. MIMO 2x2 over alpha-stable noise with $\alpha = 1.9$.

Figure 5 shows the detectors over SαS noise with parameter $\alpha = 1.3$. This scenario represents a severe impulsive noise where all detectors are degraded. We also visualize that MLD has a similar performance to other detectors in this scenario even for high GSNR values.

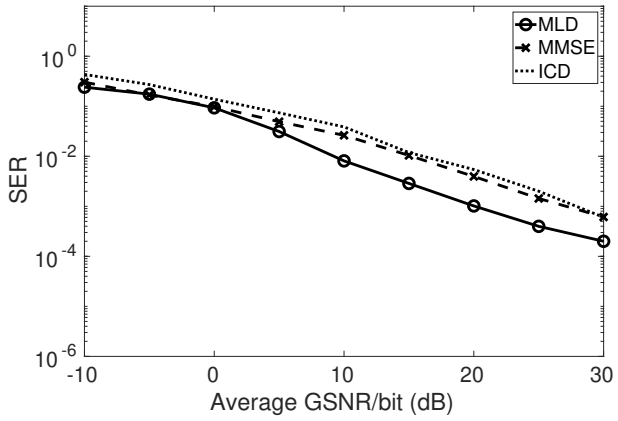

Figure 5. MIMO 2x2 over alpha-stable noise with $\alpha = 1.3$.

Figures 6 and 7 present the same detectors over GMM noise with two gaussians and means equal to zero. They have different variances, which one represents an usual class of noise with weighting of $c_1 = 0.1$, and another one represents the impulsive component with higher variance and weighting of $c_2 = 0.9$. In this scenario, the impulsiveness level is given by the relation between the variances σ_1 and σ_2. Figure 6 presents the detectors over GMM with low impulsiveness level, given by $\xi = 2$.

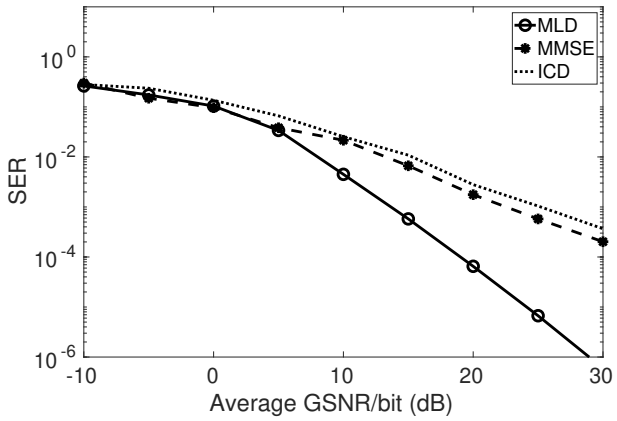

Figure 6. MIMO 2x2 over GMM noise with $\xi = 2$.

Figure 7 shows the performance of detectors over GMM with impulsiveness level given by $\xi = 10$. In this scenario, the detectors have higher SER if compared to the scenario with $\xi = 2$ due to the high impulsiveness level.

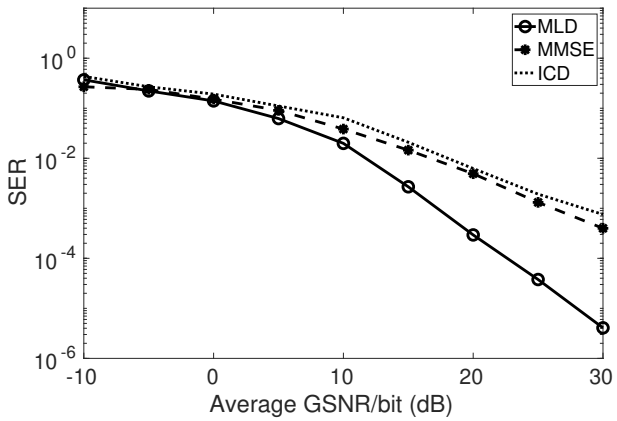

Figure 7. MIMO 2x2 over GMM noise with $\xi = 10$.

B. Impulsiveness Analysis

A crucial analysis of detectors over impulsive noise is the impulsiveness level. As the detectors are operating not respecting the gaussian assumption, then we can not affirm the exactly behavior of the system. However, we expect that less impulsive is the noise better is the performance of the detectors. So, we evaluate all detectors over the two models,

SαS and GMM, evaluating their impulsiveness level. Each model has a different parameter associated, being α for SαS and ξ for GMM. We adopt a constant GSNR for each noise model and compute the SER versus different values of α and ξ.

Figure 8 presents the SER of MIMO detectors for GSNR of 10 and values of α from 1.1 (more impulsive) to 2 (gaussian case). In high impulsiveness scenario, i.e., low value of α, higher is the SER of the detectors, as expected. However, the MLD is more sensitive to the impulsiveness level than the other detectors. In addition, we can affirm that the SαS model may represent higher impulsiveness level than the GMM, in terms of the detectors performance.

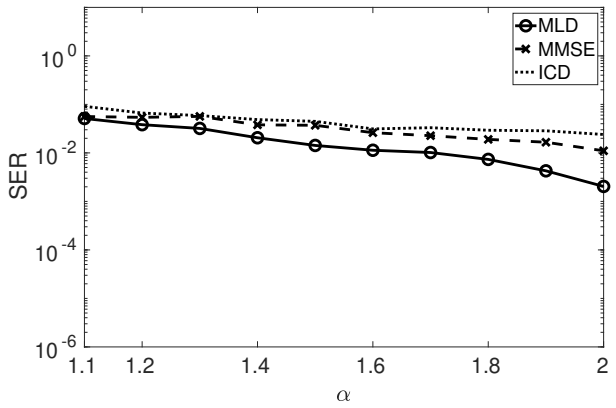

Figure 8. MIMO 2x2 over SαS with different impulsiveness levels.

Figure 9 shows the SER of MIMO detectors for GSNR of 10 and values of ξ from 2 to 20, representing the relation between the variances σ_1 and σ_2. In high impulsiveness scenario, all detectors have higher SER performance, where they are more sensitive for ξ values from 2 to 10. Also, we can note that the detectors performance degrade smoother over GMM than over SαS model.

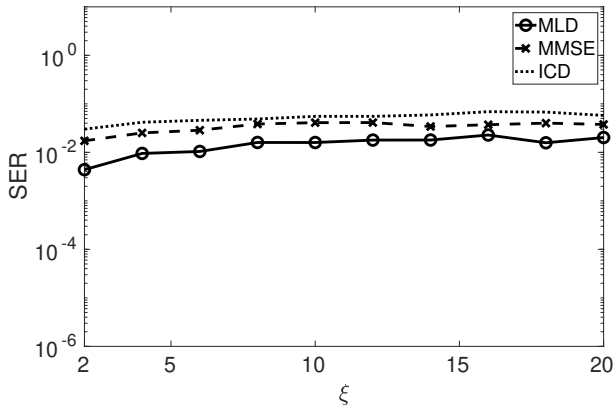

Figure 9. MIMO 2x2 over GMM with different impulsiveness levels.

VI. Conclusion

In this paper, we evaluated traditional MIMO detectors over non-gaussian scenarios for different impulsiveness levels. Indeed, the traditional MIMO detectors have high error rates in impulsive noise scenarios making them infeasible for current wireless systems. On the other hand, depending on the noise power (GSNR), the detectors work well for impulsiveness levels that are not severe. Also, depending on the model used, the detectors can be more sensitive in relation to the impulsiveness level represented by their parameters. Therefore, studies in impulsive noise scenarios must pay attention for not only the GSNR value, but also for the impulsiveness level considered and how it impacts the detectors.

Future works may investigate the Gaussian mixture model including the number of Gaussian components and its effect in impulsive noise fitting. Also, future studies may use these results to produce adaptive detectors based on impulsiveness parameters, reaching better performance than the traditional detectors.

Acknowledgment

This work was partially supported by Samsung Eletronica da Amazonia Ltda., under the auspice of the informatic law no 8.387/91.

References

[1] K. Maruta and F. Falcone, "Massive MIMO Systems: Present and Future," *Electronics*, vol. 9, no. 3, p. 385, feb 2020.

[2] Y. S. Cho, J. Kim, W. Y. Yang, and C. G. Kang, *MIMO-OFDM Wireless Communications with MATLAB®*. Chichester, UK: John Wiley & Sons, Ltd, aug 2010.

[3] P. T. V. De Souza, A. I. R. Fontes, V. S. V. De Souza, and L. F. Silveira, "A Novel Signal Detector in MIMO Systems Based on Complex Correntropy," *IEEE Access*, vol. 7, pp. 137 517–137 527, 2019.

[4] M. Shao and C. L. C. Nikias, "Signal Processing with Fractional Lower Order Moments: Stable Processes and Their Applications," *Proceedings of the IEEE*, vol. 81, no. 7, pp. 986–1010, jul 1993.

[5] L. Shhab, A. Rizaner, A. H. Ulusoy, and H. Amca, "Suppressing the Effect of Impulsive Noise on Millimeter-Wave Communications Systems," *Radioengineering*, vol. 29, no. 2, pp. 376–385, jun 2020.

[6] H. Amca, "The Presence of Thermal, Shot, Impulsive and Flicker Noise at Millimeter-Wave Frequencies," in *2019 International Symposium on Networks, Computers and Communications (ISNCC)*. IEEE, jun 2019, pp. 1–6.

[7] O. S. Badarneh and F. S. Almehmadi, "The effects of different noise types and mobility on error rate of digital modulation schemes over millimeter-wave Weibull fading channels," *Wireless Networks*, vol. 25, no. 5, pp. 2259–2268, jul 2019.

[8] D. Pena and et al., "Acoustic impulsive noise based on non-Gaussian models: An experimental evaluation," *Sensors (Switzerland)*, vol. 19, no. 12, p. 2827, jun 2019.

[9] H. Hamlili, S. Kameche, and A. Abdelmalek, "Convolutional Code Performance for OFDM System in an Alpha-Stable Noise Environment," in *2018 International Conference on Communications and Electrical Engineering (ICCEE)*. IEEE, dec 2018, pp. 1–5.

[10] M. D. Button, J. G. Gardiner, and I. A. Glover, "Measurement of the impulsive noise environment for satellite-mobile radio systems at 1.5 GHz," *IEEE Transactions on Vehicular Technology*, pp. 551–560, 2002.

[11] R. J. Kozick and B. M. Sadler, "Maximum-likelihood array processing in non-Gaussian noise with Gaussian mixtures," *IEEE Transactions on Signal Processing*, vol. 48, no. 12, pp. 3520–3535, 2000.

[12] N. Iqbal and et al., "Multipath Cluster Fading Statistics and Modeling in Millimeter-Wave Radio Channels," *IEEE Transactions on Antennas and Propagation*, vol. 67, no. 4, pp. 2622–2632, apr 2019.

 ISBN: 978-1-61208-860-0

[13] C. L. Nikias and M. Shao, *Signal Processing with Alpha-stable Distributions and Applications*. New York, NY, USA: Wiley-Interscience, 1995.

[14] G. Samorodnitsky and M. S. Taqqu, *Stable Non-Gaussian Random Processes: Stochastic Models with Infinite Variance*, ser. Stochastic Modeling Series. Taylor & Francis, 1994.

 ISBN: 978-1-61208-860-0

Optimization of Cloud Model Based on Shifted N-policy M/M/m/K Queue

Zsolt Saffer
Institute of Statistics and Mathematical Methods in Economics
Vienna University of Technology
Vienna, Austria
Email: zsolt.saffer@tuwien.ac.at

***Abstract*—In this paper, we present the performance analysis and cost optimization of an Infrastructure-as-a-Service (IaaS) cloud model with a capacity control policy. The Virtual Machines (VM) are modeled as parallel resources, which can be either in active or in standby state. The capacity of the cloud is controlled by changing the number of active VMs. We define a cost model, that the cloud provider encounters. It takes into account both energy consumption and performance measures. The major objective of the work is to provide a tractable analytic model, which is suitable for practical use. For this purpose, we model the cloud services by an $M/M/m/K$ queue. We propose a simple control policy, in which a predefined portion of VMs are always active. The remaining ones are activated simultaneously when the number of requests reaches a threshold and deactivated when the number of requests falls below the predefined portion of active VMs. We call it as shifted N-policy. We provide the stationary analysis of the model. We derive closed form results for the distribution of the number of requests and for several performance measures. The cost model leads to a discrete optimization task, which we approximate by a nonlinear continuous optimization task. After applying numerous approximations, we reduce the problem to a nonlinear equation with a specific structure including factorial terms. We provide the approximate solution of the optimization task. The major result of the work is the closed form approximate solution formula, which gives the optimal threshold under the most relevant range of parameters. The formula gives insight into the dependency of the optimum on the model and cost parameters. We provide also illustrating examples for the most important approximations and validate the approximate solution formula by numeric optimization.**

***Keywords*—optimization; cloud model; queueing model; N-policy**

I. INTRODUCTION

Cloud computing [1] [2] is a distributed computing paradigm gaining more importance in the last decade. This is driven by rapidly growing demand for computational resources needed by applications in many areas, like e.g., business, science or web-applications. In this work, we deal with Infrastructure-as-a-Service (IaaS) type Cloud service, in which computing resources are delivered to customers. One of the key attribute of Cloud services is the virtualization which enables to decouple the computing resources from the physical hardware and deliver them to customers as Virtual Machines (VM).

Performance evaluation of Cloud services plays a central role for Cloud service providers to get insights into the relationships among the used resources and the performance in order to meet the performance requirements of the user. The users want guaranteed performance and probably will also require Service Level Agreements (SLAs) on Cloud performance in a later, mature phase of business models for Cloud service. However, Cloud depends on many factors, which makes its performance evaluation to a complex issue. Analytic models are either too simplified to obtain meaningful relationships or lead to rather complex numeric solution, which does not provide an explicit relationships among the used resources and the performance. There are many research works on performance modeling of Clouds. In [3], a multi-level interacting stochastic sub-models approach is proposed, which provides a numeric method to compute the performance measures. For an overview on research works on performance evaluation of clouds the reader is referred to the survey [4] and the references herein.

Cloud cost optimization enables the Cloud service provider the service provisioning at minimum cost. It requires an energy efficient resource management technique. Such resource management and allocation policies for Clouds are summarized in [5] [6]. One efficient resource control mechanism for Clouds is the threshold based activation and deactivation of VMs, which can be modeled by hysteresis queue. Such resource control is proposed in [7], in which computational algorithms are provided for computing the optimal thresholds. Another numerical approaches to cloud cost optimization are presented in [8] and [9]. Optimization of Clouds is even more complex issue than its performance evaluation. Hence it is not surprising, that the vast majority of works on Cloud cost optimization proposes a computational solution.

In this paper, we present a performance evaluation and optimization of an IaaS Cloud model with a proposed simple threshold based resource control, but in contrast to the vast majority of relevant works we provide an approximate explicit formula for determining the only threshold of the control mechanism. The formula holds in most relevant range of parameters. The newly introduced resource control is called as shifted N-policy. According to this policy, a predefined portion of VMs are always active. The remaining ones are activated simultaneously when the number of requests reaches a threshold (like in N-policy) and deactivated when the number of requests falls below the predefined portion of active VMs. This explains the name of the policy. The cloud is modeled by multi-server $M/M/m/K$ queue. Note that, as pointed out in [10], the $M/M/m$ queue can be an acceptable approximation of the $GI/GI/m$ queue until the coefficient of variations of both the interarrival and the service times are not far from 1.

 ISBN: 978-1-61208-860-0

We present closed form results for the stationary distribution of the number of requests and for several performance measures in the shifted N-policy $M/M/m/K$ model. The cost model leads to a discrete optimization task, which can be approximated by a nonlinear continuous optimization task. It turns out that the objective function is not convex everywhere on its definition range. After applying several approximations, including Stirling's formula, we reduce the problem to a nonlinear equation with a specific structure including factorial terms. We provide the approximate solution of the optimization task for a bounded range of parameters. The major contribution of the work is the proposed shifted N-policy resource control and the closed form approximate solution formula for the optimal value of the threshold N under the most relevant range of parameters. The secondary contribution of the work is the stationary analysis of the shifted N-policy $M/M/m/K$ model. The advantage of using the proposed shifted N-policy is that it makes the cloud resource management very simple due to the approximate analytic formula for the optimal threshold, i.e., no need for computational algorithm. On the other hand it leads to somewhat higher optimal cost than other more complex computational solutions, like e.g., the hysteresis policy with multi-thresholds. The proposed optimization can be used for example for the use case "Enabling add-on services on top of the infrastructure", like e.g., computing-as-a-service, analytics or Business Intelligence(BI)-as-a-service.

We also provide illustrating examples for the most important approximations and validate the approximate solution formula by numeric optimization in the relevant range of parameters.

The rest of this paper is organized as follows. Section II is devoted to the description of the model. The stationary analysis of the queueing model is given in Section III. In Section IV, we construct the cost function to be optimized. The approximate minimization is discussed in Section V. In Section VI, we give illustrative examples for the approximations and provide the numeric validation of the approximate solution formula. The work is concluded in Section VII.

II. Cloud model description

A. IaaS cloud model

The IaaS Cloud delivers low-level computational resources to the users. The Physical Machines (PMs) are grouped into two pools: active (running) and standby machines. The PMs in standby can represent either turned-on (but not ready) or turned-off machines. The computational resources are provided to users in the form of VMs. Total number of available VMs is $M > 100$, from which $0.1M \leq L \leq 0.5M$ VMs are always active. The resource control is realized by threshold based activation and deactivation of the remaining $M - L$ VMs. The model has buffer with capacity for $K - M \geq 1$ VMs. When all active VMs are busy upon arrival of a new request then the request is directed into the buffer, where it waits until getting an access to a VM becoming free. When the buffer is full upon arrival of a new request, then the request is lost.

B. Shifted N-policy queueing model

The queueing system modeling the IaaS cloud is an $M/M/m/K$ queue with shifted N-policy. In the queueing context the VMs are called as servers. The request arrive according to Poisson process with rate $\lambda > 0$ and the service times are exponentially distributed with parameter $\mu > 0$. The arrival process and the service process are assumed to be mutually independent. The system has $m = M \geq 1$ servers and buffer capacity for $K - M \geq 1$ requests.

The control of the VMs is realized by the newly proposed shifted N-policy. According to this policy $L < M$ servers are always active. When the queueing system is empty then the remaining $M - L$ servers are in standby. They will be activated simultaneously when the number of requests in the system reaches the threshold $L + 1 \leq N \leq M$. After having all the M servers active, $M - L$ servers will be deactivated simultaneously, when the number of requests in the system reaches again L. This policy has hysteresis-like characteristic upwards (in number of requests), which makes it suitable to be used for energy efficient resource control. However, it is much simpler than the hysteresis queue, which could facilitate the developing of analytically tractable approximation.

The queue is always stable, since it can be modeled by a finite state Continuous-Time Markov chain (CTMC). The utilization of the system, denoted by ρ is given by

$$\rho = \frac{\lambda}{M\mu}. \tag{1}$$

C. Cost model

The cloud provider encounters different type of costs with different weights. These are taken into account by the help of cost parameters, which are defined by

- C_{on} - cost of an active VM/time unit,
- C_{off}- cost of a standby VM/time unit,
- C_W - cost of waiting of a request (=holding a request in the buffer)/time unit ,
- C_R - cost of loss of an arriving request,
- C_A - activation cost of a VM (changing from standby to active state),
- C_D - deactivation cost of a VM (changing from active to standby state).

Using these parameters the cloud cost can be specified by the following function

$$\begin{aligned} \mathcal{C}_{cloud} = {} & E[\text{ number of active servers }]\ C_{on} \\ & + E[\text{ number of standby servers }]\ C_{off} \\ & + E[W]\ C_W + p_{loss}\ \lambda\ C_R, \\ & + (\text{ activation rate of standby VMs })\ (M-L)\ C_A \\ & + (\text{ deactivation rate of active VMs })\ (M-L)\ C_D. \end{aligned} \tag{2}$$

where $E[\ ]$ stands for the expected value of a random variable, W is the waiting time of the requests in the buffer and p_{loss} is the probability of loss.

Note that the operation of N-policy implies that one of the major trade-off of the model is the relation $C_{on} - C_{off}$ versus C_W, which in fact appears also in the approximate formula for computing the threshold N (via parameter A see in subsection V-D).

III. Analysis of the Queuing Model

Let $n \geq 0$ be the number of requests in the system. The process $\{n(t), t \geq 0\}$ is a finite state CTMC.

A. State diagram

The state diagram of the $M/M/m/K$ queue with shifted N-policy can be seen in Figure 1.

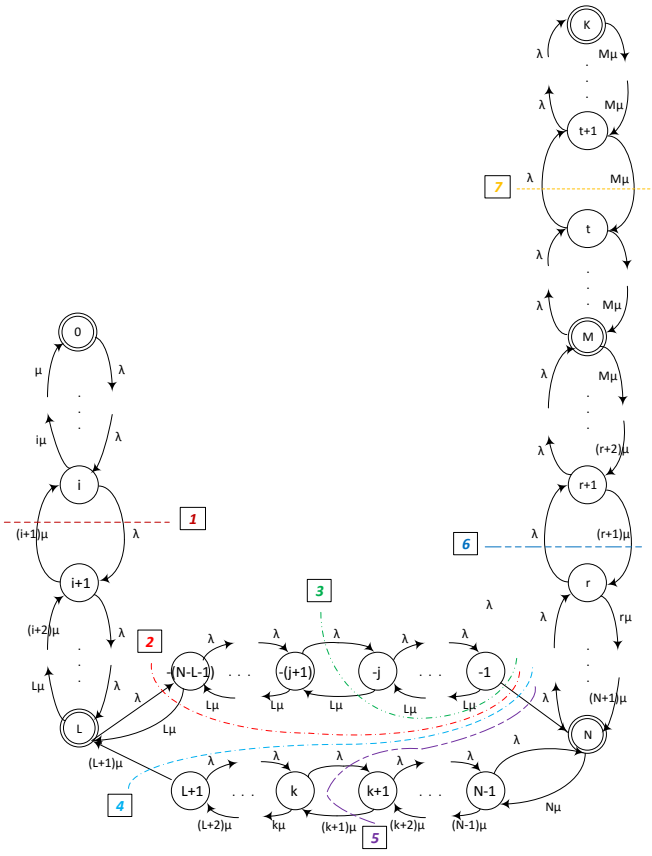

Figure 1. State diagram.

Basically the states are denoted according to the number of requests in the system. However, the notation of the states, in which the $L < n < N$, depends on the number of active servers. If there are L active servers then the states are denoted by the number $-(N-n)$. Otherwise (i.e., there are M active servers) the default numbering, n are used. On this way the states can be described as a contiguous range $[-(N-L-1), \ldots, K]$.

B. Stationary analysis

We perform the stationary analysis rather by utilizing the principle of global balance equations instead of applying the standard way by means of equilibrium equations. This results in shorter derivations for the stationary distribution of the number of requests in the system. We define the stationary probability, p_i as the probability that the system is in state i, for $-(N-L-1) \leq i \leq K$.

1) Global balance equations: We marked the selected set of states used for the balance equations on the state diagram. Each case is marked by a separator line and an associated number in small square, which is used to identify the case.

1) $(i+1)\mu p_{i+1} = \lambda p_i$, $i = 0, \ldots, L-1$,
2) $L\mu p_{-(N-L-1)} + \lambda p_{-1} = \lambda p_L$,
3) $L\mu p_{-j} + \lambda p_{-1} = \lambda p_{-(j+1)}$, $j = -(N-L-2), \ldots, -1$,
4) $(L+1)\mu p_{L+1} = \lambda p_{-1}$,
5) $(k+1)\mu p_{k+1} = \lambda p_k + \lambda p_{-1}$, $k = L+1, \ldots, N-1$,
6) $(r+1)\mu p_{r+1} = \lambda p_r$, $r = N, \ldots, M-1$,
7) $M\mu p_{t+1} = \lambda p_t$, $t = M, \ldots, K-1$.

2) Stationary distribution of the number of requests: By solving the balance equations we get the stationary distribution of the number of requests as

$$
\begin{aligned}
p_k &= \frac{(\frac{\lambda}{\mu})^k}{k!} p_0, \text{ for } k = 0, \ldots, L, \\
p_k &= (\frac{\lambda}{L\mu})^{N-L} \frac{(\frac{\lambda}{L\mu})^k - 1}{1 - (\frac{\lambda}{L\mu})^{N-L}} p_L, \\
&\quad \text{for } k = -(N-L-1), \ldots, -1, \\
p_k &= \sum_{i=L}^{k-1} \frac{i!}{k!} (\frac{\lambda}{\mu})^{k-i} p_{-1}, \text{ for } k = L+1, \ldots, N, \\
p_k &= \frac{N!}{k!} (\frac{\lambda}{\mu})^{k-N} p_N, \text{ for } k = N+1, \ldots, M, \\
p_k &= (\frac{\lambda}{M\mu})^{k-M} p_M, \text{ for } k = M+1, \ldots, K. \qquad (3)
\end{aligned}
$$

The probabilities p_L, p_{-1}, p_N and p_M are probabilities of events representing some boundary in the operation of the considered queueing model. They are given by

$$
\begin{aligned}
P_L &= \frac{(\frac{\lambda}{\mu})^L}{L!} p_0, \\
p_{-1} &= \alpha \; p_L, \text{ where } \alpha = (\frac{\lambda}{L\mu})^{N-L-1} \frac{1 - \frac{\lambda}{L\mu}}{1 - (\frac{\lambda}{L\mu})^{N-L}}, \\
p_N &= \sum_{i=L}^{N-1} \frac{i!}{N!} (\frac{\lambda}{\mu})^{N-i} p_{-1} = \frac{(\frac{\lambda}{\mu})^N}{N!} s_{L,N} \; \alpha \; p_L, \\
&\quad \text{where } s_{L,N} = \sum_{i=L}^{N-1} \frac{i!}{(\frac{\lambda}{\mu})^i}, \\
p_M &= \frac{N!}{M!} (\frac{\lambda}{\mu})^{M-N} p_N, \qquad (4)
\end{aligned}
$$

3) Performance measures: The performance measures p_{loss}, $p_{s1} = P\{$ the number of active VMs = L $\}$ and $E[W]$ influence the cloud cost. They are given by

 ISBN: 978-1-61208-860-0

$$p_{loss} = p_K = (\frac{\lambda}{M\mu})^{K-M} p_M = (\frac{\lambda}{M\mu})^K \frac{M^M}{M!} \frac{N!}{(\frac{\lambda}{\mu})^N} p_N \quad (5)$$

$$\begin{aligned} p_{s1} &= \sum_{k=0}^{L} p_k + \sum_{k=-(N-L-1)}^{-1} p_k \quad (6)\\ &= \sum_{k=0}^{L} \frac{(\frac{\lambda}{\mu})^k}{k!} p_0 + \sum_{k=-(N-L-1)}^{-1} (\frac{\lambda}{L\mu})^{N-L} \frac{(\frac{\lambda}{L\mu})^k - 1}{1-(\frac{\lambda}{L\mu})^{N-L}} p_L \\ &= \sum_{k=0}^{L} \frac{(\frac{\lambda}{\mu})^k}{k!} p_0 + \sum_{k=1}^{N-L-1} \frac{(\frac{\lambda}{L\mu})^k - (\frac{\lambda}{L\mu})^{N-L}}{1-(\frac{\lambda}{L\mu})^{N-L}} p_L \\ &= \sum_{k=0}^{L} \frac{(\frac{\lambda}{\mu})^k}{k!} p_0 + \frac{\frac{\frac{\lambda}{L\mu} - (\frac{\lambda}{L\mu})^{N-L}}{1-\frac{\lambda}{L\mu}} - (N-L-1)(\frac{\lambda}{L\mu})^{N-L}}{1-(\frac{\lambda}{L\mu})^{N-L}} p_L. \end{aligned}$$

$$\begin{aligned} E[W] &= \sum_{k=-(N-L-1)}^{-1} (k+N-L) p_k + \sum_{k=M}^{K} (k-M) p_k \\ &= \sum_{k=1}^{N-L-1} k\ p_{-(N-L)+k} + \sum_{k=M}^{K} (k-M) p_k \\ &= \tau p_L + \sigma p_M, \quad (7) \end{aligned}$$

where

$$\begin{aligned} \tau &= \frac{\frac{\lambda}{L\mu}}{(1-(\frac{\lambda}{L\mu})^2} \quad (8)\\ &\quad - (N-L)\ \frac{(\frac{\lambda}{L\mu})^{N-L}}{1-(\frac{\lambda}{L\mu})^{N-L}} \left(\frac{1}{1-\frac{\lambda}{L\mu}} + \frac{N-L-1}{2} \right), \\ \sigma &= \frac{\lambda}{M\mu} \frac{1-(\frac{\lambda}{M\mu})^{K-M+1}}{(1-\frac{\lambda}{M\mu})^2} - (K-M+1) \frac{(\frac{\lambda}{M\mu})^{K-M+1}}{1-\frac{\lambda}{M\mu}}. \end{aligned}$$

IV. Cost function

A. Constructing the cost function

The cost function, to be optimized, can be constructed by applying the cost model (2) to the shifted N-policy queue. The so far unknown terms arising in (2) can be expressed with the help of parameters, stationary probabilities and performance measures of the shifted N-policy queue as follows.

$$\begin{aligned} &E[\text{ number of active servers }] = L + (1-p_{s1})(M-L), \quad (9)\\ &E[\text{ number of standby servers }] = p_{s1}(M-L),\\ &(\text{ activation rate of standby VMs }) = p_{-1}\lambda,\\ &(\text{ deactivation rate of active VMs }) = p_{L+1}(L+1)\mu. \end{aligned}$$

Substituting the expressions (9) into (2) we get the cost function, F_1 as

$$\begin{aligned} F_1 = {} & p_{-1}\lambda\ (M-L)\ C_A + p_{L+1}(L+1)\mu\ (M-L)\ C_D \\ & + (L + (1-p_{s1})(M-L))\ C_{on} + p_{s1}(M-L)\ C_{off} \\ & + E[W]\ C_W + p_{loss}\ \lambda\ C_R. \quad (10) \end{aligned}$$

After performing several rearrangements on (10) and using the balance equation $(L+1)\mu p_{L+1} = \lambda p_{-1}$ as well as (4), (5) and (7) we get the cost function in terms of p_L and p_{s1} as

$$\begin{aligned} F_1 = {} & ((\lambda(C_A + C_D)(M-L) + \eta\ s_{L,N})\alpha + C_W\tau)\, p_L \\ & - (C_{on} - C_{off})(M-L)p_{s1} + MC_{on}, \text{ where} \quad (11)\\ \eta \ = {} & \left(C_R\lambda(\frac{\lambda}{M\mu})^K \frac{M^M}{M!} + C_W\sigma \frac{(\frac{\lambda}{\mu})^M}{M!} \right). \end{aligned}$$

B. Approximating the cost function

The optimization of (11) with respect to N seems not to be tractable on analytic way due to the complex dependency of several of its terms on N, like $s_{L,N}$ or p_{s1}. Therefore we establish approximation for (11), which on the other hand restricts the parameter range, for which it holds.

1) Approximations for α, τ and p_{s1}: When $N-L \gg 1$ then $(\frac{\lambda}{L\mu})^{N-L} \gg 1$ holds for the traffic range $\frac{\lambda}{L\mu} > 1$ and thus the term $1-(\frac{\lambda}{L\mu})^{N-L}$ and $(N-L-1)$ can be approximated by $-(\frac{\lambda}{L\mu})^{N-L}$ and $(N-L)$, respectively. Utilizing it in the expression of α, τ and p_{s1} ((4), (8) and (6)) gives the approximation α^*, τ^* and p_{s1}^*, respectively as

$$\begin{aligned} \alpha^* \ &\approx 1 - \frac{L\mu}{\lambda}, \\ \tau^* \ &\approx \frac{\frac{L\mu}{\lambda}}{1-\frac{L\mu}{\lambda}} \left(\frac{1}{1-\frac{L\mu}{\lambda}} - (N-L) \right) + \frac{(N-L)(N-L)}{2}, \\ p_{s1}^* &\approx (N-L)p_L, \quad (12) \end{aligned}$$

where at evaluating p_{s1}^* we also used the upper limit $\sum_{k=0}^{L} \frac{(\frac{\lambda}{\mu})^k}{k!} \le \frac{1}{1-\frac{L\mu}{\lambda}} \frac{(\frac{\lambda}{\mu})^L}{L!}$ for $L \gg 1$.

2) Utilizing the approximately N independent regions of p_0: Unfortunately p_0, which is involved in almost every term of (11) via the expression of p_L, depends on N. Now we identify parameter regions, in which p_0 is approximately independent of N. This leads to further restriction on the parameter range. By defining the probability sums

$$\begin{aligned} p_{s1w} &= \frac{1}{p_0} p_{s1} \\ p_{s2w} &= \frac{1}{p_0} \sum_{L+1}^{N} p_k = \sum_{L+1}^{N} \frac{(\frac{\lambda}{\mu})^k}{k!} \sum_{i=L}^{k-1} \frac{i!}{(\frac{\lambda}{\mu})^i} \alpha \frac{p_L}{p_0}, \\ p_{s3w} &= \frac{1}{p_0} \sum_{N+1}^{M} p_k = \frac{N!}{(\frac{\lambda}{\mu})^N} \sum_{N+1}^{M} \frac{(\frac{\lambda}{\mu})^k}{k!} \frac{p_N}{p_0} \\ &= \sum_{N+1}^{M} \frac{(\frac{\lambda}{\mu})^k}{k!} \sum_{i=L}^{N-1} \frac{i!}{(\frac{\lambda}{\mu})^i}\ \alpha\ \frac{p_L}{p_0}, \\ p_{s4w} &= \frac{1}{p_0} \sum_{M+1}^{K} p_k. \end{aligned}$$

p_0 can be given by $p_0 = \frac{1}{p_{sw}}$ with $p_{sw} = p_{s1w} + p_{s2w} + p_{s3w} + p_{s4w}$. It can be seen by taking the difference of p_{s2w}

and p_{s3w} with respect to N that the sum $p_{s2w} + p_{s3w}$ is approximately independent of N and equals to $\sum_{i=L+1}^{M} \frac{(\frac{\lambda}{\mu})^k}{k!}\ \alpha^*$. Furthermore it can be seen that the magnitude of $p_{s2w} + p_{s3w}$ increases rapidly with ρ and for $M/L \gtrapprox 2$ with $\rho \gtrapprox 1.2\frac{L}{M}$ it is much higher than the one of p_{s1w}, which depends on N approximately linearly due to $p_{s1w} \approx (N-L)\frac{p_L}{p_0}$. Moreover the term p_{s4w} is independent of N in this parameter range. We omit the details here due to the limitation on the size of the paper. Summarizing all the above, if $M/L \gtrapprox 2$ and $\rho \gtrapprox 1.2\frac{L}{M}$ then p_{sw} and therefore also p_0 is approximately independent of N. For this case the minimizing task reduces to find the minimum of the function F_2, which can be obtained from (11) by omitting the N independent term MC_{on} and dividing it by p_L. This results in

$$F_2 = ((\lambda(C_A + C_D)(M-L) + \eta\ s_{L,N})\alpha + C_W\tau) - (C_{on} - C_{off})(M-L)\frac{p_{s1}}{p_L}. \quad (13)$$

3) Applying the approximations for α, τ and p_{s1}: The minimizing task can be further reduced to find the minimum of the objective function F_{2app}, which can be obtained by applying the approximations (12) in (13). This leads to

$$F_{2app} = (\lambda(C_A + C_D)(M-L) + \eta\ s_{L,N})\,(1 - \frac{L\mu}{\lambda}) + C_W\frac{\frac{L\mu}{\lambda}}{1-\frac{L\mu}{\lambda}}\left(\frac{1}{1-\frac{L\mu}{\lambda}} - (N-L)\right) + C_W\frac{(N-L)(N-L)}{2} - (C_{on} - C_{off})(M-L)(N-L). \quad (14)$$

C. Approximate equation for determining the local minimum

We obtain an approximate equation for determining the local minimum of (13) by taking its difference with respect to N and setting $\Delta_N F_{2app} \approx 0$. Using $\Delta_N s_{L,N} = \frac{(N-1)!}{(\frac{\lambda}{\mu})^{N-1}}$ and $\Delta(N-L)(N-L) \approx 2(N-L)$ this leads to the equation

$$\eta(1 - \frac{L\mu}{\lambda})\frac{(N-1)!}{(\frac{\lambda}{\mu})^{N-1}} = (C_{on} - C_{off})(M-L) + C_W\frac{\frac{L\mu}{\lambda}}{1-\frac{L\mu}{\lambda}} - C_W(N-L). \quad (15)$$

V. Approximate Minimization of the Cost Function

In order to get closer to the solution of equation (15) first we investigate its structure.

A. Structure of the equation

To identify the structure of equation (15), we simplify its form by applying further approximations. The relation $K - M - 1 >> 1$ holds usually under practical settings. Hence the term $(\frac{\lambda}{M\mu})^{K-M+1}$ can be neglected due to $\rho = \frac{\lambda}{M\mu} < 1$, which gives an approximation for σ as

$$\sigma = \frac{\lambda}{M\mu}\frac{1-(\frac{\lambda}{M\mu})^{K-M+1}}{(1-\frac{\lambda}{M\mu})^2} - (K-M+1)\frac{(\frac{\lambda}{M\mu})^{K-M+1}}{1-\frac{\lambda}{M\mu})} \approx \frac{\rho}{(1-\rho)^2}. \quad (16)$$

Applying again the negligibility of the term $(\frac{\lambda}{M\mu})^{K-M}$ in the expression of η and further rearrangement leads to an approximation for η as

$$\eta = \left(C_R\lambda(\frac{\lambda}{M\mu})^{K-M}\frac{(\frac{\lambda}{\mu})^M}{M!} + C_W\sigma\frac{(\frac{\lambda}{\mu})^M}{M!}\right) \approx C_W\frac{\rho}{(1-\rho)^2}\frac{(\frac{\lambda}{\mu})^M}{M!}. \quad (17)$$

Using (17) in the equation (15) and further rearrangement gives the simplified form of the equation as

$$\frac{(\frac{\lambda}{\mu})^M}{M!}\frac{(N-1)!}{(\frac{\lambda}{\mu})^{N-1}}u_0(\rho) = r(\rho,N), \text{ where} \quad (18)$$

$$u_0(\rho) = C_W\frac{\rho}{(1-\rho)^2}(1 - \frac{1}{\rho\frac{M}{L}}) \quad \text{and}$$

$$r(\rho,N) = C_W\left(A(M-L) + \frac{1}{\rho\frac{M}{L}-1} - (N-L)\right)$$

$$\text{with } A = \frac{C_{on} - C_{off}}{C_W}.$$

The term $\frac{(\frac{\lambda}{\mu})^M}{M!}\frac{(N-1)!}{(\frac{\lambda}{\mu})^{N-1}}$ on the left hand side (lhs) of (18) constitutes the structure of the equation. Its magnitude varies in a huge range for larger M and N depending on the value of the parameters. Therefore we also use its natural logarithm in the course of the analysis. By introducing the notation

$$p(\rho,N) = \frac{(\frac{\lambda}{\mu})^M}{M!}\frac{(N-1)!}{(\frac{\lambda}{\mu})^{N-1}}, \quad (19)$$

the equation (18) can be given in a short form as

$$p(\rho,N)u_0(\rho) = r(\rho,N). \quad (20)$$

B. Properties of function $p(\rho, N)$

The approximate global solution of the considered minimization task requires the knowledge of several properties of function $p(\rho, N)$.

1) Dependency on ρ: Applying the Stirling formula $n! \approx \sqrt{2\pi}n^{(n+1/2)}e^{-n}$ to both M and $N-1$ in the expression (19) gives an approximation for $p(\rho, N)$ as

$$p(\rho,N) = \frac{(\frac{\lambda}{\mu})^M}{M!}\frac{(N-1)!}{(\frac{\lambda}{\mu})^{N-1}} = (\frac{\lambda}{\mu M})^{(M-N+1)}\frac{M^M}{M!}\frac{(N-1)!}{M^{N-1}} \approx \rho^{(M-N+1)}e^{(M-N+1)}\sqrt{\frac{N-1}{M}}(\frac{N-1}{M})^{N-1}. \quad (21)$$

It can be seen from (21) that the dependency of $p(\rho, N)$ on ρ is exponential. This leads to rapid changes under the typical model parameter settings, e.g., increasing ρ by 2.5% at $M - N + 1 = 95$ leads to 10 times multiplication due to $1.025^{95} \approx 10$.

2) Dependency of $p(\rho, N)$ on N: Taking the natural logarithm of (21) we get

$$\ln(p(\rho, N)) = (M - N + 1)\,(\ln(\rho) + 1) + \left((N-1) + \frac{1}{2}\right)\ln(\frac{N-1}{M}).$$

By introducing the notation

$$\beta = \frac{N-1}{M}. \tag{22}$$

this can be rewritten as

$$\ln(p(\rho, \beta)) = M\left((1-\beta)(\ln(\rho)+1) + (\beta + \frac{1}{2*M})\ln(\beta)\right). \tag{23}$$

Taking its first derivative with respect to β gives

$$\frac{d\ln(p(\rho,\beta))}{d\beta} = M\left(\ln(\frac{\beta}{\rho}) + \frac{1}{2*M*\beta}\right) \approx M\ln(\frac{\beta}{\rho}), \tag{24}$$

since in the typical model parameter ranges $M >> 100$ and thus the term $\frac{1}{2*M*\beta}$ can be neglected. The first derivative of $(p(\rho, N)$ with respect to N comes by using $\frac{d(p(\rho,N))}{dN} = \frac{d(e^{\ln(p(\rho,N))})}{dN} = p(\rho,N)\frac{d\ln(p(\rho,\beta))}{d\beta}\frac{d\beta}{dN} = p(\rho,N)\frac{1}{M} * \frac{d\ln(p(\rho,\beta))}{d\beta}$, which yields

$$\frac{d(p(\rho,N))}{dN} \approx p(\rho,N)\ln(\frac{\beta}{\rho}). \tag{25}$$

The $\ln(\frac{\beta}{\rho})$ divides the $\beta-\rho$ plane into two disjunct subareas regarding the characteristic of $p(\rho, N)$ with respect to N as

$$p(\rho,N) \text{ is } \left\{ \begin{array}{l} \text{monotone decreasing, if } \beta < \rho \\ \text{monotone increasing, if } \beta \geq \rho \end{array} \right\}. \tag{26}$$

Hence the dependency of $p(\rho, N)$ on N is faster than exponential, since $|\ln(\frac{\beta}{\rho})|$ is increasing with decreasing N and increasing N in the range $\beta < \rho$ and $\beta > \rho$, respectively.

3) The "low magnitude range": We investigate the case when $p(\rho, N) = e^{const}$ holds, where $const$ is a given real constant. With the notation of β this equation can be given by

$$M\left((1-\beta)\ (\ln(\rho)+1) + (\beta + \frac{1}{2*M})\ ln(\beta)\right) = const. \tag{27}$$

Observe that this equation implicitly defines a boundary function $\beta(\rho)$, which separates the "low magnitude range" $p(\rho, N) \leq e^{const}$ from the complementer range, in which $p(\rho, N) > e^{const}$. In the range $p(\rho, N) \leq e^{const}$ the magnitude of $p(\rho, N)$ is less than $const$, which explains the name "low magnitude range". We say that a $\beta-\rho$ point is inside and outside of the "low magnitude range" if $p(\rho, \beta) \leq e^{const}$ holds and does not hold for that point, respectively. By rearranging (27) we get the expression of $\ln(\rho)$ along the boundary function as

$$ln(\rho) = \frac{const}{(1-\beta)*M} - \frac{\beta}{1-\beta}\ln(\beta) - 1 - \frac{1}{(1-\beta)*2*M}\ln(\beta). \tag{28}$$

Therefore, the sensitivity of $ln(\rho)$ with respect to the $const$, ζ is given by

$$\zeta = \frac{1}{(1-\beta)*M}. \tag{29}$$

An upper limit for the factor $\ln(\frac{\beta}{\rho})$ determining the relation between $p(\rho, N)$ and its first derivative with respect to N (see (25)) along the boundary function can be obtained as

$$\begin{aligned}\ln(\frac{\beta}{\rho}) = \ln(\beta) - \ln(\rho) &= \ln(\beta) + \frac{\beta}{1-\beta}\ln(\beta) + 1 \\ &- \left(\frac{const}{(1-\beta)*M} - \frac{1}{(1-\beta)*2*M}\ln(\beta)\right) \\ &\leq \frac{1}{1-\beta}\ln(\beta) + 1 \leq -\frac{1}{2}(1-\beta) < 0.\end{aligned} \tag{30}$$

where we used the inequality $\ln(\beta) \leq -(1-\beta) - \frac{1}{2}(1-\beta)^2$ and that the term in the brackets is non-negative. Hence the boundary curve lies under the line separating the $\beta-\rho$ plane into parts with monotone decreasing and increasing $p(\rho, N)$ with respect to N. The relevant region of the $\beta-\rho$ plane is restricted by $\beta > \beta_{low} = \frac{L}{M}$ and $\rho \geq \beta_{low}$ due to the limitations $N > L \Leftrightarrow \frac{N}{M} > \frac{L}{M}$ and $\frac{\lambda}{\mu} > L \Leftrightarrow \rho > \frac{L}{M}$, respectively. The cross point of the horizontal $\beta = \beta_{low}$ and the boundary curve is called boundary ρ and denoted by ρ_b. All these are shown on the illustrating example Figure 2.

C. Constructing the approximate minimization

1) Solution regimes: For the sake of better understanding the idea of the solution, first we consider a modified form of the equation (20) as

$$p(\rho, N) = r(\rho, N). \tag{31}$$

The idea of the approximate solution is based on the concept of "low magnitude range". When setting the r.h.s of (31) to 0 and the solution of $r(\rho, N) = 0$, let us say N_s, falls inside of the "low magnitude range" with $const = \ln(C_W)$, then it ensures that the value of $r(\rho, N)$ reaches the value of $p(\rho, N) \leq e^{const} = C_W$ by decreasing N not more than 1, since $\frac{d(r(\rho,N))}{dN} = -C_W$ and both the value of $p(\rho, N)$ and its first derivative are $<< C_W$ in a large portion of the "low magnitude range" (up to close to its boundary). Therefore, N_s can be considered as approximate solution of (31).

More precise specification of the inside area of the needed boundary requires both $p(\rho, N) < C_W$ and $\frac{d(p(\rho,N))}{dN} \approx$

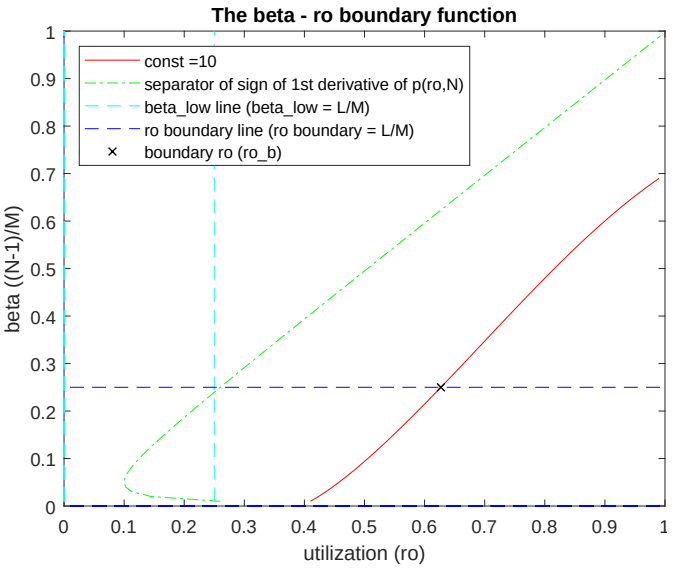

Figure 2. The $\beta - \rho$ boundary function, const = 10, M=200, L=50.

$p(\rho, N)\ln(\frac{\beta}{\rho}) > -C_W$ to be hold. However, the second condition leads to curve on $\beta-\rho$ plane very close to the boundary curve of the "low magnitude range" with $const = \ln(C_W)$. This is because

$$M\left((1-\beta)\ (\ln(\rho)+1) + (\beta + \frac{1}{2*M})\ ln(\beta)\right) + \ln(-\ln(\frac{\beta}{\rho})) = \ln(C_W) \quad (32)$$

leads to a change in $\ln(\rho)$ in absolute value as $\frac{\ln(-\ln(\frac{\beta}{\rho}))}{(1-\beta)*M}$, which is very small under the most relevant range of parameters. For example it is ≤ 0.04 in absolute value for $M \geq 100$ and $-2.3 \leq \ln(\frac{\beta}{\rho}) \leq -0.35$ due to $0.1 \leq \frac{\beta}{\rho}$ for $\beta_{low} \geq 0.1$ as well as using $\beta \leq 0.7$, which can be shown from the properties of this second $\beta - \rho$ curve. Therefore, the second curve can be neglected from the specification of the required inside area and hence it is enough to specify the needed boundary by $p(\rho, N) = const$ for any $const$.

We denote the boundary ρ under the specific condition $const = \ln(C_W)$ by ρ_0. Approximately at $N = L$, the first derivative of $p(\rho_0, N)$ equals to $-C_W$. At this point $r(\rho_0, L) > p(\rho_0, L)$. By decreasing N, from that point the first derivative of $p(\rho_0, N)$ is in absolute value greater than that one of $r(\rho_0, N)$, and hence an other cross point of the functions $p(\rho_0, N)$ and $r(\rho_0, N)$ must arise, let us say at $N = N_1$. This is a maximum point of the cost function, since (in N) below this point the sign of $p(\rho, N) - r(\rho, N)$ changes from negative to positive. Further decreasing N it reaches the point $N = N_2$, where the value of the cost function is less then at N_s. The situation is illustrated on Figure 3.

The above discussed decrease in any range of N, in which $p(\rho_0, N)$ is monotone decreasing with respect to N, causes an increase in the value of $p(\rho_0, N)$, which equivalently can be

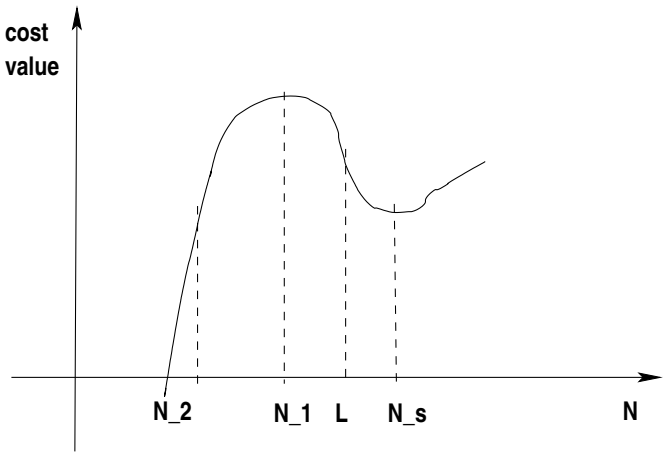

Figure 3. Example cost function.

also considered as a change in $const$ of (27) while keeping N unchanged. This change in $const$ corresponds to a shift of the boundary curve to right. If $\rho > \rho_0$ then the point in β corresponding to N_1 can fall over the β_{low} line. Until N_2 falls still below the β_{low} line, the value of the cost function at β_{low} is still higher than at β_s (corresponding to N_s), and therefore the global minimum of the cost function is still at N_s. However, if N_2 also falls above β_{low} line then the global minimum of the cost function is at β_{low} (corresponding to N=L+1). If $\rho > \rho_0$, it can also happen that β_s falls outside of the "low magnitude range" (= under the boundary curve). In this case $|\frac{d(p(\rho,N))}{dN}| > C_W$ and there is no cross point at all, the cost function is monotone increasing with respect to N and hence the global minimum at β_{low}. Note that in the range $N > N_s$ there can not be any cross point of the functions $p(\rho, N)$ and $r(\rho, N)$, since $p(\rho, N) > 0$ and $r(\rho, N) < 0$ in that range.

It follows from the above argumentation that the global minimum of the cost function is approximately at N_s in the range of $\rho < \rho_0$ and $\beta_{low} \leq \beta_s < 1$. Above ρ_0 there is a gap in ρ until a specific point, ρ_s, at which N_2 reaches the β_{low} line and hence the global minimum of the cost function is still at N_s (for $\beta_{low} \leq \beta_s < 1$). Finally above ρ_s the position of the global minimum of the cost function changes to $N = L + 1$.

The position of ρ_s depends on $\Delta const$, which is the change in $const$ causing a shift of the boundary ρ from ρ_0 to ρ_s.

2) The magnitude of $\Delta const$*:* The solution of $r(\rho, N) = 0$, N_s can be given from (18)) as

$$N_s = A(M-L) + \frac{1}{\rho\frac{M}{L} - 1} + L. \quad (33)$$

We use the notation

$$\Delta N = N_s - L = A(M-L) + \frac{1}{\rho\frac{M}{L} - 1}. \quad (34)$$

The magnitude of $\Delta const$ is about $2\ln(\Delta N)$. The first $\ln(\Delta N)$ stands for the increase $p(\rho_0, L) \rightarrow p(\rho_0, N_1)$, i.e., from C_w up to $(N_s - N_1)C_W \approx (N_s - L)C_W =$

ΔNC_w (= the value of $r(\rho_0, N)$ at β_{low}), on ln level which is $\ln(\frac{\Delta NC_w}{C_w})$. The second one stands for increase $p(\rho_0, N_1) \rightarrow p(\rho_0, N_2)$, on ln level. During $N_1 \rightarrow N_2$ the cost function F_{2app} decreases so much as its increases during $N_s \rightarrow N_1$, which is approximately $(N_s - N_1) \times |$ maximum value of $\frac{d\ F_{2app}}{dN}$ in $[N_s, N_1]| = (N_s - N_1)|p(\rho_0, L) - r(\rho_0, L)| \approx (N_s - N_1)|C_w - (N_s - L)C_w| \approx (N_s - N_1)\Delta NC_w$. On the other hand the change of the cost function F_{2app} during $N_1 \rightarrow N_2$ is in the magnitude of $p(\rho_0, N_2) - p(\rho_0, N_1)$ (again due to the exponential character of function $p(\rho_0, N)$, but we omit the details here due to the limitation on the size of the paper). Putting all these together $\ln \frac{p(\rho_0, N_2)}{p(\rho_0, N_1)} = \ln(\frac{(N_s - N_1)\Delta NC_w}{(N_s - N_1)C_w} + 1) \approx \ln(\Delta N)$. Note that $(N_s - N_1) \times |$ maximum value of $\frac{d\ F_{2app}}{dN}$ in $[N_s, N_1]|$ overestimates the increase of the cost function F_{2app} during $N_s \rightarrow N_1$ and hence $2\ln(\Delta N)$ also overestimates $\Delta const$.

In order to estimate $2\ln(\Delta N)$, we impose a condition on A, which ensures that the term $A(M-L)$ dominates over $\frac{1}{\rho^{\frac{M}{L}}-1}$. For this purpose an upper bound is set on $\frac{1}{\rho^{\frac{M}{L}}-1}$, which can be obtained by setting a lower bound for ρ as $\beta_{low}\xi < \rho < 1$. With $\xi = 1.2$ this gives $\frac{1}{\rho^{\frac{M}{L}}-1} \leq 5$. We set $A(M-L)/(\text{A (M-L)}+\frac{1}{\rho^{\frac{M}{L}}-1}) \geq 0.9$, which causes a difference of $2\ln(0.9) = -0.2$ in the value of $\Delta const$ corresponding to difference of $\frac{-0.2}{(1-0.5)100} = 0.004$ on $\ln(\rho)$ level when assuming $M \geq 100$ and $\beta_{low} < 0.5$. With this setting we get $A(M - L) \geq 45$ which implies the condition on A as

$$A \geq \frac{45}{M - L}, \tag{35}$$

under which $A(M - L) + \frac{1}{\rho^{\frac{M}{L}}-1} \approx A(M - L)$.

Now we can estimate $2\ln(\Delta N)$ as

$$\begin{aligned} 2\ln(\Delta N) &\approx \ln(A(M - L)) \\ &= 2\ln(A) + \ln(M) + \ln(1 - \beta). \end{aligned} \tag{36}$$

3) Relation for ρ_s: So far we discussed the way of solution without considering the term $u_0(\rho)$ on the lhs of equation (20). Now taking into account also the term $u_0(\rho)$, the relation for the boundary curve crossing the β_{low} line at ρ_s can be given by

$$\begin{aligned} &M\left((1 - \beta_{low})\ (\ln(\rho_s) + 1) + (\beta_{low} + \frac{1}{2 * M})\ ln(\beta_{low})\right) \\ &+ \ln(u_0(\rho_s)) = \ln(C_W) + 2\ln(\Delta N). \end{aligned} \tag{37}$$

By substituting the expression of $u_0(\rho)$ from (18) and using $(1 - \frac{1}{\rho_s \frac{M}{L}}) = (1 - \frac{\beta_{low}}{\rho_s}) = \frac{\beta_{low}}{\rho_s}(\frac{\rho_s}{\beta_{low}} - 1)$ we get

$$\begin{aligned} &M\left((1 - \beta_{low})\ (\ln(\rho_s) + 1) + (\beta_{low} + \frac{1}{2 * M})\ ln(\beta_{low})\right) \\ &+ \ln(C_W) + \ln(\rho_s) + \ln(\frac{1}{(1 - \rho_s)^2}) + \ln(\beta_{low}) - \ln(\rho_s) \\ &+ \ln(\frac{\rho_s}{\beta_{low}} - 1) = \ln(C_W) + 2\ln(\Delta N). \end{aligned}$$

Rearranging yields

$$\begin{aligned} &M\left((1 - \beta_{low})\ (\ln(\rho_s) + 1) + (\beta_{low} + \frac{1}{2 * M})\ ln(\beta_{low})\right) \\ &= 2\ln(\Delta N) - \ln(\beta_{low}) - \ln(\frac{1}{(1 - \rho_s)^2}) - \ln(\frac{\rho_s}{\beta_{low}} - 1). \end{aligned} \tag{38}$$

We approximate the term $\frac{1}{(1-\rho_s)^2}(\frac{\rho_s}{\beta_{low}} - 1)$ by setting 1, which gives an uncertainty of ≈ 7 on right hand side (rhs) of (38) ($1 \leq \frac{1}{(1-\rho_s)^2} \leq 100$ for $\rho_s \leq 0.9$ and $0.28 \leq (\frac{\rho_s}{\beta_{low}} - 1) \leq 9$ for $\beta_{low} \geq 0.1$ and $\frac{\rho_s}{\beta_{low}} \geq 1.28$ following from (30) with $\beta_{low} \leq 0.5$ and thus $ln(9 \times 100) < 7$) corresponding to difference of $\frac{7}{(1-0.5)200} \approx 0.07$ on $\ln(\rho_s)$ level when assuming $M \geq 200$ and again $\beta_{low} < 0.5$. The relation $\rho_s \leq 0.9$ can be justified by the approximate solution of (38) for ρ_s by assuming that its rhs ≤ 14 and setting $\beta = \max(\beta_{low}) = 0.5$, since the solution in ρ_s is monotone increasing with respect to β_{low}. Using the above approximation and (36) we get the final form of the relation for ρ_s as

$$\begin{aligned} &M\left((1 - \beta_{low})\ (\ln(\rho_s) + 1) + (\beta_{low} + \frac{1}{2 * M})\ ln(\beta_{low})\right) \\ &= 2\ln(A) + \ln(M) + \ln(1 - \beta_{low}) - \ln\beta_{low}, \end{aligned} \tag{39}$$

D. Approximate solution formula

Now putting all together we get the approximate solution formula.

Conditions

1) $100 \leq M$,
2) $0.1 \leq \beta_{low} \leq 0.5$ with $\beta_{low} = \frac{L}{M}$,
3) $\rho \geq \beta_{low}\xi$ with $\xi = 1.2$,
4) $N - L >> 1$, practically $N > L + 10$,
5) $K - M >> 1$, practically $K > M + 10$,
6) $A \geq \frac{45}{M-L}$

Solution formula

If **Conditions** 1-6 hold, then

$$N_{opt} = \left\{ \begin{array}{ll} \min_{(\lfloor A(M-L) + \frac{1}{\rho^{\frac{M}{L}}-1} + L \rfloor, M)} & \text{if } \rho \leq \rho_s, \\ L + 1 & \text{if } \rho_s < \rho < 1, \end{array} \right\}$$

where

$$\begin{aligned} ln(\rho_s) &= \frac{2\ln(A) + \ln(M) + \ln(1 - \beta_{low}) - \ln\beta_{low}}{(1 - \beta_{low}) * M} \\ &- \frac{\beta_{low}}{1 - \beta_{low}}\ln(\beta_{low}) - 1 \\ &- \frac{1}{(1 - \beta_{low}) * 2 * M}\ln(\beta_{low}). \end{aligned} \tag{40}$$

Observe that the approximate optimal N does not depend on C_A, C_D and C_R. This is because they have no impact on N in the considered range of parameters. The cost parameters C_A, C_D influence N only via p_0 and hence they effect the optimal N in the range, in which p_0 depends on N. The cost parameter C_R has impact on the optimal N via η and hence it is effective only for small values of $K - M$.

 ISBN: 978-1-61208-860-0

VI. Numerical comparisons

In this Section, we illustrate the approximations and validate the approximate solution formula by numeric optimization. The setting $C_{on} = 50$, $C_{off} = 15$ $C_a = 30$, $C_d = 20$ and $C_R = 20$ was used for all experiments. The parameters C_{off}, C_{on} have impact to the solution formula only via the parameter A, which was varied through C_W. The parameters C_a, C_d and C_R have no impact on the approximate solution formula in the considered range of (other) parameters. We applied $100 < M < 1000$ for all experiments.

A. Illustrating the approximations

1) N independent region of p_0: Figure 4 shows the dependency of p_0 for the parameter setting $M = 300$, $L = 100$, $K = 350$ and $\rho = 0.6$. It can be seen on the figure that p_0 is independent of N for $N \gtrapprox 120$, which corresponds to $N - L \approx 20 \gg 1$.

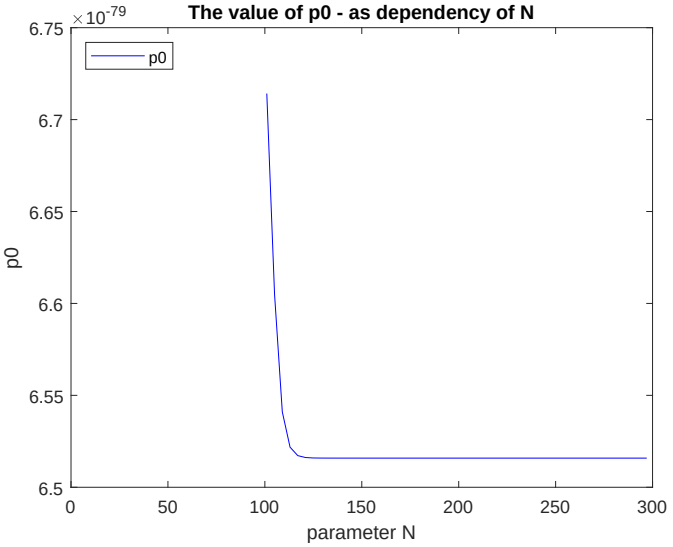

Figure 4. Probability p_0 in dependency of threshold N.

2) Approximation of F_2 by F_{2app}: Figure 5 illustrates the approximation of the cost function F_2 (without taking into account p_0) by F_{2app} in dependency of threshold N for the parameter setting $M = 300$, $L = 100$, $K = 350$, $C_W = 50$, $\mu = 1$ and $\rho = 0.6$. The figure shows a very good match. The mismatch on the left side of the curve is caused by violating the condition $N - L >> 1$ as N becomes close to L.

B. Illustration of the approximate solution formula

The comparison of the exact and approximate optimal N of F_2 can be seen in Figure 6 in dependency of ρ for the parameter setting $M = 400$, $L = 100$, $K = 450$, $C_W = 50$, $\mu = 1$ and $\rho > 0.25 = \frac{L}{M}$.

Figure 7 shows the exact and approximate optimal value of F_1 in dependency of ρ for different values of M with the parameter setting $L = 50$, $K = M + 100$, $C_W = 50$, $\mu = 1$ and $\rho > 0.25 = \frac{L}{M}$.

Both figures show a very good match. The small bias between the exact and approximated ρ_s in Figure 6 can be explained by the uncertainity introduced by setting $\frac{1}{(1-\rho_s)^2}(\frac{\rho_s}{\beta_{low}} - 1)$ to 1.

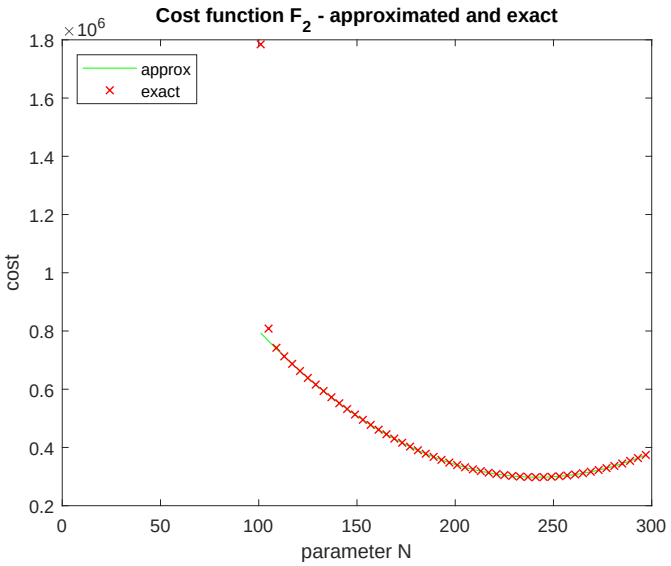

Figure 5. Exact and approximate values of the cost function F_2 in dependency of threshold N.

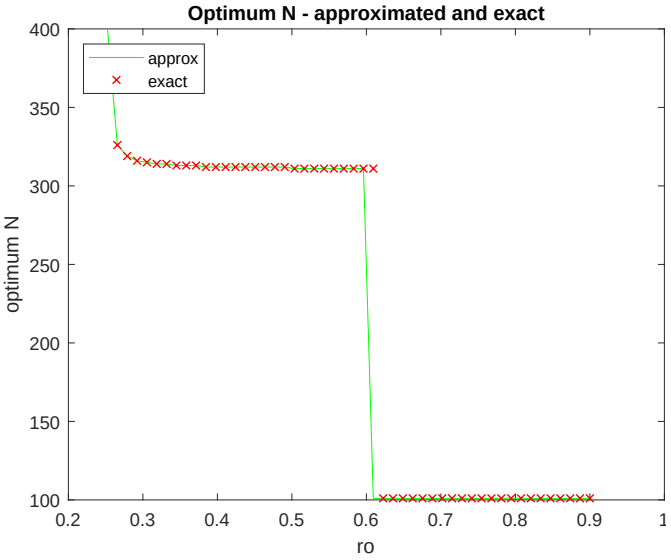

Figure 6. Exact and approximate optimal N (F_2) in dependency of ρ.

C. Validation of the approximate formula

We validated the approximate solution formula by numeric optimization in the considered range of parameters. Figure 8 shows the ratio of the approximated and the exact optimal value of F_1 for the range of parameters $100 \leq M \leq 700$ and $\rho > \frac{L}{M}$ with the parameter setting $L = 50$, $K = M + 100$, $C_W = 50$, $\mu = 1$.

Similarly Figure 9 shows the ratio of the approximated and the exact optimal value of F_1 for the range of parameters

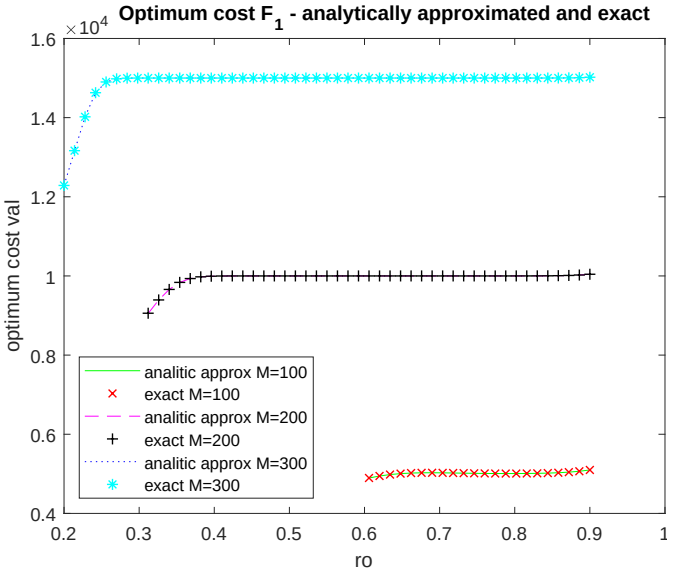

Figure 7. Exact and approximate optimal value (F_1) in dependency of ρ for different values of M.

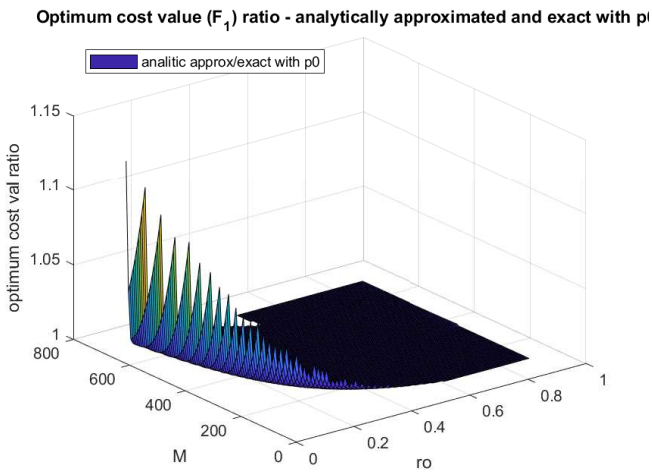

Figure 8. Ratio of the approximated and exact optimal value (F_1) for $100 \leq M \leq 700$ and $\frac{L}{M} < \rho$.

$0.1 \leq C_W \leq 100$ and $\rho > 0.25 = \frac{L}{M}$ with the parameter setting $L = 50$, $M = 200$, $K = 300$, $\mu = 1$.

Both figures show a very good match until approaching the ρ boundary $\frac{L}{M}$, where the condition 3, does not hold any more.

VII. Conclusion

In this paper, we proposed shifted N-policy for a simple, but energy efficient control of number of active VMs in the IaaS cloud. Besides of the stationary analysis of the underlying queueing model, we provided an approximate formula for computing the optimal threshold N, which minimizes the cloud provider's cost, in the most relevant parameter range. The validation of the approximate solution formula by means of numeric optimization shows a good match in the considered parameter range. The closed form approximate solution

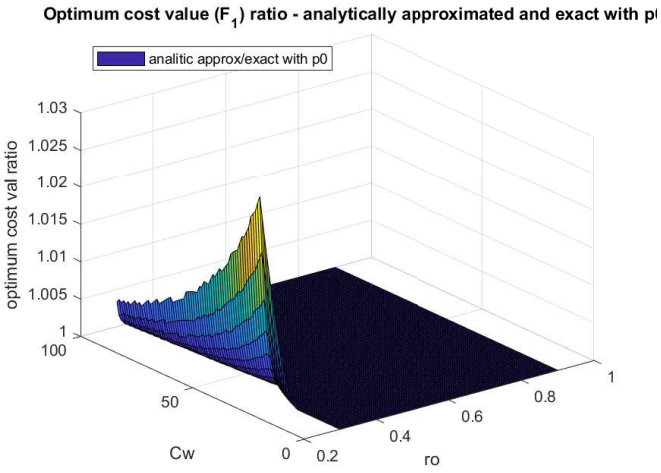

Figure 9. Ratio of the approximated and exact optimal value (F_1) for $0.1 \leq C_W \leq 100$ and $\frac{L}{M} = 0.25 < \rho$.

formula enables a simple management of the cloud and gives an insight into the dependency of the optimal threshold N on the model and cost parameters.

A future research work is to investigate an approximate solution also for the remaining parameter ranges not considered in this work. Another, more difficult future research topic is the joint optimization of parameters L and N.

References

[1] F. Durao, J. Fernando, S. Carvalho, A. Fonseka and V. C. Garcia, "A systematic review on cloud computing," J. Supercomput., vol. 68, no. 3, pp. 1321–1346, 2014.

[2] C. Chapman, W. Emmerich, F. G. Mrquez, S. Clayman and A. Galis, "Software architecture definition for on-demand cloud provisioning," Cluster Comput., vol. 15, no. 2, pp. 79–100, 2011.

[3] R. Ghosh, F. Longo, V.K. Naik, and K.S. Trivedi, "Modeling and performance analysis of large scale IaaS clouds Future Generation Computer Systems," Future Generation Computer Systems, vol. 29, pp. 1216–1234, 2013.

[4] Q. Duan, " Cloud service performance evaluation: status, challenges, and opportunities a survey from the system modeling perspective," Digital Communications and Networks, vol. 3, no. 2, pp. 101–111, 2017.

[5] F. Nzanywayingoma and Y. Yang, "Efficient resource management techniques in cloud computing environment: a review and discussion," International Journal of Computers and Applications, vol. 41, no. 3, pp. 165–182, 2019.

[6] T. Ma, Y. Chu, L. Zhao and O. Ankhbayar, "Resource Allocation and Scheduling in Cloud Computing: Policy and Algorithm," IETE TechnicalReview, vol. 31, no. 1, pp. 4–16, 2014.

[7] T. Tournaire, H. Castel-Taleb, E. Hyon, and T. Hoche, " Generating optimal thresholds in a hysteresis queue: application to a cloud model," MASCOTS 2019: 27th International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems, Rennes, France, Oct 2019, pp.283–294.

[8] B. Wan, J. Dang, Z. Li, H. Gong, F. Zhang and S. Oh, "Modeling Analysis and Cost-PerformanceRatio Optimization of Virtual Machine Scheduling in CloudComputing," IEEE Transactions on Parallel and DistributedSystems, vol. 31, no. 7, pp. 1518–1532, 2020.

[9] Y. Mansouri, A. N. Toosi and R. Buyya, "Cost Optimization for Dynamic Replication and Migration of Data in Cloud Data Centers," in IEEE Transactions on Cloud Computing, vol. 7, no. 3, pp. 705-718, 2019.

[10] W. Whitt, "Approximations for the GI/G/m queue," Production Oper. Management, vol. 2, pp. 114–161, 1993.

 ISBN: 978-1-61208-860-0