Scalability of IoT Systems: Do Execution Costs Predict the Quality of Service?

Ahmed Al-Qasmi[1], Huda Al Shuaily[1], Kennedy E. Ehimwenma[2(✉)], and Safiya Al Sharji[1]

[1] University of Technology and Applied Sciences, Al Khuwair 33, PC113 Muscat, Sultanate of Oman
{ahmed.alqasmi,huda.alshuaily,safiya.alsharji}@utas.edu.om
[2] College of Science and Technology, Wenzhou Kean University, Wenzhou, China
kehimwen@kean.edu

Abstract. Execution costs are broadly used in the evaluation of the scalability of IoT systems. A well-known concern in their use is the extent to which their scalability desiderata best predicts Quality of Service (QoS). At first, past studies did not ratify a relationship between the scalability approaches and QoS in IoT systems. More recently, however, the correlations between these have begun to emerge. In this paper, we extend those findings and open up new avenues to further research by proposing a statistical testing approach for scrutinizing this relationship. The initial findings delineate that there is a significant correlation between the scalability approach employed and QoS in IoT systems. Our results strengthen the use of execution costs in the scalability of IoT systems confirming that QoS can be successfully predicted.

Keywords: Execution costs · Quality of service · Scalability · System performance

1 Introduction

The evaluation of IoT systems has a well-established process in experimental design, with some currently common metrics including latency, packet loss indicators, jitter, bandwidth, and throughput, which are generally used for optimization of network performance [1]. While every organization is constantly seeking QoS when adopting its information systems; it can be presumed that the QoS measured by using these metrics at the default level of the system's configuration will predict the actual QoS in practice. The performance of IoT systems [2], referred to as QoS in this paper, can thus typically be quantified using measures derived from a number of computing services including cloud computing, be it IaaS (Infrastructure as a Service), PaaS (Platform as a Service) or SaaS (Software as a Service), which might all affect the scalability of the IoT systems.

The concept of scalability is crucial to ensure adequate IoT systems since such systems are characterized by an excessive increase in the number of connected things to

T. Pereira et al. (Eds.): IoECon 2022, LNICST 458, pp. 91–100, 2023.
https://doi.org/10.1007/978-3-031-25222-8_8

deal with billions of services. Scalability is defined as the measure of a system's ability [3, 4] to increase or decrease in its performance and cost with respect to changes in the applications and/or system processing demands. As such, it is a metric that expresses how a system's performance is maximized without any degradation in the QoS it provides when additional bandwidth is needed as a consequence of an increase in the load-handling capacity requirement. This will ensure a certain level of performance which might be quantitatively measured when evaluating the network QoS model. For this reason, one might presume that the overall goal of scalable solutions is thus to enhance the QoS.

There are actually two approaches to increase the load-handling capacity of IoT systems: the vertical scalability (or scale-up) approach and horizontal scalability (scale-out) approach. Scale-up solutions are normally characterized by a large symmetric multiprocessing system that shares memory in the same machine. This handles the whole load when it is increased, by using a more powerful computer. Scale-out solutions, on the other hand, are characterized by the usage of smaller clusters of machines. Each machine works with its own operating system. Decades ago, the vertical approach was the most dominant, prior to the horizontal approach becoming widespread due to high-throughput web applications [5]. The current study substantiates a relationship between these two approaches to identify which one is more suitable for the expansion of IoT systems.

Much research in the evaluation of IoT systems has focused on improving the above-mentioned measures, which often assume that, during the analysis of throughput against the bandwidth [3–6], any throughput that is significantly lower than the bandwidth is an indication of a poor network performance. However, this kind of evaluation has been criticized by a number of researchers [1, 15, 21, 22] because the characteristics of the measures used do not pertain to the scalability approaches of IoT systems. This paves the way for a new direction in the evaluation of IoT systems towards the use of execution costs as a measure in determining the relationship between the scale-up and scale-out approaches employed in IoT systems to quantitatively measure their performance. This is important because as [6] assert, the real issue in adopting IoT systems is not whether latency and packet loss indicators are high and low respectively by a statistically significant percentage, but rather whether the execution cost is relatively low in these systems.

This study aims to build upon, and expand, the existing knowledge as to whether or not execution costs are applicable for predicting the scalability of IoT systems and consequently their QoS in terms of performance. We describe a scenario of a smart educational institute equipped with smart surveillance cameras, smart energy management, smart lighting and smart water quality monitoring. No doubt this institute would require additional bandwidth on an as-needed and when-needed basis (e.g., to deploy a highly reliable ANPR solution for smart gate security); this means that the existing infrastructure has to be able to support any increasing number of connected devices, application features, and users, in other words, it must be able to dynamically scale as and when the network requires changes to its topology. Previous research [5, 7–9, 19] into scalability has been conducted, but these studies did not focus on the extent to which in such environments, scaling up (vertical) and scaling out (horizontal) approaches are reliable. However, it is important to adopt the appropriate scaling approach so that the overall performance is maximized without any degradation in the QoS. To the best of our

knowledge, this study is the first of its kind to identify the performance of IoT systems using the statistical variance of execution costs to compare the extent to which scale-up and scale-out approaches vary in scaling the IoT systems. This strengthens the use of execution costs in the evaluation of the scalability of IoT systems, confirming that the QoS of IoT systems can be successfully predicted based on the scalability approach adopted.

The remainder of this paper is structured as follows: in the following section, we describe the related literature on system scalability and QoS in IoT systems, while Sect. 3 elaborates on the correlation between the scaling approaches and QoS in IoT systems along with the details of our experimental setup. Section 4 discusses the results from our experiments, and Sect. 5 considers the implications of our findings for the evaluation of IoT systems. We present our proposal for future work and conclude the paper in Sect. 6.

2 Literature Review

The work found in the literature review regarding the most effective scalability approach that best predicts Quality of Service (QoS), is limited. An evaluation mechanism for service composition mechanisms was proposed in [9]; their research however, focused on the review and evaluation of these mechanisms which fall under the umbrella of scalability, but a major limitation of their approach lies in the lack of quality metrics for dynamic reconfiguration to add value to the quality of their functional scalability desiderata. The research study by [10] also provided a major contribution to this field. They used the MapReduce application, which allows the offloading of the computation and communication portions of a scale-up implementation, from the cores of a chip-multiprocessor to their accelerators. While they included a valuable contribution and demonstrated how their hardware-based solution was highly scalable, they did not provide the full range of optimized design aspects.

White et al. [11] conducted a comprehensive study related to QoS for IoT systems in which three different aspects of QoS are discussed: (1) the types of research conducted in the field; (2) the factors of quality being measured; (3) the layers of the IoT architecture. By providing a correlation between scalability approaches, the current study extends the last aspect of this study, in which only consolidated high-level quality characteristics are provided.

Recently, in the field of IoT systems, researchers have also shown an interest in QoS measurements. For instance, the work of Singh et al. [12] categorized QoS measurements by their quality characteristics as three main types including, the QoS of computing, the QoS of things, and the QoS of communication. However, despite the fact that the term metrics is employed in this work, their study focuses on quality characteristics rather than on the formulas which actually define QoS metrics. A few more metrics are discussed by these authors in [13] in the area of wireless sensor networks.

Finally, Staron et al. [14] examined the metrics of the Quality of System architecture including the number of interfaces, the number of coupled components and changes in the architecture per time unit; yet, this study provides a discussion of the metrics which are generally applied in the IoT environment rather than the quality of end devices. However, the research by both [5] and [15] delineated the results of their investigation

of the scalability approach that outperforms IoT systems scalability approaches, but the authors did not sufficiently elaborate their experimental set up for further research. In the current study, we have followed a different experimental approach which indicated that the scale-out approach yields a better result by reducing the cost of execution in the cloud; which is also consistent with the results found in both [5] and [15].

3 Methodology

In devising the methodology, we wanted to utilize the cloud execution costs metric to gauge the performance of the two scalability approaches for an IoT system based on a PaaS platform of our smart educational institute. Such an environment is inferred to have similar settings in all devices it is equipped with, including cloud computing required for the full deployment of the IoT system as specified by the cloud service provider. In past experiments, evaluation of IoT systems was performed by coding a simulation of that environment whereby various flexible and scalable java classes provided by the CloudSim package were employed.

However, the set-up costs of these experiments are costly in nature as they require specific code to be applied through each application program; to provide a more controlled experiment, it is preferable to use cloud simulators as they enable quantitative measurement of the systems' performance (i.e. a low execution cost) under different scenarios rather than merely measuring the instruction execution speed. For details about simulators, the reader is referred to [20], the scope of this study is to tackle the QoS problem from the overall perspective of an IoT system to compare the scale-up and scale-out approaches. As such, this scope is a delimitation of our study and we do not focus specifically on QoS metrics. Thus, the overview provided in this paper does not focus either on the quality metrics of specific code or on the criteria for general testing coverage.

3.1 Testing the Scalability Approach

The scale-out approach used in this study is conceptualized in eight different configurations in eight different locations (which we call scenario 1 for our scale-up approach) with one main configuration (scenario 2 for the scale-up approach) at the high-end of each configuration [15], without forcing all solutions into a single setup. We hypothesize that when additional bandwidth is required on an as-needed and when-needed basis, the scale-up or scale-out approach will provide the same QoS for the same specification setup of computing devices; this null hypothesis H_0, can then statistically be interpreted as:

"when an additional bandwidth is required in the network infrastructure, there is no significant difference in the mean of the execution costs with respect to the scalability approach adopted".

In this paper, we use simulation represented in the two scenarios mentioned above, which involve different aspects of the cloud (network, storage, memory, CPU) for our smart educational institute to validate its early stage evaluation rather than adopting real-world testbeds which are both more complex and more expensive [2]; Furthermore, we

use the default configurations of the system since these provide the data members which define the bandwidth, RAM, MIPS (million instructions per seconds) and the size of the architecture [1]. We then extract the execution costs in terms of the system's Response Time (RT) and the data center Processing Time (PT), from both the scale-up and scale-out approaches. With these measures, we were able to investigate the correlation between the scale-up and scale-out approaches. In this section, we outline the experimental set up in which we used the default configurations of our IoT systems provided by the service provider in comparing these approaches based on similar architecture.

3.2 Simulation Configurations

The simulated cloud computing for these experiments consists of different configurations equipped with several hosts in each configuration, where one large host representing the scale-up approach, is of greater capability than all other hosts, used for the scaling out approach. For each approach, if we name our eight different configurations as top1, top2, top3, top4, top5, top6, top7 and top8, assuming that each configuration is modelled with the large host, we can use this large host for testing the scale-up approach so that we do not compare apples with oranges. This allows each location to have eight different configurations top1-top8 for the scaling out approach, including one large host which is modelled from the total specifications of the scaling out approach [15].

CloudAnalyst [16][1], an open-source simulator – chosen because of its popularity and capability of simulating tasks requiring flexibility and sophisticated reconfigurability in addition to continuously allowing the experiments to be repeated - was then configured at each host. Since Facebook is a large scaled application that could benefit from the cloud, it was then used for our simulation as workload input, to scrutinize the behaviour of such an application under different deployment configurations. A table of the details of the parameter configurations of our experiment set up of the system (at 1/10th of the scale of Facebook) is elaborated in the appendix along with the distribution of the application by all users (collected on January 15th, 2022) at each user base (top1-top8). It is worth noting that the application was simulated during the peak hours[2] for a total of 60 h [18].

4 Results and Analysis

This section is divided into two parts: 1) the QoS based on the scalability (scale-up and scale-out) approach and 2) the analysis of the results from varying levels of the scalability approach.

4.1 QoS Using Scale-Up and Scale-Out Approach

Assuming that the above application is deployed in each host based on the configurations (top1 to top8) detailed in the appendix, the output results of the simulations in these

[1] https://www.opensourceforu.com/2016/11/best-open-source-cloud-computing-simulators/ Last Accessed on 15/Jan/2022.

[2] https://www.omantel.com/ Last Accessed on 10/Jan/2022.

series of experiments are shown in Table 1. Using a paired t-test, a statically significant difference in QoS (execution costs – both the response time in milliseconds – ms - and the processing time in milliseconds) between the two approaches is also illustrated in Table 2 and 3. From Table 1, it is clear that, using the scale-out approach, the execution costs were reduced as compared to the execution costs required in the scale-up approach. These results are further graphically depicted in Fig. 1. The spikes in both the response time and the processing time can be seen clearly with the highest points for the top1 configuration while the lowest points are mostly obtained with more powerful hardware configurations (top8).

Table 1. Execution costs in scale-up and scale-out approaches

	Scale-Up		Scale-Out	
	RT(ms)	PT(ms)	RT(ms)	PT(ms)
top1	274.48	82.47	220.20	40.29
top2	238.70	101.47	255.98	88.56
top3	173.35	48.15	155.05	39.32
top4	174.42	48.10	157.92	39.95
top5	120.24	23.45	101.74	15.65
top6	147.06	21.95	127.60	13.52
top7	113.62	22.62	96.22	12.65
top8	110.57	17.46	92.55	10.66

4.2 Differences Between Scale-Up and Scale-Out Approaches

The observed significant differences in the scalability approaches in IoT systems is probably due to the large difference - shown in Table 2 and 3 - in the execution costs between the scale-up and scale-out approaches. Here, we reduce the gap between the execution costs of scale-up and scale-out approaches to determine the impact on the system's performance (the QoS). The gap reduction is achieved by deducting [17] the execution costs values of the system using the scale-out approach from the scale-up approach (absolute difference), and then sorting the obtained results by the lowest differences in execution costs (Table 2 and 3 delineates the outputs results based on response time and processing time respectively). For the sake of consistency (with top6, see appendix), these outputs are normalized to equal the size set of users of the simulated application. The attribute StaSigDif used in Table 2 and 3, indicates whether or not there exists a statistically significant difference between the two scalability approaches. This allows us to detect when this significant difference in execution costs between scale-up and scale- out approaches disappear. We observe that both tables have significant differences (using a paired t-test), except in top5 for both RT and PT.

Table 2. Absolute difference in scaling approaches (RT)

	top1	top2	top3	top4	top5	top6	top7	top8
Scale-Up	163.45	234.70	124.59	144.63	85.76	147.06	91.77	91.82
Scale-Out	131.13	251.69	111.44	130.95	75.95	127.60	77.72	76.85
StaSigDif	0.00**	0.02*	0.00**	0.00**	0.13	0.03*	0.00**	0.02*

** $p < 0.01$, * $p < 0.05$

Table 3. Absolute difference in scaling approaches (PT)

	top1	top2	top3	top4	top5	top6	top7	top8
Scale-Up	49.11	99.77	34.61	39.88	17.51	21.95	18.27	14.58
Scale-Out	23.99	87.08	28.26	33.13	11.68	13.52	10.22	8.85
StaSigDif	0.01**	0.00**	0.00**	0.00**	0.15	0.04*	0.01**	0.02*

** $p < 0.01$, * $p < 0.05$

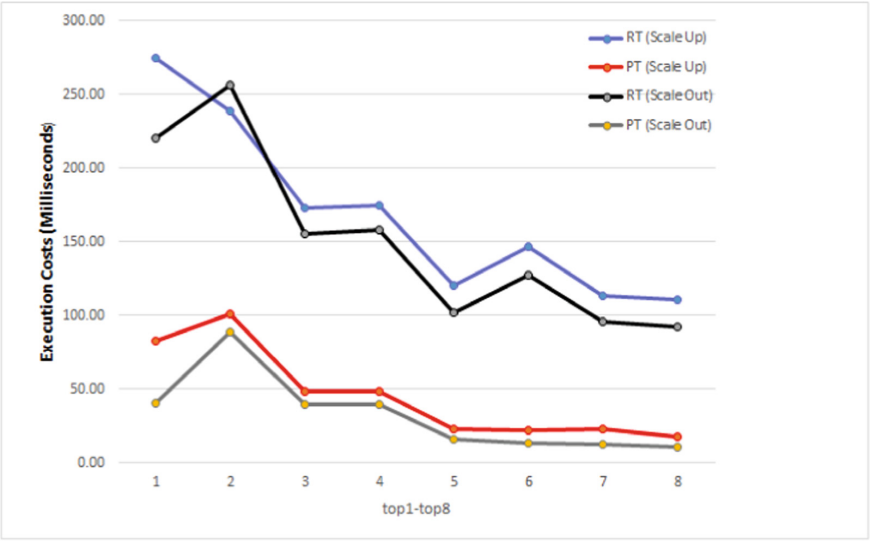

Fig. 1. Graphical representation of execution costs in scale-out and scale-up approaches

5 Discussion

Many research studies previously investigated the important problem of quality metrics (see Sect. 2) for dynamic reconfiguration of the IoT systems, which means reducing the execution costs in these systems. A few of these studies employed simulation approaches without using statistical variance of execution costs, and struggled to establish a significant correlation between the scalability approach employed and the QoS in IoT systems.

In a later study [15] some relationship problems started to be tackled. Following a different approach from earlier research, the current study has distinctively addressed the same issue and provided further evidence to analyze this phenomenon.

Moreover, as far as execution costs are concerned, it is normally expected that they are reduced using a large host (more powerful RAM) rather than a small host (less powerful RAM) and our findings demonstrated a consistency with this common fact. Nonetheless, our experiments showed that variations in simulations performed at different locations based on a large host modelled from the total specifications of smaller hosts, behaved differently and discerned differences between pairs of IoT systems having a tiny absolute difference in execution costs. In other words, costs are significantly ($p < 0.00$) reduced for the IoT scalability in the scale-out approach than in the scale-up approach, indicating a correlation between the scaling approaches and QoS in IoT systems. However, this correlation is reliable only if specific codes are deployed since scalability is context-dependent [15], as demonstrated in previous studies.

From our experiment, we also observe that with large hosts, this correlation is not always more effective, as shown in both top2 (RT and PT) and top6 (RT only) as compared to top1 (RT and PT) and top5 (RT only) respectively. The reason for the lack of correlation in this case is an issue to be addressed and we leave this as an open research question for further studies. We suggest further experiments with real-time monitoring and users' experience. Furthermore, we observe also that execution costs in the scale-out approach is always lower than the execution costs in scale-up approach when the absolute difference is larger than (0.01). This indicates that variations in execution costs can predict the IoT performance or the QoS in IoT systems.

A conclusion drawn from our experiments and results obtained is that execution costs can be used to measure the most suitable scalability approach in IoT systems when additional bandwidth is needed and can thus predict the QoS of an IoT system. Previous studies which failed to predict the scalability approach as the most suitable in the expansion of the IoT systems, may have conducted the simulation techniques in their experiments with inaccurate type of simulator without considering the most important layer – hardware layer - for dynamic reconfiguration of the system.

6 Conclusions and Further WorK

The application of statistical tests used in our experiment to the evaluation of our hypothesis, has allowed us to reject our null hypothesis; we conclude that for the expansion of the IoT systems, using the scale-up approach may not be as beneficial as using the scale-out approach. We demonstrated that the scale-out approach is slightly more effective compared to the scale-up approach. A critical analysis to identify the reason behind the effectiveness of the scale-out approach as compared to the scale-up approach is another open research question. Furthermore, despite using the CloudAnalyst simulator, we do not argue that these results are exhaustively replicable since we did not focus specifically on QoS metrics in our experiments. Thus, the overview provided in this paper focuses neither on quality metrics of specific code nor on the criteria of general test coverage, which could form part of future studies.

However, preference as to whether the scale-up or scale-out approach is well-suited for the scalability of a system is context-dependent, therefore, using these experiments

as a starting point, we believe that researchers can design more scalability metrics and more effective evaluation experiments. We have used the statistical variance of execution costs to compare the extent to which scale-up and scale-out approaches vary in scaling the IoT systems. Our study has implications for hardware resources and computing services for the scalability of the IoT systems. Our future work includes modelling framework tools for simulation rather than just using them for evaluation purposes, in addition to incorporating a failure handling mechanism into the simulation. We have learnt that simulation experiments have rich potential in identifying and experimenting with effective measures in the scalability of IoT systems. Unfortunately, due to time constraints, we did not extract the execution costs in terms of hosts utilization and power consumption, which could be beneficial metrics to measure the QoS in IoT systems.

Appendix

Default Configurations in Simulation.

Parameter	Value used
top1 (502,003 users)	4 GB total (from 1 small host modelled with 4 GB RAM)
top2 (304,050 users)	8 GB total (from 2 small hosts modelled with 4 GB RAM each)
top3 (415,952 users)	12 GB total (from 3 small hosts modelled with 4 GB RAM each)
top4 (360,534 users)	16 GB total (from 4 small hosts modelled with 4 GB RAM each)
top5 (400,453 users)	20 GB total (from 5 small hosts modelled with 4 GB RAM each)
top6 (298,952 users)	24 GB total (from 6 small hosts modelled with 4 GB RAM each)
top7 (370,119 users)	28 GB total (from 7 small hosts modelled with 4 GB RAM each)
top8 (360,017 users)	32 GB total (from 8 small hosts modelled with 4 GB RAM each)
Processing Speed	10000 MIPS (Host)
Transmission Rate	1.54 Mbps
Bandwidth (MB)	10000
Cloud latency	100 ms

References

1. Gross, T.R., Hennessy, J.L., Przybylski, S.A., Rowen, C.: Measurement and evaluation of the MIPS architecture and processor. ACM Trans. Comput. Syst. **6**(3), 229–257 (1988)
2. Jain, R.: The Art of Computer Systems Performance Analysis. John Wiley & Sons, Hoboken (2008)
3. Sun, X., Ansari, N.: Edge IoT: mobile edge computing for the internet of things. IEEE Commun. Mag. **54**(12), 22–29 (2016)
4. Li, L., Li, S., Zhao, S.: QoS-aware scheduling of services-oriented internet of things. IEEE Trans. Ind. Inform. **10**(2), 1497–1505 (2014)

5. Michael, M., Moreira, J.E., Shiloach, D., Wisniewski, R.W.: Scale-up x scale-out: a case study using nutch/lucene. In: IEEE International Parallel and Distributed Processing Symposium, pp. 1–8 (2007)
6. Taniuchi, Y.: On-demand virtual system service. Fujitsu Sci. Tech. J **46**(4), 420–426 (2010)
7. Misra, P.: Build a scalable platform for high-performance IoT applications. Technical report, TCS Experience Certainty (2016)
8. Sarkar, C., SN, A.U.N., Prasad, R.V., Rahim, A., Neisse, R., Baldini, G.:. DIAT: a scalable distributed architecture for IoT. IEEE Internet Things J.**2**(3), pp.230–239 (2014)
9. Arellanes, D., Lau, K.K.: Evaluating IoT service composition mechanisms for the scalability of IoT systems. Future Gener. Comput. Syst. **108**, 827–848 (2020)
10. Addisie, A., Bertacco, V.: Collaborative accelerators for in-memory mapreduce on scale-up machines. In: Proceedings of the 24th Asia and South Pacific Design Automation Conference, pp. 747–753 (2019)
11. White, G., Nallur, V., Clarke, S.: Quality of service approaches in IoT: a systematic mapping. J. Syst. Softw. **132**, 186–203 (2017)
12. Singh, M., Baranwal, G.: Quality of service (QOS) in internet of things. In: IEEE 3rd International Conference on Internet of Things: Smart Innovation and Usages (IoT-SIU), pp. 1–6 (2018)
13. Snigdh, I., Gupta, N.: Quality of service metrics in wireless sensor networks: a survey. J. Inst. Eng. (India): Series B **97**(1), 91–96 (2014). https://doi.org/10.1007/s40031-014-0160-6
14. Staron, M., Meding, W.: A portfolio of internal quality metrics for software architects. In: Winkler, D., Biffl, S., Bergsmann, J. (eds.) SWQD 2017. LNBIP, vol. 269, pp. 57–69. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-49421-0_5
15. Rahman, F.H., Au, T.W., Shah Newaz, S.H., Haji Suhaili, W.S.: A performance study of high-end fog and fog cluster in iFogSim. In: Omar, Saiful, Haji Suhaili, Wida Susanty, Phon-Amnuaisuk, Somnuk (eds.) CIIS 2018. AISC, vol. 888, pp. 87–96. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-03302-6_8
16. Wickremasinghe, B., Calheiros, R.N., Buyya, R.: CloudAnalyst: a cloudsim-based visual modeller for analysing cloud computing environments and applications. In: IEEE 24thInternational Conference on Advanced Information Networking and Applications, pp. 446–452 (2010)
17. Turpin, A., Hersh, W.: User interface effects in past batch versus user experiments. In: 25th Annual International Conference ACM SIGIR Conference on Research and Development in Informational Retrieval, pp. 431–434 (2002)
18. Jena, S.R., Ahmed, Z.: Response time minimization of different load balancing algorithms in cloud computing environments. Int. J. Comput. Appl. **69**(17), 22–27 (2013)
19. Luntovskyy, A., Globa, L.: Performance, reliability and scalability for IoT. In: IEEE International Conference on Information and Digital Technologies (IDT), pp. 316–321 (2019)
20. Bahwaireth, K., Tawalbeh, L., Benkhelifa, E., Jararweh, Y., Tawalbeh, M.A.: Experimental comparison of simulation tools for efficient cloud and mobile cloud computing applications. EURASIP J. Inf. Secur. **2016**(1), 1–14 (2016). https://doi.org/10.1186/s13635-016-0039-y
21. Karakus, M., Durresi, A.: A scalability metric for control planes in software defined networks (SDNs). In: IEEE 30th International Conference on Advanced Information Networking and Applications (AINA), pp. 282–289 (2016)
22. Lilja, D.J.: Measuring Computer Performance: A Practitioner's Guide. Cambridge University Press, Cambridge (2005)