2011 IEEE Ninth International Conference on Dependable, Autonomic and Secure Computing

Assessing Measurements of QoS for global Cloud Computing Services

Jens Myrup Pedersen, M. Tahir Riaz
Department of Electronic Systems
Aalborg University
Aalborg, Denmark
{jens,tahir}@es.aau.dk

Joaquim Celestino Júnior
Computer Networks & Security Laboratory (LARCES)
State University of Ceará (UECE),
Fortaleza, Ceará, Brazil
celestino@larces.uece.br

Bozydar Dubalski, Damian Ledzinski
Institute of Telecommunications
University of Technology & Life Sciences
Bydgoszcz, Poland
{dledzinski, dubalski}@utp.edu.pl

Ahmed Patel
Software Technology & Management Research Center
Faculty of Information Science & Technology,
University Kebangsaan Malaysia
UKM Bangi, Sengalor, Malaysia
whinchat2010@gmail.com

***Abstract*—Many global distributed cloud computing applications and services running over the Internet, between globally dispersed clients and servers, will require certain levels of Quality of Service (QoS) in order to deliver and give a sufficiently smooth user experience. This would be essential for real-time streaming multimedia applications like online gaming and watching movies on a pay as you use basis hosted in a cloud computing environment. However, guaranteeing or even predicting QoS in global and diverse networks supporting complex hosting of application services is a very challenging issue that needs a stepwise refinement approach to be solved as the technology of cloud computing matures. In this paper, we investigate if latency in terms of simple Ping measurements can be used as an indicator for other QoS parameters such as jitter and throughput. The experiments were carried out on a global scale, between servers placed in universities in Denmark, Poland, Brazil and Malaysia. The results show some correlation between latency and throughput, and between latency and jitter, even though the results are not completely consistent. As a side result, we were able to monitor the changes in QoS parameters during a number of 24-hour periods. This is also a first step towards defining QoS parameters to be included in Service Level Agreements for cloud computing in the foreseeable future.**

Keywords: Cloud computing, IT Infrastructure, Quality of Service, Service Level Agreements.

I. INTRODUCTION

Cloud Computing (CC) technology and its supporting services are currently regarded as an important trend towards future's distributed and pervasive computing services offered over the global Internet. Several architectures exist for CC, which differ in what kind of computing services are offered, culminating with the advances with Web 3.0 and Web 4.0 technologies [1]. Detailed studies of different approaches to CC can be found in [2][3]. To keep it simple, CC can be divided into two domains. The first domain consists of resources for computations and applications access, and their use by users – seen as traditional client server model. The second domain consists of network or more specifically the Internet, which enables computation for accessing and sharing computation and data resources for servers and clients in an array of complex arrangements [4]. Although the word "cloud" is a metaphor for Internet, based on depictions in computer network diagrams to abstract the complex infrastructure it conceals [5], CC is a general term for anything that involves delivering hosted services over the Internet such as Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS) and Software-as-a-Service (SaaS) [6]. Key factors of CC growth are increase in computation power, as well as in broadband network availability. Today, the use of CC is growing at an exponential rate [7]. Examples of these services are remote distributed and centralized data storage, remote offices such as Google Docs, Office Live, Cloud gaming services, virtual desktops, multimedia streaming, and grid computing [6][8]. It is widely predicted that Web 4.0 will create a myriad of possibilities for developing new applications within clouds needing QoS [1].

In order for clients to smoothly connect to services offered from servers and/or other clients located all over the globe, a specified Quality of Service (QoS) is often required, sometimes expressed in Service Level Agreements (SLAs). If these servers and clients are just connected through the Internet, it can be challenging to obtain a consistent service: The traffic is dynamically routed through different providers, from end-user access at the edge through distribution networks to national, international and even transcontinental backbones. This also makes it difficult to provide guarantees or even predictions of QoS since most often we do not have insight into the provider's networks and routes of global connections which are likely to change dynamically and continuously. Known behaviors, such as temporal changes in traffic amounts, may also be different in the different networks, making it difficult to come up with a simple

978-0-7695-4612-4/11 $26.00 © 2011 IEEE
DOI 10.1109/DASC.2011.120

prediction model. Moreover, different kinds of traffic may be prioritized based on e.g. packet sizes, protocols and source/destination addresses, adding to the complexity of modeling and predicting behaviors, or even continuously monitoring changes in QoS in a simple manner. Working in this uncontrolled environment also makes it hard to apply existing QoS techniques which focus on providing guarantees based on admission control [9][10].

In general, existing methods for measuring end-to-end QoS can be divided into two main classes: Active monitoring, where the measurements are based on traffic/packets/probes injected into the system, and passive monitoring, where measurements are based on observing existing traffic and studying e.g. throughput and response times for receiving acknowledgements. The advantage of passive monitoring is that it does not add any communication overhead. It makes good sense for some applications, e.g. video streaming [11], where it is possible to observe the parameters that are important for the application, but is less useful for others. This paper investigates if active monitoring based on only Ping packets can be used as an indicator for the most important QoS parameters, taking advantage of the active monitoring approach while keeping the overhead low.

The most commonly used QoS parameters include delay/latency, jitter, packet loss and bandwidth. Different applications have different QoS requirements: While some applications are tolerant to all parameters, and will do fine with whatever best-effort service is available, others will be critical with respect to one or two parameters, and others will be demanding in terms of more parameters. This criticalness will depend on the nature of the application, but as CC matures as technology, more time- and safety critical applications are expected to be developed.

While there is no strict relation between these parameters, there is reason to expect a certain correlation since common problems in the networks such as congestion and router/link failures can be assumed to affect all the parameters in a negative way. On the other hand, absolute correlations cannot be expected. For example, the fact that link capacities are limited does not imply congestion, and so it is obvious that we can experience good delay/jitter/packet loss performance even with a limited bandwidth.

In this paper, we present a practical investigation of the hypothesis: Is there a relation between the changes in delay and other QoS parameters between machines in the global Internet. Moreover, we present measurements done throughout 24 continuous hours between different network points, giving an idea of how the QoS in the current Internet changes within a one day time frame.

The main contribution of the paper is testing the hypothesis against real-life operations and the practical results, where we measure how different QoS parameters change over time in different long-distance networks. We show that smaller increases in latency often also leads to longer file transfer times, even though the results are not fully consistent, and the opposite relation was not observed in our results. This knowledge is important to take into account when designing cloud services and architectures and when defining SLAs: Simply measuring Ping throughout a session can give a warning that network conditions are worsening or improving, triggering a check against the SLA whether to escalate alerts to the service provider to either optimize the service or seek reimbursements for failure to deliver against the SLA.

The paper is organized as follows. After the introduction, Section II presents relevant background and definitions. Section III presents the methods and test setup applied, leading to the results in Section IV. Section V presents conclusion and further works.

II. BACKGROUND AND DEFINITIONS

The classical QoS parameters are bandwidth/throughput, delay, jitter and packet loss. Since the work in this paper mainly deals with high-capacity connections between universities, the focus is on the following three parameters, all relevant to different real-time and non-real-time CC services:

Throughput: Measured as the average maximum data rate from sender to receiver when transmitting a file. Thus, in our terms throughput is measured one direction at a time. For the experiments in this paper, it is measured by the time it takes to transmit a file of a certain length.

Delay: Measured as the round trip time for packets, simply using the standard Ping command.

Jitter: Measured based on variation in delay. This paper bases the measurements on the Ping packets as described above, and adopts the definition from RFC 1889 [12]: $J=J'+(|D(i-1,i)|-J')/16$. So the jitter J is calculated continuously every time a Ping packet is received, based on the previous jitter value, J', and the value of $|D(i,j)|$ which is the difference in Ping times between the *i'th* and *j'th* packets.

Packet loss is not considered during this study, since it is expected to be statistically insignificant (the assumption was actually confirmed during the experiments).

In order to be able to compare trends, it is necessary to smooth the observed values of throughput and delay. This is explained in details in Section III.

III. METHODS

A. Description of experiments

The idea is to measure how the different QoS parameters change over a 24 hour period, between different geographically dispersed clients connected to the Internet. In particular, both inter-European and trans-continental connections are studied. For this reason, a setup is established with a main measurement node at Aalborg University, Denmark (AAU), and other measurement stations at remote universities in a) Bydgoszcz, Poland, b) UKM Bangi, Malaysia, and c) Fortaleza, Brazil. Between AAU and each of the measurement stations, experiments are conducted over 24 hour periods, where latency (Ping packets sent from AAU), jitter (derived from the Ping measurements) and throughput (based on FTP transfers from AAU to the measurement stations) were continuously measured. The Ping packets were sent in 10 seconds intervals, whereas the

FTP transfer was done by transmitting a 20 MB file from AAU to the measurement stations every 5 minutes.

The smoothing function mentioned in Section II was chosen to avoid small variations (e.g. due to OS scheduling) destroying the bigger picture, while on the other hand the smoothening intervals were not so long to either distort or obviate the actual trends being monitored. It was chosen to show Ping and throughput in moving non-weighted averages over 10 minute intervals. This means that the throughput for measurement *t* is given by the average of the measurements *t-1*, *t* and *t+1*. For the latency, it is the average of the Ping values measured within 5 minutes before and after the actual measurement time. The jitter is by definition a moving average, and no additional smoothing has been applied.

For each destination, the measurements were carried out for two 24-hour intervals, in order to confirm the validity of the results. Due to technical problems only one measurement was done for the server in Malaysia.

B. *Test Setup*

The test setup consists of 4 dedicated PCs running Ubuntu OS, connected to the Internet, and assigned public IP addresses. All the PCs were running FTP servers, and also having Ping enabled. The python scripting was used in order to implement the proposed measurement methodology. Beside that MySQL DB was used for storing the results. The AAU location was selected for the primary execution of the scripts. Both Ping and FTP were accessed in system terminal through python's "subprocess" module. The output from Ping and FTP were then parsed using Phython's "re" module. In order to avoid any bottleneck etc., only one simultaneous test to any given remote server was performed. A high level flow-diagram is shown in Figure 1.

IV. RESULTS

First the possible correlation between latency and throughput is investigated, based on the results shown in Figures 2-6. In order to be able to observe temporal behaviors, the measurement values are arranged from midnight to midnight, even though the actual experiments are starting at different times as listed in the figure captions.

It is hard to find a consistent correlation between the two parameters, and in many places the parameters seem to change independently of each other. This is for example the case for the "spike" of increase in latency in Figure 6, shown in more details in Figure 7. The latency increases significantly for a while, but this is not matched by an increase in file transfer times. In other of the figures there appear to be a relationship, where an increase in latency also results in increased file transfer times. This is most visible where the latency is quite stable over time, small spikes in latency are matched by spikes in throughput. This is clearly visible in Figures 2, 4 and 8, where the latter shows a more detailed view of the first part of Figure 2, and can also be seen in other figures. The tendency was confirmed also by studying more of the experiments closer. The opposite does not seem to hold: file transfer time seems to vary even when the latency remains constant, and when latency increases this is not necessarily reflected in the file transfer times.

What can also be observed from these figures is that there are no consistent variations over the 24 hour periods. The variations are generally locally varying over time, with some rather dramatic changes, for which we do not know the reasons. For the Polish results (Figures 3-4) there could be a relation, where file transfer times and to some extent latency increase during working hours, but it is hard to tell, and the patterns are not really similar for the two days.

Calculating correlation coefficients for the relationship between Ping and latency did not lead to conclusive results. The strongest correlation was found in the first experiment between AAU and Poland, where the correlation coefficient was 0.49. For the other experiments the values were 0.22 (Poland), -0.35 and 0.29 (Brazil) and 0.37 (Malaysia).

Next the relationship between latency and jitter is studied. At a first glance, there is a close dependency between latency and jitter, where the spikes in latency is followed also by spikes in jitter. See Figures 9-14. Some relationship was also confirmed by the correlation coefficients, which were respectively 0.69 and 0.33 (Poland), 0.00 and 0.47 (Brazil) and 0.58 (Malaysia).

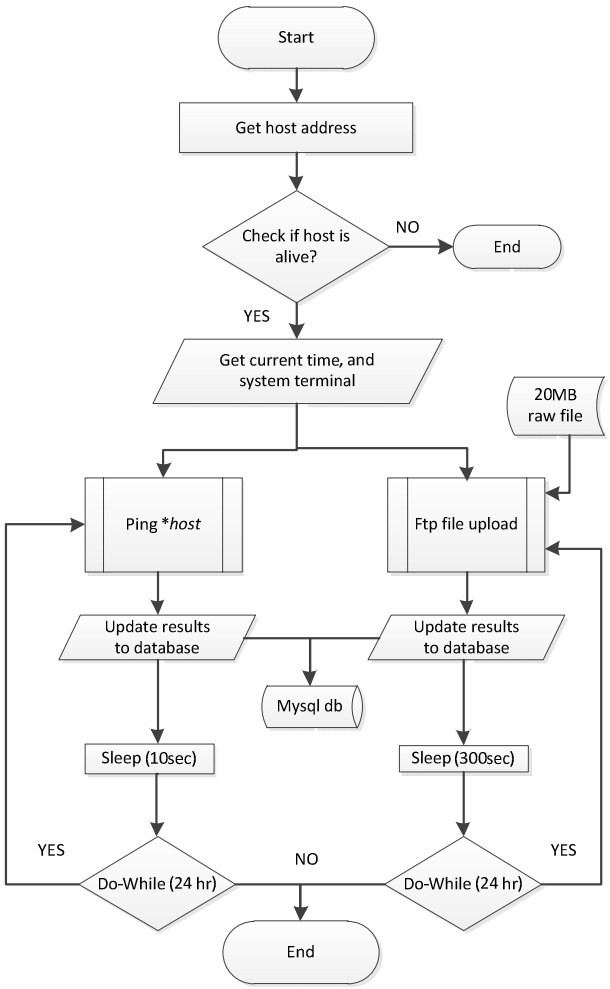

Figure 1: Flow diagram of the experimental setup.

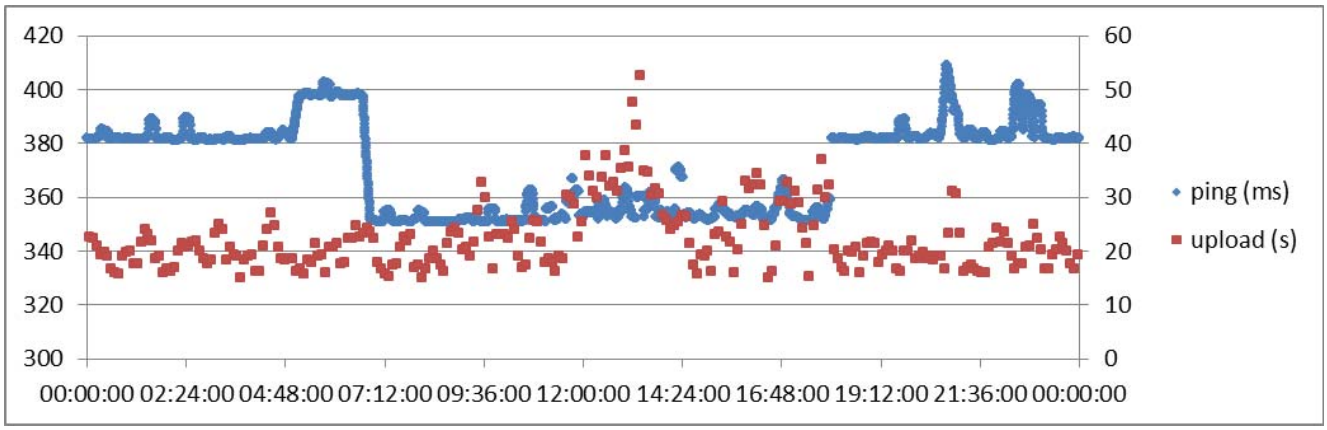

Figure 2: Latency (Ping, left scale) and throughput (upload times, right scale) for the first experiment between AAU and the server in Brazil. The experiments were carried out between 18:00 and 18:00 (Danish time, UTC+1).

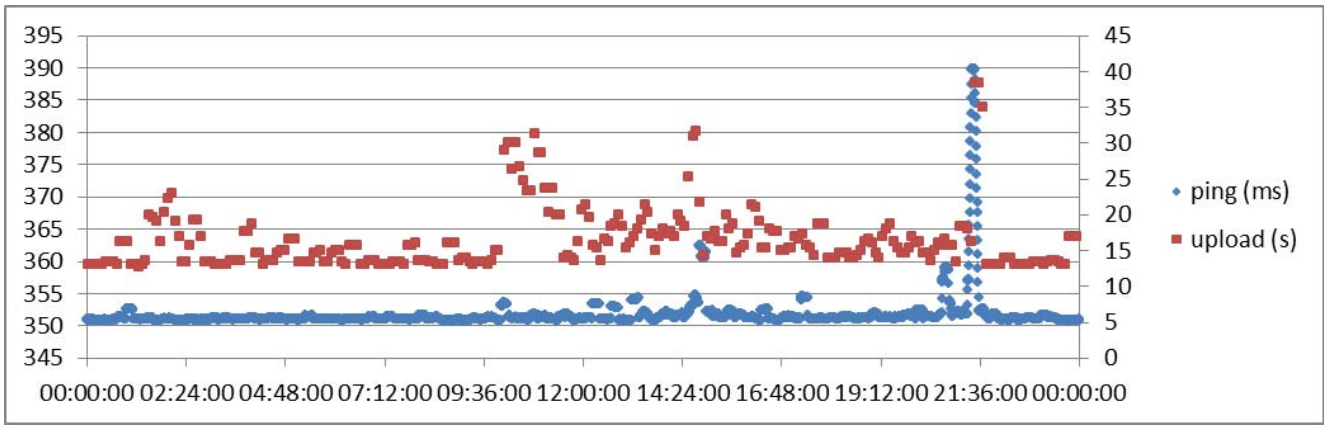

Figure 3: Latency (Ping, left scale) and throughput (upload times, right scale) for the second experiment between AAU and the server in Brazil. The experiments were carried out between 10:00 and 10:00 (Danish time, UTC+1).

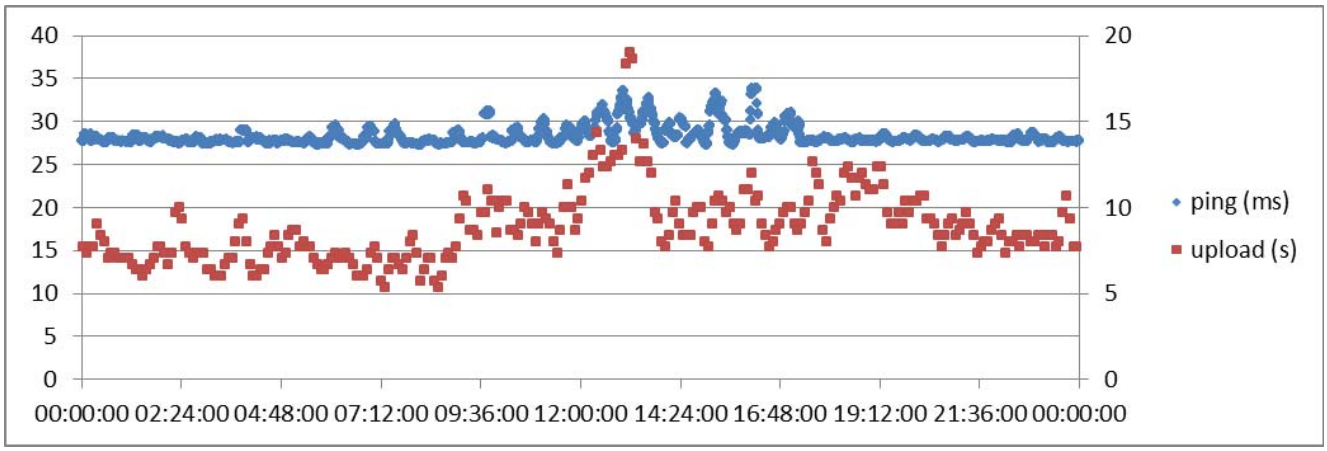

Figure 4: Latency (Ping, left scale) and throughput (upload times, right scale) for the first experiment between AAU and the server in Poland. The experiments were carried out between 11:40 and 11:40 (Danish time, UTC+1).

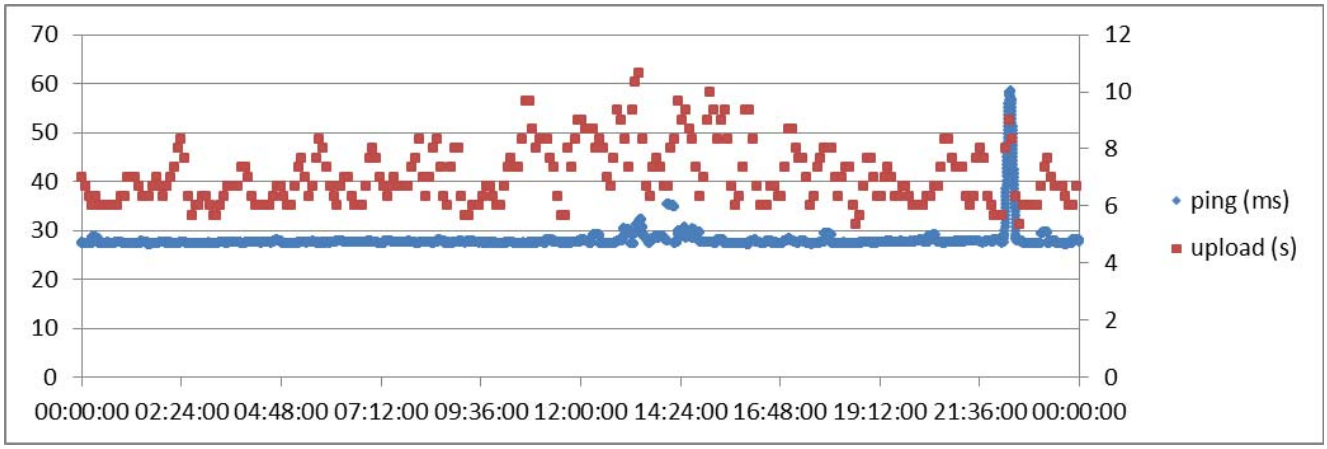

Figure 5: Latency (Ping, left scale) and throughput (upload times, right scale) for the second experiment between AAU and the server in Poland. The experiments were carried out between 10:00 and 10:00 (Danish time, UTC+1).

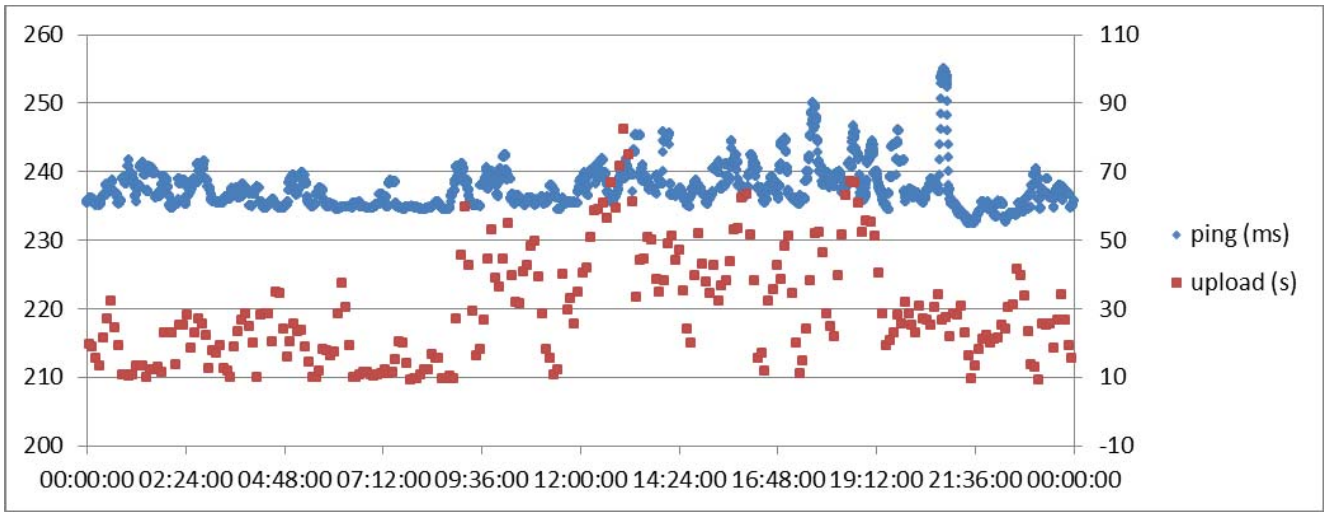

Figure 6: Latency (Ping, left scale) and throughput (upload times, right scale) for the experiment between AAU and the server in Malaysia. The experiments were carried out between 11:30 and 11:30 (Danish time, UTC+1).

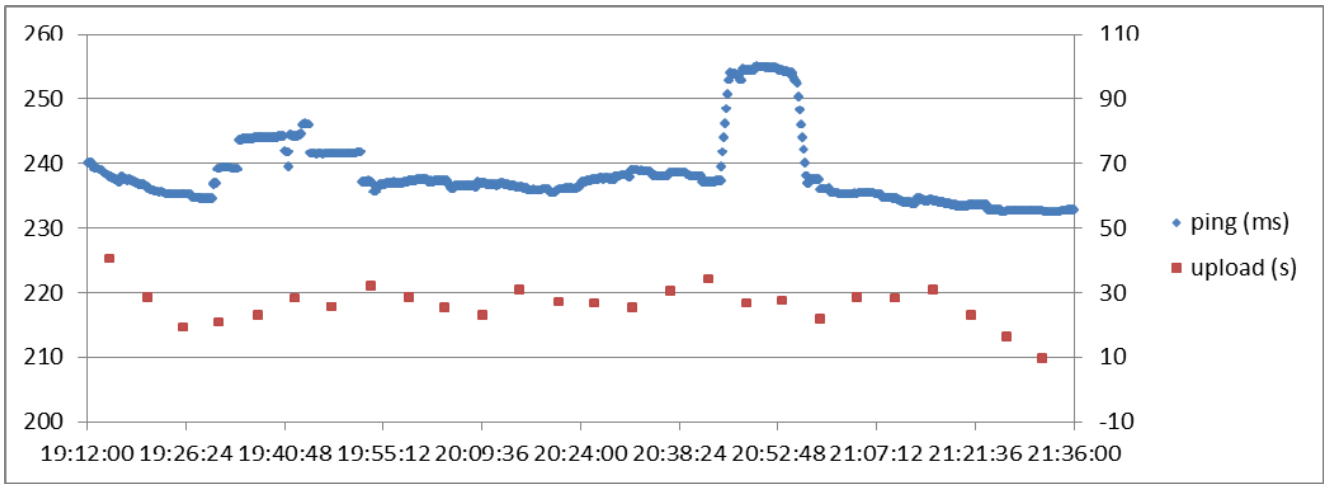

Figure 7: The same results as shown in Figure 6, but showing only the time from 19:12 to 21:36, and thus including both a smaller and larger spike. There seems to be no significant correlation between the Ping and upload times in this figure.

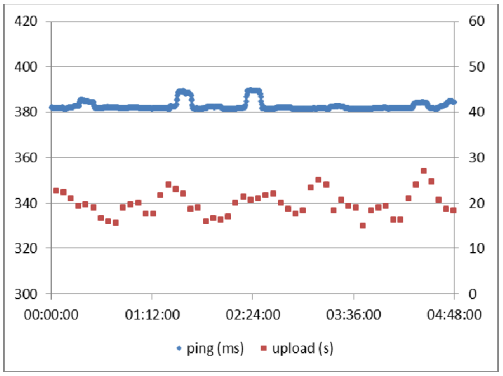

Figure 8: The results from Figure 2, for the first 4:48 hours. The upload times seem to increase during the latency spikes.

This is not surprising as the jitter is by definition expressing changes in latency, so the correlation could be due to the spiky nature of variations. An interesting observation is that when latency increases and stabilizes for a while, the jitter seem to fall back to the previous levels, adding to the difficulty in establishing a clear relationship.

V. CONCLUSION

Predicting network QoS is essential when choosing how (and where) to run services in the cloud. This paper investigated if it is possible to use latency as an indicator for the other QoS parameters throughput and jitter.

Experiments were carried out between servers in globally spread locations (Denmark, Poland, Malaysia and Brazil). Based on these it was not possible to find any fully consistent relationships between the parameters. However, it seems that in many cases smaller spikes in latency is correlated to spikes also in file transfer times. The results were not fully consistent though, and several cases we observed changes in throughput without corresponding changes in latency. Observing the correlation coefficients did not bring fully conclusive results either. All in all, there is probably some correlation between changes in latency and changes in throughput, and to some extent changes in latency can be used to predict that throughput may also change. Our observations also show some relationship between latency and jitter. This seems to be partly due to the spiky nature of latency: when latency increases for a short while and decreases again, this leads to higher jitter during this peak.

The results obtained are important to keep in mind when designing cloud services and/or cloud architectures, as well as defining QoS requirements and related SLAs, where it might be necessary to constantly monitor all relevant QoS parameters in order to be able to act upon network changes – for example by adapting or moving services. For other services, it might be sufficient to simply send Ping packets with certain intervals, to check if things appear to be stable.

For future research it could be interesting to collect larger amounts of experimental data, which would also make it possible to use a more analytical approach when looking for patterns. It could also be interesting to check whether responsive times for TCP acknowledgements could be used instead of Ping, allowing for passive monitoring.

Future research should also investigate the smoothing function(s). As most of the changes in latency seem to be rather short, it might give a more clear correlation between latency and other parameters to smoothen less and/or over shorter time intervals. This would require a higher sampling rate of file transfers, creating more load on the network during the experiments and potentially affecting latency measurements. The same problem occurs if file sizes are increased to decrease the inaccuracy created by small variations in the times it take to e.g. initiate each transfer.

ACKNOWLEDGMENTS

We would like to thank the staff at UKM Computer Centre and FTSM Technical Support, Malaysia, and Rudy Matela, UECE, Brazil for helping with the experiments.

REFERENCES

[1] L. Na, A. Patel, R. Latih, C. Wills, Z. Shukur, R. Mulla, “A study of mashup as a software application development technique with examples from an end-user programming perspective”. Journal of Computer Science 6 (11), November 2010.

[2] I. Foster, Y. Zhao, I Raicu, S. Lu, “Cloud Computing and Grid Computing 360-Degree Compared”. In: Grid Computing Environments Workshop, 2008. GCE'08, pp. 1–10.

[3] Qi Zhang, Lu Cheng, Raouf Boutab, “Cloud computing: state-of-the-art and research challenges”. Journal of Internet Services and Applications, Volume 1, Number 1, pp. 7-18

[4] G. Wang, T. S. Eugene Ng, “Understanding Network Communication Performance in Virtualized Cloud”. IEEE Multimedia Communication Technical Committee E-Letter, April 2011.

[5] A. Marinos, G. Briscoe, “Community cloud computing”. In: First International Conference Cloud Computing, CloudCom, volume 5931 of Lecture Notes in Computer Science, pages 472–484.

[6] Ahmed Patel, Ali Seyfi, Yiqi Tew, Ayman Jaradat, “Comparative study and review of grid, cloud, utility computing and software as a service for use by libraries”. Library Hi Tech News, Vol. 28 Iss: 3..

[7] T. Rings, G. Caryer, J. Gallop, J. Grabowski, T. Kovacikova, S. Schulz, I. Stokes-Rees, “Grid and Cloud Computing: Opportunities for Integration with the Next Generation Network”. Journal of Grid Computing, Volume 7, Number 3.

[8] M. Armbrust, A. Fox, R. Griffith, A. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, M. Zaharia, “Above the Clouds: A Berkeley view of Cloud Computing”. University of California, Berkeley,2009. [Online]. Available: http://d1smfj0g31qzek.cloudfront.net/abovetheclouds.pdf.

[9] O. T. Brewer, A. Ayyagari, “Comparison and Analysis of Measurement and Parameter Based Admission Control Methods for Quality of Service (QoS) provisioning”. Proceedings of the 2010 Military Communications Conference, IEEE 2010.

[10] S. Y. Ban, J. K. Choi, H.-S. Kim, “Efficient End-to-End QoS Mechanism Using Egress Node Resource Prediction in NGN Network”. In Proceedings of ICACT 2006, IEEE 2006.

[11] L. Lehikoinrn, T. Räty, “Monitoring End-to-End Quality of Service in Video Streaming System”. Proceedings of the 8th International Conference on Computer and Information Science, 2009.

[12] RFC 1889: “RTP: A Transport Protocol for Real-Time Applications”. The Internet Engineering Task Force (IETF), 1996. http://www.ietf.org/rfc/rfc1889.txt

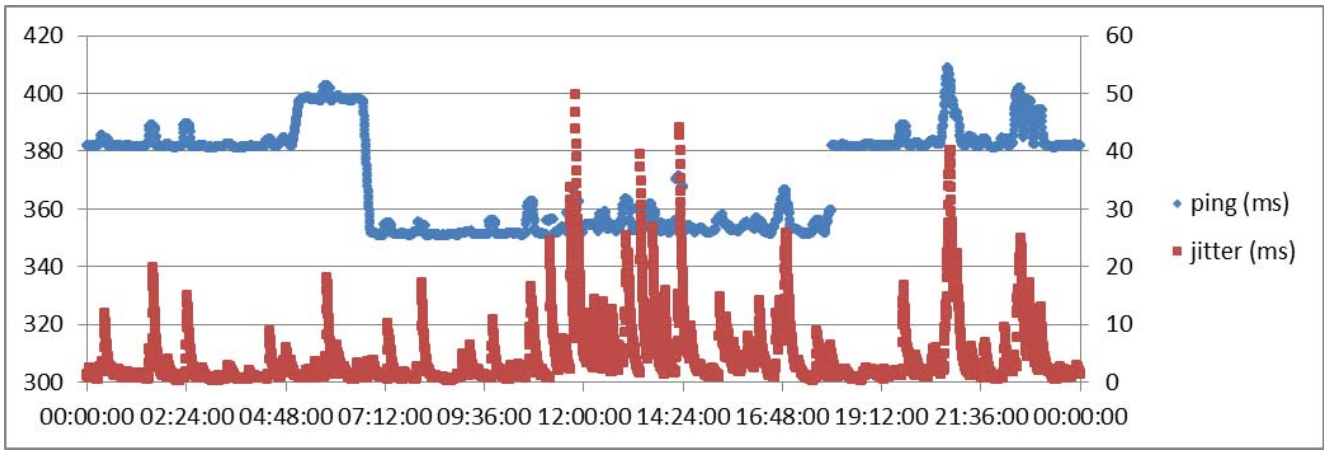

Figure 9: Latency (Ping, left scale) and jitter (right scale) for the first experiment between AAU and the server in Brazil. The experiments were carried out between 18:00 and 18:00 (Danish time, UTC+1).

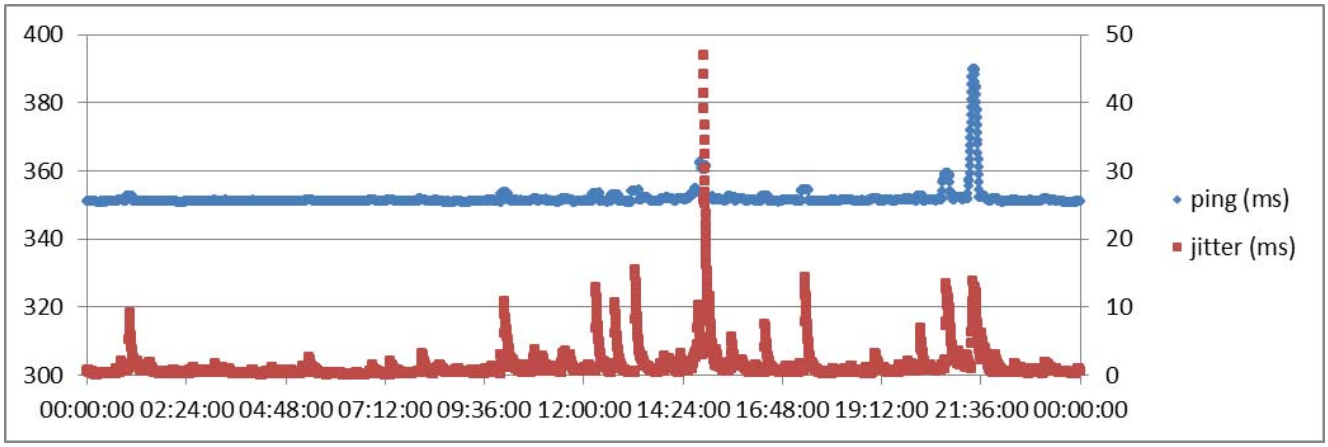

Figure 10: Latency (Ping, left scale) and jitter (right scale) for the second experiment between AAU and the server in Brazil. The experiments were carried out between 18:00 and 18:00 (Danish time, UTC+1).

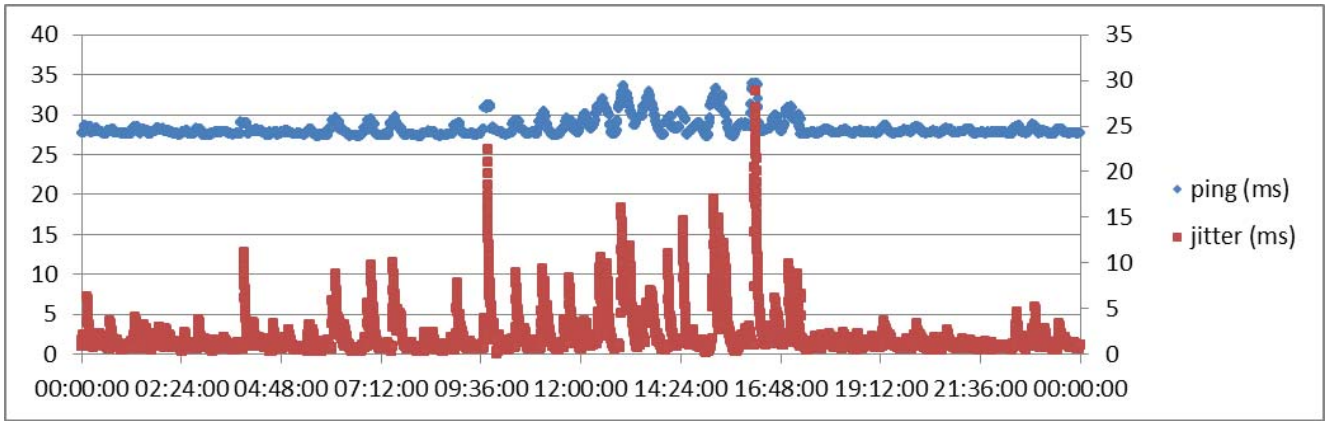

Figure 11: Latency (Ping, left scale) and jitter (right scale) for the first experiment between AAU and the server in Poland. The experiments were carried out between 11:40 and 11:40 (Danish time, UTC+1).

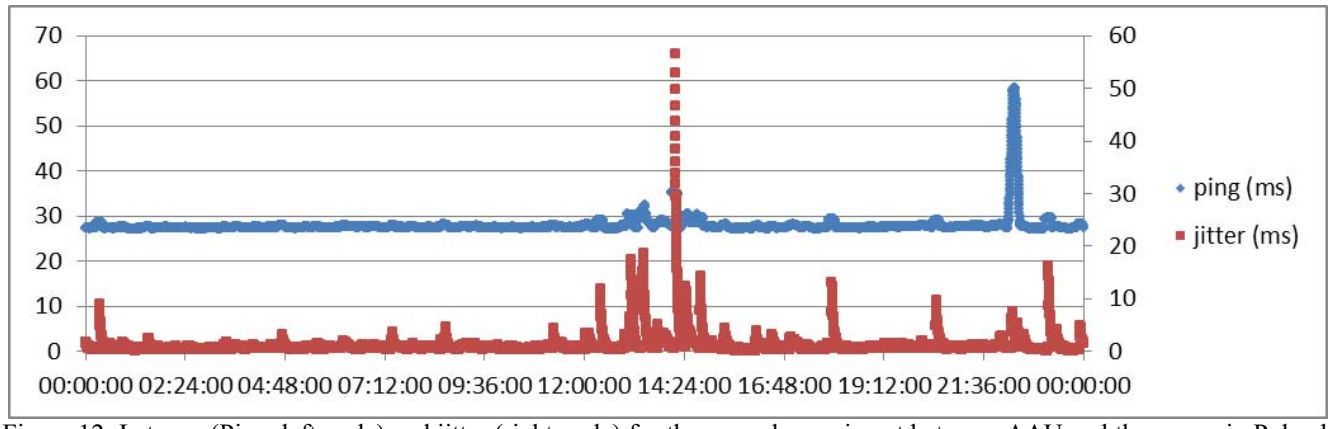

Figure 12: Latency (Ping, left scale) and jitter (right scale) for the second experiment between AAU and the server in Poland. The experiments were carried out between 10:00 and 10:00 (Danish time, UTC+1).

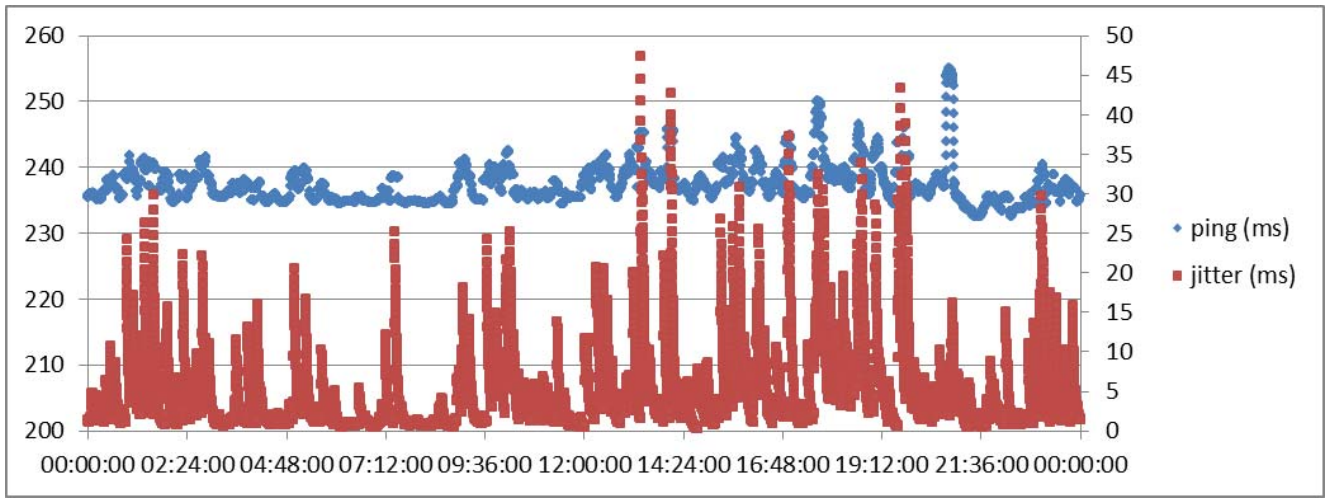

Figure 13: Latency (Ping, left scale) and jitter right scale) for the experiment between AAU and the server in Malaysia. The experiments were carried out between 11:30 and 11:30 (Danish time, UTC+1).

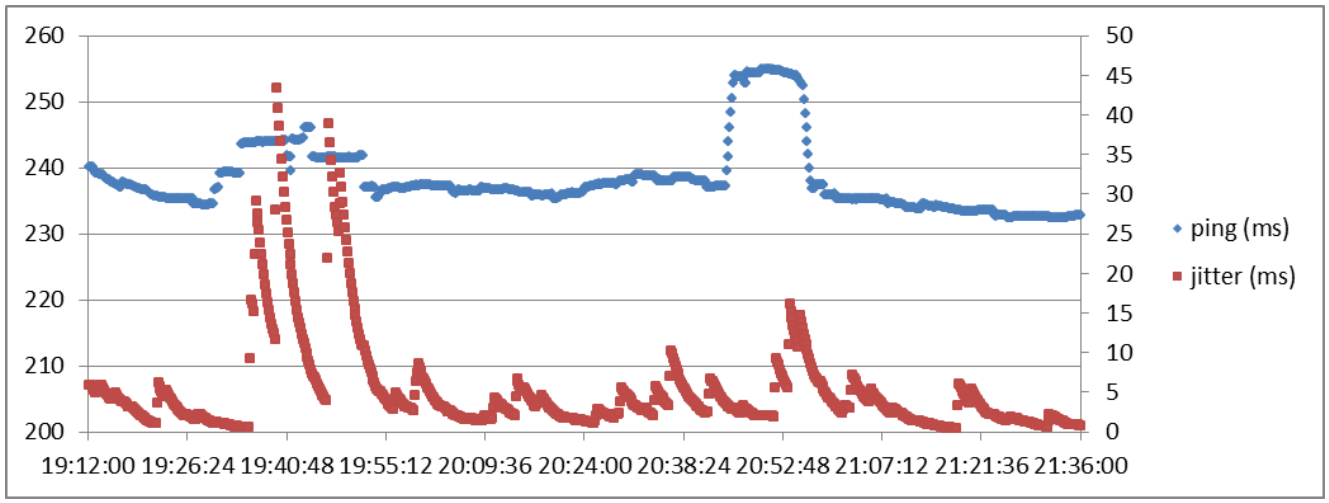

Figure 14: The same results as shown in Figure 13, but showing only the time from 19:12 to 21:36, and thus including both a smaller and larger spike in Ping times.