ISSN 0511-5728
The West Indian Journal of Engineering
Vol.40, No.2, January/February 2018, pp.32-41

Performance Measurement of Broadband Connections: An Enhanced Tool

Rishi Latchmepersad [a], and Tricia Ragoobar-Prescod [b,Ψ]

Department of Electrical and Computer Engineering, Faculty of Engineering, The University of the West Indies, St. Augustine, Trinidad and Tobago, West Indies;

[a]E-mail: rishi.latchmepersad@my.uwi.edu
[b]E-mail: tricia.ragoobar-prescod@sta.uwi.edu

[Ψ] *Corresponding Author*

(Received 30 June 2017; Revised 05 December 2017; Accepted 05 January 2018)

Abstract: *The performance of broadband Internet connections is an important qualifier in discussions about user experience, Internet Service Provider (ISP) assessments and national economic status. Performance can be quantified in a number of ways, including speed, latency, jitter and packet loss. While several tools have been developed to measure these parameters, many of these tools measure only one metric and offer limited flexibility for user configurations, such as single broadband connections and fixed test times. This paper discusses the development of enhanced algorithms and, correspondingly, a software application for the performance measurement of broadband Internet connections. The enhanced tool, TINQA (Totally Integrated Network Quality Application), is a native Windows® application, developed using the C# programming language and the .NET framework, and measures speed, latency, jitter and packet loss. The algorithms used to measure latency, jitter and packet loss were based on the employment of Windows® Raw Sockets and the Internet Control Message Protocol (ICMP), while the algorithm used to measure speed was based on downloading and uploading relatively large files (>250 MB) from and to several public File Transfer Protocol (FTP) and Hypertext Transfer Protocol (HTTP) speed test servers. TINQA produced results similar to those obtained from some of the most popular existent performance testing tools, including speedtest.net, testmy.net and pingtest.net. Additionally, the results were consistent across multiple tests, indicating that the algorithms were robust and that the added flexibility in testing did not compromise the accuracy of the tests in the application.*

Keywords: *Broadband Performance, Network Quality, Speed, Latency, Jitter, Packet Loss*

1. Introduction

The quality of broadband Internet connections can not only significantly affect individual users' experiences with web-based applications, but also, on a larger scale, influence a country's economic status. Ericsson (2013) found that increases in broadband speeds could drive increases in Gross Domestic Product (GDP), allow for better access to social services (such as improved healthcare) and promote improvements in energy efficiency within a country. At the other end of the spectrum, Claypool and Claypool (2010) investigated the impact of broadband quality on individual users, and found specifically that delays in the arrival of data packets could severely degrade the performance of online games, thereby ruining the experience for players. Additionally, Apteligent (2015) found that 48% of their surveyed users would uninstall or stop using an application if it regularly ran slowly (suffered from high latencies).

These findings demonstrate that various parameters can influence and can be used to measure the performance of web-based applications, and consequently show that a definite need exists to quantitatively characterise the quality of broadband connections according to key performance indicators.

Beuran et al. (2003) showed that speed, latency, jitter and packet loss all correlated directly to user-perceived quality (UPQ) for web-based services such as file transfers and Voice over Internet Protocol (VoIP) calls, while Sugeng et al. (2015) stated that these same four parameters can have a significant effect on applications such as video streaming. Ookla (2010) agreed with both these authors, stating that speed, latency, jitter and packet loss are widely accepted as the fundamental attributes necessary for a quality Internet experience.

At the present time, several tools exist to measure these four metrics. However, the current tools individually focus on a subset of the metrics, thereby resulting in the need to orchestrate multiple tests in order to attain an overall quality characterisation for a given broadband connection. For example, Ookla allows users to measure download and upload speeds at their speedtest.net website, while latency, jitter and packet loss tests are facilitated at their pingtest.net website (Ookla, 2010). Additionally, most of these tools lack functionality for automatic, long-term testing. Such testing allows for higher-level analyses of broadband connections, such as performance with respect to time, network architecture, and Internet Service Provider

(ISP), as illustrated by the Federal Communications Commission (FCC, 2016) in their annual report on fixed-broadband performance.

With this in mind, this paper describes the development of an application that was created to fill the aforementioned gap by providing an improved testing platform. This platform not only facilitates measurement of some of the most critical metrics of a broadband connection from a single application, but also allows for much greater flexibility in the test configurations as well as in the presentation of results.

More specifically, TINQA draws from the best measurement techniques that are implemented in the existing tools. The remainder of this paper is structured as follows: the following section defines the four key broadband performance metrics as they are used in this paper. Subsequently, the paper examines the most ubiquitous tools and methods that currently exist, feeding into the discussion that follows on the development of TINQA. A comparative, empirical analysis of the existing tools and TINQA is then rendered before the conclusion, which summarises the key features of TINQA and potential improvements, is presented.

2. Defining broadband performance metrics

The significance of establishing definitions for the broadband performance metrics lies with the need to ensure that comparisons are made on a like-for-like basis and that the user is aware of what is being measured. This section presents the background to the development of the definitions for each of the measurement metrics, in turn, under study. It begins by outlining the standard IETF definitions and then describing how definitions were developed for use in this research, for each parameter.

2.1 Speed

In their discussion on broadband speed measurements, Beur et al. (2010) stated that capacity, available bandwidth and bulk transfer capacity were three of the most popular references for speed in the context of broadband connections. Chimento and Ishac (2008) discussed capacity and available bandwidth in depth in the IETF's RFC 5136. In their discussions, they defined Internet Protocol (IP) layer link capacity, *C*, as the maximum number of IP-layer bits that can be transmitted from the source *S* and correctly received by the destination *D* over the link *L*, during the interval [*t*, *t+i*], divided by *i*, where i refers to the interval over which the transmission occurs. Building upon this definition, they provided the following mathematical description of available bandwidth, *AvailB*.

$$AvailB = C * (1 - Util)$$

where *Util* refers the link utilisation, which is in turn defined as the fraction of the capacity that is being used and is a value between 0 (that is, nothing is used) and 1 (that is, the link is fully saturated). This relationship between capacity and available bandwidth/capacity is illustrated in Figure 1.

Mathis and Allman (2001) discussed bulk transfer capacity in the IEFT's RFC 3148, where the term was defined as a measure of a network's ability to transfer significant quantities of data with a single congestion-aware transport connection (e.g. TCP). Of these definitions for capacity, available bandwidth and bulk transfer capacity, Beur et al. (2010) state that bulk transfer capacity, as described above, is the parameter that is most often reported by research in this area and most often the focus of broadband quality tests. Strauss and Kaashoek (n.d.) also discuss these terms and similarly conclude that bulk transfer capacity is the most relevant, as it allows for accurate estimates of TCP application performance over a network.

Although Mathis and Allman (2001) in RFC 3148 state that bulk transfer capacity must be defined in terms of a single connection, Beur et al. (2010) show that multiple connections are required for the most accurate speed measurements. The latter also show that many of the most popular performance measurement tools utilise multiple connections. With these considerations in mind, in this research, speed is defined using a modified form of the IETF's standard definition of bulk transfer capacity in RFC 3148. Namely, speed is defined as a measure of the amount of data which can be transferred along a network path in a particular interval of time, using multiple connections and a congestion-aware protocol such as TCP.

Figure 1. Illustration of the relationship between (total) capacity, utilised capacity and available bandwidth (capacity)
Source: Abstracted from AppNeta (2012)

2.2 Latency

Kwon (2015) explains that packets experience delays at each node in their movement from one node to another. Four types of delays are identified: processing delay, queuing delay, transmission delay and propagation delay. Kwon (2015) considers end-to-end latency to be the sum of these four delays. However, as discussed by Luckie et al. (2001), the Ping tool - the most widely used tool to investigate network delay - calculates round-trip-time (RTT), rather than the end-to-end latency. RTT or round-trip-delay (RTD) refers to the delay between the transmission of a packet and the reception of an acknowledgement or reply for that packet (adapted from Almes et al., 1999). In order to establish a common baseline with the Ping tool, therefore, latency is defined as the total time taken for a packet to be transmitted from one host to another, and for a response to be received from that host. This definition therefore includes the total processing delay, queuing delay, transmission delay and propagation delay for both directions of packet transfer.

2.3 Jitter

Demichelis et al. (2002) define jitter (in the context of IP packet delay variation) in the IETF's RFC 3393 as the difference between the one-way-delay of a selected packet pair in a stream of packets going from measurement point 1 to measurement point 2. Zhang et al. (2002) further clarify the term by providing the following diagram which illustrates the delay variation in consecutive packets.

However, this definition of jitter by Demichelis et al. (2002) assumes that both measurement points are under the control of the tester, as it requires timestamps to be collected at both points for the measurement of one-way-delay. However, it was desired that the jitter test developed could be conducted when only a single measurement point was controllable. Therefore, the definition of jitter was modified slightly from the IETF's to utilise RTT instead of one-way-delay. Jitter is therefore defined in the context of this research as the difference between the RTT of a selected packet pair in a stream of packets going from measurement point 1 to measurement point 2, then returning to point 1.

Figure 2. Diagram illustrating jitter in transmitted packets
Source: Adapted from Zhang, et al. (2002)

2.4 Packet Loss

DVEO (2016) defines packet loss as the phenomenon where one or more packets, transmitted over an IP network, fail to arrive at a destination. The rate of occurrence of this phenomenon is the packet loss rate, typically measured as a percentage. Freire (2007) provides a similar definition, stating that the packet loss rate refers to the fraction of the total transmitted packets that did not arrive at the intended receiver.

Based on this description, packet loss will be defined simply as the phenomenon where a transmitted packet is never received by its intended recipient, with the packet loss rate defined mathematically as:

$$Packet\ loss\ (\%) = [(P_T - P_R) / P_T] * P_T$$

where: P_T = Total number of packets transmitted, and
P_R = Total number of packets received successfully.

3. Analysis of existing tools- and metric measurement techniques

This section examines some of the most popular tools associated with broadband connection quality measurement, with particular attention paid to the techniques utilised to gather the required data. The outcome of this analysis is the identification of the best approaches for performance measurement of the metrics under study, which will subsequently drive the development of TINQA.

3.1 speedtest.net

Ookla's speedtest.net testing platform is arguably one of the most popular broadband performance testing tools (Gavaletz et al., 2012). The platform is affiliated with over 80% of the world's ISPs and has facilitated over 9 billion tests (Ookla, n.d.). It is unsurprising, therefore, that many investigations have been performed on their testing methodologies. Beur et al. (2010) performed one such analysis, in which they presented detailed descriptions of the techniques employed by Ookla. The following steps summarise their description of the download speed test:

1. A series of small files are downloaded to roughly gauge the user's download speed, after which a suitable file size is chosen for the test.
2. The test downloads several copies of the file size chosen, through (up to) eight parallel hypertext transfer protocol (HTTP) connections.
3. Samples of the download speed are taken at a rate of (up to) 30 Hz.
4. The samples are sorted, and the highest 10% and lowest 30% are removed. The average of the remainder is then computed as the average measured download speed.

The upload speed test follows the same methodology, with the direction of data transfer reversed, up to step 4. At step 4, only the upper 50% of the samples are used to compute the average measured upload speed.

Beur et al. (2010) praised the use of multiple connections, stating that it effectively saturated connections and moved the bottleneck in the test to the access link rather than to the buffers on the end devices. Additionally, they investigated the use of sampling and the discarding of the highest 10% and lowest 30% of samples for the download test, stating that this method of skimming was found to improve the average speed computed by considering only the most representative samples of network capacity and also by compensating for TCP's deliberately conservative congestion control algorithms by discarding a larger fraction of the lowest sample values. However, they also noted that the skimming percentages chosen were not systematically determined, and as such they could possibly be further optimised. Additionally, the only explanation given for the discarding of the lowest 50% of samples in the upload test was that this eliminated anomalies in the test. It was therefore decided that for the improved tool to be developed, skimming percentages of 10% and 30% would be used for both the upload and download tests initially, and the percentages would then be further optimised through systematic testing.

Beur et al. (2010) highlighted two further observations: that the test examines a number of servers to determine which would allow for the lowest latency during the test and that the default test length is 10 seconds. The use of a low-latency test server is crucial as high latencies severely degrade throughput on TCP-based connections, such as the HTTP connections by Ookla (Rogier, 2016). The default test length of 10 seconds is also used by other tools and was confirmed by Beur et al. (2010) to be an acceptable testing time for many cases.

Moreover, Abolfazli et al. (2015) conducted tests involving speedtest.net and concluded that the methodology produced accurate results with a standard deviation of ≤0.44 Mbps. Therefore, the testing process described above was taken as an important baseline for the development of the improved tool in this research.

3.2 Network Diagnostic Tool (NDT)

M-Lab's Network Diagnostic Tool (NDT) was also analysed by Beur et al. (2010). They discussed how the tool differed from many others by transferring as much data as possible in a specified window of time (10 seconds) for their speed test, rather than utilising a fixed file size for download while testing the network connection.

This unique approach was found to be advantageous for several reasons. Firstly, it eliminated the setup time incurred by other tests due to the need to estimate an appropriate file size for the connection. Secondly, this technique ensured that the connection under test remained saturated throughout the predefined test period. With smaller file sizes, tests may have completed prematurely due to completed file transfers.

Feamster (2016), however, who investigated the link between connection count and achievable throughput over TCP, stated that NDT's speeds results were inaccurate due to its use of only one connection for testing. It was therefore concluded that if our improved tool was to utilise the novel approach presented by M-Labs, this methodology had to be strengthened using other techniques such as the utilisation of multiple connections for the test.

3.3 testmy.net

Although no third-party analyses of the testmy.net tool were evident in the literature, the unique flexibility and functionality offered by this tool provided several ideas for the development of the improved tool. The site offers users the choice of using single-threaded or multi-threaded tests to measure speed, rather than letting the number of connections be determined algorithmically during the testing process. This added flexibility would be extremely useful for users such as application designers, who wish to investigate the performance that their application would achieve with particular numbers of parallel connections. Additionally, the site offers graphing capabilities that allow users to observe how their speeds varied for the duration of the test. Figure 3 illustrates a sample of this graphing functionality.

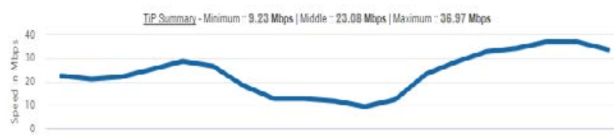

Figure 3. Example of a speed test graph
Source: Obtained from http://testmy.net

Additionally, for registered users, the tool logs measurements to a database. This allows deeper analyses, such as being able to compare results across several tests or across several users of the same ISP. These features of testmy.net provided guidance for additional functionality in the improved tool, in all of the metric modules.

3.4 Ping

Ping is a widely used latency measurement utility (Shamsi and Brocmeyer, 2009). It utilises ICMP to transmit a series of echo request packets and waits for the echo reply packets usually sent in response. The sequence numbers of the echo requests and replies are compared to attempt to tag each request-reply pair with a timestamp or RTT value. This process is illustrated in Figure 3.

The RTT values are then processed and minimum, average and maximum values are computed. Huston (2003) mentioned further measurement techniques using the Ping utility, stating that the data returned by the tool could be used to infer variance (jitter) and dropped packets (packet loss) as well.

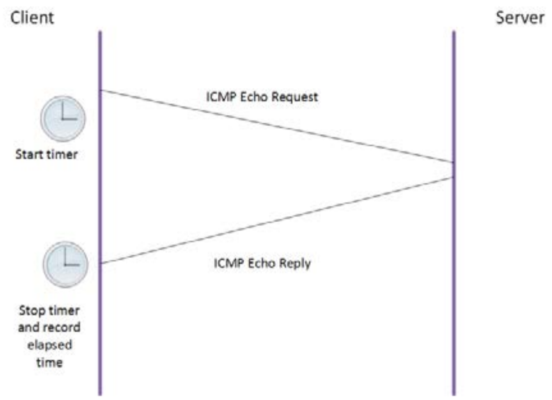

Figure 3. The process of RTT measurement using Ping

Anuskiewicz (2008) investigated jitter measurement in detail and describes the following methodology for determining jitter:

1. The RTT for the first packet successfully received by a host is calculated and stored as a reference RTT.
2. When the second packet is received, the RTT is also calculated and stored as the current RTT, and the difference between this current RTT and the reference RTT is calculated and stored as one value of jitter. The reference RTT is now updated to the last RTT measured.
3. Step 2 is repeated until the desired numbers of jitter measurements are obtained.
4. Minimum, maximum and average jitter values are then computed.

Based on this methodology, it was concluded that the Ping tool could be adapted for the determination of jitter. Moreover, Shamsi and Brocmeyer (2009) describe packet loss measurement strategies and, based on their descriptions, Figure 5 illustrates how the Ping tool could be used to infer packet loss that was created for this research.

Based on these discussions, the techniques employed by the Ping tool were deemed valuable for use in the latency, jitter and packet loss test modules of the enhanced tool.

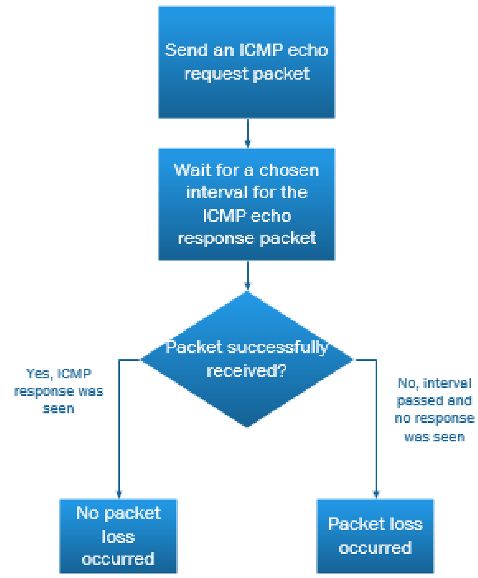

Figure 5. The technique to measure two-way packet loss
Source: Adopted from Shamsi and Brocmeyer (2009)

4. Summary of considerations for design of enhanced tool

This section presents the development of TINQA. It focuses on the algorithms used in the development of the four performance tests – speed, latency, jitter and packet loss tests – which ultimately make up the four main modules of the application. The approaches will be utilised in the development of TINQA, based on the analyses of existing tools. For each of the developed modules, the incorporation of the techniques as summarised in Table 1 is referenced.

Table 1. Summary of the techniques adopted for TINQA Development

Tool	Metric	Techniques adopted for TINQA	Justification of adopted techniques
speedtest.net	Speed	Multiple TCP connections	Creates saturated connections; bottleneck in access link rather than end device buffers
		Discarding of highest 10% and lowest 30% of samples	Accounts for throughput losses due to TCPs congestion management mechanisms
		Low latency test servers	Reduction of throughput degradation on TCP-based connections
		Default test length of 10 sec	Widely used and proven adequate
NDT	Speed	Transfer of as much data as possible in a fixed time period	Elimination of setup time; test connection remains saturated throughout test period
testmy.net	Speed	Choice of single-threaded or multi-threaded tests	Can examine performance using parallel connections
		Graphing capabilities	Visual analyses can be performed
		Logging of metric measurements	Greater flexibility in performance analysis/reporting
Ping	Jitter, Packet loss, Latency	Use of ICMP echo request and echo response packets	Allows for accurate measurement using a popular technique, which allows for easy comparison of results.

4.1 Speed test module

The speed test module was designed to determine both download and upload speeds of a user's connection. The following steps describe the algorithm utilised for the test:

1) The user is prompted for several configuration parameters including their preferred test server, number of download and upload connections to be run in parallel, and time over which the tests should be run.

   The servers available for the tests consist of several FTP and HTTP servers around the world, specifically configured for running speed tests. All the chosen servers have large files (>512MB) available for downloading, which ensured that the tests would be suitable for high speed connections. The number of parallel connections was set to 6 by default, as this was shown by Altman et al. (2006) to achieve almost 95% link utilisation (independent of the link capacity). The time for which the tests were run was set to 10s by default, based on the findings of Beur et al. (2010).
2) Once the user initiates the test, TINQA begins the download test and immediately opens a number of parallel TCP connections (depending on user preference). A copy of the largest speed test file available on the selected server is then downloaded through each of the open connections, and the cumulative download speed over all the connections is sampled at a rate of 5Hz. The results are graphed to the user.

   This step combines the advantages of M-Lab's NDT tool and Ookla's speedtest.net tool, allowing TINQA to produce speed test results almost instantaneously, while utilising multiple parallel TCP connections to ensure that the network link is fully saturated during the test.
3) The download speed is continuously sampled for the time configured by the user, after which the downloads are cancelled and the partially downloaded files are deleted.
4) TINQA then processes all the download speed samples, removing the top 10% and bottom 25%. These numbers were adjusted from the findings of Beur et al. (2010), as they were found, through testing, to produce more accurate results. The remaining 65% of the samples are used to determine the average download speed.
5) TINQA then prepares several large (250 MB) files consisting solely of 0s, and opens a number of parallel TCP connections (determined by the user) to the chosen FTP upload server.
6) The files are uploaded to the server, the upload speed is sampled at a rate of 5Hz and graphed for the user.
7) Once the chosen test time has elapsed, the uploads are cancelled. The files created for uploading are deleted from the user's machine, and the server similarly deletes the received files.
8) TINQA repeats step 4 for the upload speed samples to determine the average upload speed of the link. Although Beur et al. (2010) showed that Ookla skimmed the entire lower 50% of values, it was found that using the same percentages of the download speed test produced accurate results for the upload test as well.
9) The user is given the option of storing the recorded speed results in a database, which allows them to fetch the results at a later time and observe how their connection's download and upload speeds vary over time.

4.2 Latency test module

This module was designed to allow users to perform easily configurable tests to measure the latency of their connection. The following steps describe the algorithm used by this module:

1) The user is prompted for several parameters governing the test. These parameters include the remote server to test to, the packet data size, the time-to-live (TTL) value, the timeout value, the interval between sending ICMP echo requests, and the option to set the don't-fragment flag. The defaults for all these parameters are set to the defaults of the Windows Ping tool, to leverage users' expected familiarity with that tool. The user is also given the choice of running the test for a specific time period or until a specific number of ICMP echo requests are transmitted.
2) The test is initiated, and ICMP echo request packets are formed according to RFC 792 by Postel (1981). The packets are transmitted out of a socket and a timer is started. When a corresponding ICMP echo reply message is received, the timer is stopped and the RTT for the packet pair is recorded as one sample of latency.
3) Step 2 is repeated for either the time or the number of packets set by the user. Each sample of latency is added to a graph as it is obtained.
4) Once the test has completed, the samples are processed and the minimum, maximum and average latency are computed.
5) The user is given the option of storing the results in a database, which allows them to later view how their connection's latency varies over time.

4.3 Jitter test module

This module was designed to allow users to perform easily configurable tests to measure the jitter of their connection. The following steps describe the algorithm used by this module:

1) The user is prompted for several parameters governing the test. These parameters include all parameters mentioned for the latency test module.

2) The test is then initiated, and ICMP echo request packets are again created. However, for this test, the first RTT value recorded is not sampled, but is simply stored as the reference RTT.
3) Step 2 is repeated and another value of RTT is obtained. The modulus of the difference between this new RTT value and the reference RTT value is then calculated, and the result of this calculation is stored as the first sample of jitter.
4) Steps 2 and 3 are repeated for either the time or the number of packets set by the user. Each sample of jitter is added to a graph as it is obtained.
5) Once the test has completed, the samples are processed and the minimum, maximum and average jitter are computed.
6) The user is given the option of storing the results in a database, which allows them to later view how their connection's jitter varies over time.

4.4 Packet loss test module

This module was designed to allow users to perform easily configurable tests to measure the packet loss of their connection. The following steps describe the algorithm used by the module:

1) The user is prompted for several parameters governing the test. These parameters include all parameters mentioned for the latency test module.
2) The test is initiated, and ICMP echo request packets are again created. However, for this test, TINQA does not measure RTT values, but rather simply listens for ICMP echo replies. Once the echo request is transmitted, a timer is started. If the corresponding ICMP echo reply is received within the timeout period specified by the user, then no packet loss has occurred. However, if the timeout period passes and no reply is received, packet loss is said to have occurred. In both cases, one sample of packet loss has been obtained.
3) Step 2 is repeated for either the time or the number of packets set by the user. Each sample of packet loss is added to a graph as it is obtained.
4) Once the test has completed, the samples are processed and the packet loss percentage is computed.
5) The user is given the option of storing the results in a database, which allows them to later view how their connection's packet loss varies over time as a simple average. This test sends packets at a fixed rate (with respect to time), and as such no weightings are considered for the results to be presented as a moving average.

4.5 Integration of Modules

The four modules were then coded using C#, using the .NET framework to combine the functional code with the GUI. A MySQL database was used to store the results of the tests and subsequently fetch them for viewing, and a fifth module was added to control the flow of the gathered results to and from the database, ensuring that users were able to easily view their results on demand.

5. Comparative analysis of TINQA and other broadband performance measurement tools

In this section, the performance of TINQA is compared to those of existing tools in order to verify its functionality, accuracy and robustness. Tests of each of the modules are presented. All tests were conducted on an ADSL2+ Internet connection with advertised speeds of 2Mbps/512Kbps at approximately the same time of day, with no other hosts or applications utilising the network and with the same host machine used for all tests.

5.1 Comparison of speed test tools

Since Beur et al. (2010) showed that tools which utilise multiple parallel connections produce the most accurate speed test results, two popular multithreaded tools - speedtest.net and testmy.net - were chosen for comparison.

All tools were configured to utilise servers in northern Europe in order to reduce the influence of latency on the tests. testmy.net was additionally configured to use 6 servers for the test, to match the default number of connections of TINQA. speedtest.net did not allow for such configuration. Additionally, speedtest.net was found to be the only other popular speed test tool which offered multithreaded upload tests. Therefore, the upload speed test results from testmy.net were measured using a single connection. Tables 2 and 3 show the results of the download and upload testing, respectively.

Table 2. Results of comparison of download tests

Test number	TINQA result	Tool used	Tool result	Percentage difference
1	1.88 Mbps	speedtest.net	1.95 Mbps	-3.72%
2	1.88 Mbps	speedtest.net	1.99 Mbps	-5.85%
3	1.89 Mbps	speedtest.net	1.98 Mbps	-4.76%
4	1.88 Mbps	testmy.net	1.60 Mbps	14.89%
5	1.88 Mbps	testmy.net	1.30 Mbps	30.85%
6	1.88 Mbps	testmy.net	1.50 Mbps	20.21%

Table 3. Results of comparison of upload tests

Test number	TINQA result	Tool used	Tool result	Percentage difference
1	0.66 Mbps	speedtest.net	0.56 Mbps	15.15%
2	0.66 Mbps	speedtest.net	0.57 Mbps	13.64%
3	0.66 Mbps	speedtest.net	0.59 Mbps	10.61%
4	0.62 Mbps	testmy.net	0.54 Mbps	12.90%
5	0.69 Mbps	testmy.net	0.62 Mbps	10.14%
6	0.70 Mbps	testmy.net	0.61 Mbps	12.86%

The results obtained in Table 2 illustrate that the download speeds recorded by both TINQA and speedtest.net were very consistent, with all speeds within a ±0.1 Mbps range. Conversely, the results obtained by testmy.net varied significantly, with a maximum deviation of 0.3 Mbps. This variance was attributed to testmy.net's method of selecting an appropriate file size for the download test. The tool measured the minimum file size, which took longer than 7 seconds to download to the host machine, and used this file size for the test. However, it was found that this tool utilised file sizes smaller than 3 MB for all tests in this comparison, which resulted in tests completing very quickly. Larger file sizes may have produced more stable results.

The results in Table 3 illustrate that the upload speeds recorded by TINQA were consistently higher than those recorded by both speedtest.net and testmy.net, with a maximum deviation of +0.1 Mbps. This was attributed to the difference in processing of the samples for the upload speed and it was decided that further testing would be required to tune the skimming percentages for the upload test, to ensure that the results were more in line with other tools.

5.2 Comparison of latency test tools

For the latency test comparison, TINQA's results were compared to the results produced by the Windows® Ping tool, since this allowed for comparison using the same protocol (ICMP) and for similar configurations for both tools. Ookla's pingtest.net tool was also chosen to compare the latency measured using ICMP in TINQA to the latency measured using TCP in pingtest.net. For all tests, the Bright House Networks server located in Orlando, Florida, was used, since this server was available for testing in all the tools.

Table 4 shows the result of the testing. It was observed that there were no significant differences in latency results between the three tools. Although a high difference of 23.2% was recorded in one test, this result was likely an outlier caused by varying traffic demands in the link between the tool and the test server, since this result was much lower than the others and since all other tests produced much lower differences. Increasing the number of packets used for the test may reduce such fluctuations and therefore decrease the occurrences of these significant outliers.

Table 4. Results of comparison of average latency using selected latency test tools

Test number	TINQA result	Tool used	Tool result	Percentage difference
1	200 ms	pingtest.net	217 ms	-8.5 %
2	271 ms	pingtest.net	236 ms	12.9 %
3	228 ms	pingtest.net	203 ms	11.0 %
4	206 ms	Windows Ping	199 ms	3.4 %
5	206 ms	Windows Ping	232 ms	-12.6 %
6	155 ms	Windows Ping	119 ms	23.2 %

It was also observed that the latency measured by TINQA using ICMP and the latency measured by Ookla's pingtest.net using TCP were similar, indicating that neither protocol was being prioritised over the other along this route. However, details about Ookla's configuration for this test, such as default packet size and TTL, could not be located, so deeper analyses into these tests could not be performed.

5.3 Comparison of Jitter Test Tools

For the jitter test comparison, the line quality test by freeola (n.d.) and Ookla's pingtest.net were chosen since they allowed for specification of the server to be used for testing, allowing for servers in close proximity to be used for all of the tests. The servers chosen were all located in the southern part of the UK.

The results of the comparison are presented in Table 5. It was observed that the differences in jitter values varied more significantly than the other tests, but were still relatively close. The variation in the results were attributed to variations in traffic patterns between the client and server in the tests, since these tests were all performed from a client machine located in Trinidad and Tobago to a server located in the UK, which meant that there were a large number of nodes and links in the path being tested, and each of these nodes and links would exhibit their own traffic and device load fluctuation. However, similarly to the latency test, increasing the number of packets used for the test may decrease these outliers.

Table 5. Results of comparison of average jitter using selected jitter test tools

Test number	TINQA result	Tool used	Tool result	Percentage difference
1	171.8 ms	pingtest.net	145 ms	26.8 %
2	109.9 ms	pingtest.net	129 ms	-19.1 %
3	234 ms	pingtest.net	277 ms	-18.4 %
4	121 ms	freeola	149 ms	-23.1 %
5	252 ms	freeola	265 ms	-5.2 %
6	254 ms	freeola	225 ms	11.4 %

5.4 Comparison of packet loss test tools

The packet loss test comparison was performed using the same tools as the jitter test comparison, since these tools also provided results for packet loss. The results of the comparison are presented in Table 6. All results were observed to be similar, with only one outlier of 2% observed. This outlier was likely due to a transient traffic spike along the path being tested, which resulted in a node along the path discarding some of the test packets.

Table 6. Results of comparison of packet loss percentage using selected packet loss test tools

Test number	TINQA result	Tool used	Tool result	Percentage difference
1	0 %	pingtest.net	2 %	2 %
2	0 %	pingtest.net	0 %	0 %
3	0 %	pingtest.net	0 %	0 %
4	0 %	freeola	0 %	0 %
5	0 %	freeola	0 %	0 %
6	0 %	freeola	0 %	0 %

6. Conclusion

In this paper, some of the most popular tools and techniques used for the measurement of broadband Internet connection quality were examined and the findings used to develop a new tool - TINQA. TINQA provides a unified testing platform, allowing for measurement of four of the most important key performance indicators (speed, latency, jitter and packet loss) from a single application. Additionally, TINQA offers greater flexibility than existing tools in how tests are configured, allowing additional user configurations for parameters such as test time and number of parallel connections. Moreover, TINQA offers better processing and presentation of results, allowing for storage of results in a database and graphing capabilities. Based on the initial testing of TINQA, it is evident that the tool provides results similar to those provided by the most popular tools on the market. This aids to validate its performance and demonstrates that the increased flexibility of the tool does not compromise its functionality and reliability.

Despite its enhanced capabilities over existing performance measurement tools, TINQA can be further improved. More extensive testing can be undertaken in order to perform additional tuning of TINQA's tests, by utilising tools such as NetEM (described by The Linux Foundation (2016)) to manipulate parameters such as RTT and packet loss. Another possible avenue for optimisation is in the percentage of samples skimmed from the speed tests, with the actual percentage being determined by conducting a large number of tests over connections of different speeds and architectures. Additionally, support can be provided for a greater number of testing servers and for tests, using protocols such as User Datagram Protocol (UDP) and HTTP-based speed tests, and TCP-based latency, jitter and packet loss tests.

In addition, TINQA can be ported to other platforms so that it can be used on Linux and Macintosh-based systems. This final improvement would pave the way for the tool to be used in specialised hardware-based testing platforms, such as the platform developed by SamKnows for the Federal Communications Commission (FCC, 2016).

References:

Abolfazli, S., Tabassi, A., Sanaei, Z. and Rosen, S., (2015), "Throughput measurement in 4G wireless data networks: Performance evaluation and validation", *Proceedings of the IEEE Symposium on Computer Applications and Industrial Electronics (ISCAIE2015)*, Langkawi, Malaysia, April, DOI: 10.1109/ISCAIE.2015.7298322

Almes, G., Kalidindi, S. and Zekauskas, M., (1999), *A Round-trip Delay Metric for IPPM*, Available at: https://tools.ietf.org/html/rfc2681 (Accessed 27 June 2017).

Altman, E., Barman, D., Tuffin, B. and Vojnovic, M. (2006). "Parallel TCP sockets: Simple model, throughput and validation", *Proceedings of the 25th IEEE International Conference on Computer Communications (INFOCOM 2006)*, Barcelona, Spain, April, Available at: http://ieeexplore.ieee.org/document/4146757/ (Accessed 13 Dec. 2017).

Anuskiewicz, J. (2008), *Measuring jitter accurately*, Available at: http://www.lightwaveonline.com/articles/2008/04/measuring-jitter-accurately-54886317.html (Accessed 20 June 2017).

AppNeta, (2012), *Show Me the Bandwidth*, Available at: https://www.appneta.com/blog/show-me-the-bandwidth/

Apteligent, (2015), *Lag Kills! How App Latency Wrecks*. Available at: https://www.apteligent.com/wp-content/uploads/2016/02/STL_Europe-AppLAGReport-Oct2015-p2-1.pdf (Accessed December 2017).

Beuran, R., Ivanovici, M., Dobinson, B., Davies, N. and Thompson, P. (2003), "Network quality of service measurement system for application requirements evaluation", *Proceedings of the International Symposium on Performance Evaluation of Computer and Telecommunication Systems*, Montreal, Canada, 3 pp.380-387. Available at: https://www.jaist.ac.jp/~razvan/publications/qos_measurement.pdf (Accessed 13 Dec. 2017).

Beur, S., Clark, D. D. and Lehr, W. (2010), *Understanding broadband speed measurements*, s.l.: M-Lab. Available at: https://www.measurementlab.net/publications/understanding-broadband-speed-measurements.pdf

Chimento, P. and Ishac, J., (2008), *Defining Network Capacity*, Available at: https://tools.ietf.org/html/rfc5136#section-2.3.6

Claypool, M. and Claypool, K. (2010), "Latency can kill: precision and deadline in online games", Proceedings of the first annual ACM SIGMM conference on Multimedia systems (MMSys 2010), ACM, Phoenix, Arizona, February, pp.215-222. Available at: https://dl.acm.org/citation.cfm?id=1730863 (Accessed 13 Dec. 2017).

Demichelis, C., and Chimento, P., (2002), *IP Packet Delay Variation Metric for IP Performance Metrics (IPPM)*, Available at: https://tools.ietf.org/html/rfc3393#ref-8

DVEO (2016), *What Causes Packet Loss IP Networks*, Available at: http://www.dveo.com/pdf/What-Causes-Packet-Loss-in-IP-Networks-Article.pdf (Accessed 16 June 2017).

Ericsson, (2013), *Socioeconomic Effects of Broadband Speed*, s.l.: s.n. Available at: http://nova.ilsole24ore.com/wordpress/wp-content/uploads/2014/02/Ericsson.pdf

Feamster, N. (2016), *Revealing Utilisation at Internet Interconnection Points*. Available at: https://arxiv.org/pdf/1603.03656.pdf (Accessed 20 June 2017).

FCC (2016), *Measuring Fixed Broadband Report*, Federal Communications Commission s.l.: s.n. Available at: http://data.fcc.gov/download/measuring-broadband-america/2016/2016-Fixed-Measuring-Broadband-America-Report.pdf

freeola (n.d.), *Broadband Line Quality Test*, Available at: http://freeola.com/line-test/ (Accessed 22 June 2017).

Freire, M., (2007), *Encyclopedia of Internet Technologies and Applications*. s.l.: IGI Global. Available at: http://ru6.cti.gr/ru6-old/publications/2641165.pdf

Gavaletz, E., Hamon, D. and Kaur, J. (2012), *In-Browser Network Performance Measurement*, Available at: https://www.w3.org/2012/10/unc.edu.perf.paper.pdf (Accessed 20 June 2017).

Huston, G., (2003), "Measuring IP Network Performance", *The Internet Protocol Journal*, Vol.6, No.1. Available at: https://www.cisco.com/c/en/us/about/press/internet-protocol-journal/back-issues/table-contents-23/measuring-ip.html

Kwon, M., (2015), "A Tutorial on Network Latency and Its Measurements", In: Soyota, T. (ed.), *Enabling Real-Time Mobile Cloud Computing through Emerging Technologies*, s.l.:IGI Global. Available at: https://www.researchgate.net/publication/281455260_Enabling_

Real-Time_Mobile_Cloud_Computing_through_Emerging_Technologies

Luckie, M. J., McGregor, A. J. and Braun, H.-W., (2001), *Towards Improving Packet Probing Techniques*, ACM, California

Mathis, M. and Allman, M., (2001), *RFC 3148 - A Framework for Defining Empirical Bulk Transfer Capacity Metrics*, Available at: https://tools.ietf.org/html/rfc3148 (Accessed 16 June 2017).

Ookla, (2010), *Understanding Broadband Measurement*, Available at: http://www.ookla.com/docs/UnderstandingBroadbandMeasurement.pdf (Accessed 14 June 2017).

Ookla (n.d.) *The Definitive Source for Global Internet Metrics*, Available at: http://www.ookla.com (Accessed 20 June 2017).

Postel, J. (1981), *Internet Control Message Protocol*. Available at: https://tools.ietf.org/html/rfc792 (Accessed 21 June 2017).

Rogier, B. (2016), *Network Performance: Links between Latency, Throughput and Packet Loss*. Available at: http://blog.performancevision.com/eng/earl/links-between-latency-throughput-and-packet-loss (Accessed 20 June 2017).

Shamsi, J. and Brocmeyer, M. (2009), *Principles of Network Monitoring*, Available at: https://www.researchgate.net/publication/228831271_Principles_of_Network_Monitoring (Accessed 20 June 2017).

Strauss, J. and Kaashoek, M. F. (n.d.), *Estimating Bulk Transfer Capacity*. Available at: http://web.mit.edu/jastr/Public/paper.pdf (Accessed 16 June 2017).

Sugeng, W., Istiyanto, J. E., Mustofa, K. and Ashari, A. (2015), "The impact of QoS changes towards network performance", *International Journal of Computer Networks and Communications Security*, Vol.3, No.2, pp. 48-53.

testmy.net (n.d.), *Automatic Speed Test*, Available at: http://testmy.net/ (Accessed 20 June 2017).

The Linux Foundation (2016), *NetEM*, Available at: https://wiki.linuxfoundation.org/networking/netem

Zhang, L., Zheng, L. and Ngee, K.S. (2002), "Effect of delay and delay jitter on voice/video over IP", *Computer Communications*, Vol. 25, No.25, pp.863-873

Authors' Biographical Notes:

Rishi Latchmepersad *is an Associate Professional employed at The University of the West Indies (The UWI), St. Augustine Campus. He obtained a BSc degree in Electrical and Computer Engineering from The UWI in 2016. He is interested primarily in the field of IP networking.*

Tricia Ragoobar-Prescod *is currently a Lecturer in Communication Systems in the Department of Electrical and Computer Engineering, at The University of the West Indies (The UWI), St. Augustine Campus. She graduated with a BSc degree in Electrical and Computer Engineering at The UWI in 2002, and obtained an MSc and a PhD from the University of Strathclyde, Glasgow in 2005 and 2012 respectively. Dr. Ragoobar-Prescod's research interests are in areas of next-generation networks and telecommunications policy development.*

■