United States Patent	12388739
Kind Code	B2
Date of Patent	August 12, 2025
Inventor(s)	Robitaille; Claude et al.

System and method for out-of-line real-time in-service performance measurement

Abstract

A system for out-of-line testing of performance of a network, comprising a multiplexer at an input to the network; a demultiplexer at an output from the network; the multiplexer further comprising a traffic generator to insert synthetic traffic, and a first switch to accept an incoming customer traffic stream and join the incoming customer traffic stream with a synthetic traffic stream to form a total traffic stream, the total traffic stream fed to the input to the network; and the demultiplexer comprising a second switch to receive the total traffic stream from the output of the network, and separate the total traffic stream into the synthetic traffic stream and the customer traffic stream, and a traffic analyzer to analyze the separated synthetic traffic stream.

Inventors: **Robitaille; Claude (St-Placide, CA), Reamy; Chris (Colorado Springs, CO), Levesque; Germain (Saint-Laurent, CA)**

Applicant: **Accedian Networks Inc.** (Saint-Laurent, CA)

Family ID: **1000008749587**

Assignee: **ACCEDIAN NETWORKS INC. (Saint-Laurent, CA)**

Appl. No.: **18/461566**

Filed: **September 06, 2023**

Prior Publication Data

Document Identifier	Publication Date
US 20230421456 A1	Dec. 28, 2023

Related U.S. Application Data

continuation parent-doc US 16882818 20200526 US 11792083 child-doc US 18461566
continuation parent-doc US 16369098 20190329 US 10700941 20200630 child-doc US 16882818

continuation parent-doc US 15784339 20171016 US 10291484 20190514 child-doc US 16369098
continuation parent-doc US 15047031 20160218 US 9819553 20171114 child-doc US 15784339
continuation parent-doc US 14254932 20140417 US 9300565 20160329 child-doc US 15047031

Publication Classification

Int. Cl.: **H04L43/50** (20220101); **H04L12/46** (20060101); **H04L41/14** (20220101); **H04L43/026** (20220101); **H04L43/0823** (20220101); **H04L43/0852** (20220101); **H04L43/0888** (20220101); **H04L43/12** (20220101); **H04L43/18** (20220101)

U.S. Cl.:

CPC **H04L43/50** (20130101); **H04L12/4641** (20130101); **H04L41/14** (20130101); **H04L41/145** (20130101); **H04L43/026** (20130101); **H04L43/0823** (20130101); **H04L43/0852** (20130101); **H04L43/0888** (20130101); **H04L43/12** (20130101); **H04L43/18** (20130101);

Field of Classification Search

USPC: None

References Cited

U.S. PATENT DOCUMENTS

Patent No.	Issued Date	Patentee Name	U.S. Cl.	CPC
5745645	12/1997	Nakamura et al.	N/A	N/A
5793976	12/1997	Chen et al.	N/A	N/A
6901357	12/2004	Patiejunas	N/A	N/A
7539414	12/2008	Zami et al.	N/A	N/A
8289845	12/2011	Baldonado et al.	N/A	N/A
8705395	12/2013	Scott et al.	N/A	N/A
8917596	12/2013	Rochon	N/A	N/A
9300565	12/2015	Robitaille et al.	N/A	N/A
9819553	12/2016	Robitaille et al.	N/A	N/A
10003506	12/2017	Rochon	N/A	N/A
10291484	12/2018	Robitaille et al.	N/A	N/A
10700941	12/2019	Robitaille et al.	N/A	N/A
11792083	12/2022	Robitaille	370/253	H04L 43/50
2003/0108033	12/2002	Raisanen et al.	N/A	N/A
2003/0156548	12/2002	Sapp	N/A	N/A
2004/0236866	12/2003	Dugatkin et al.	N/A	N/A
2007/0266148	12/2006	Ruiz et al.	N/A	N/A
2008/0049623	12/2007	Qiu	370/241	H04M 7/0084
2009/0059804	12/2008	Fujikami et al.	N/A	N/A
2009/0219932	12/2008	Kobayashi	N/A	N/A
2010/0008248	12/2009	Constantine et al.	N/A	N/A
2011/0158112	12/2010	Finn et al.	N/A	N/A
2011/0261698	12/2010	Kamerkar et al.	N/A	N/A
2012/0051234	12/2011	Gintis et al.	N/A	N/A

2012/0131466	12/2011	Bugenhagen	N/A	N/A
2012/0311132	12/2011	Tychon et al.	N/A	N/A
2013/0077500	12/2012	Hutchison et al.	N/A	N/A
2013/0343207	12/2012	Cook	370/252	H04L 43/0852
2013/0343390	12/2012	Moriarty	370/392	H04L 43/0852
2014/0160945	12/2013	Robitaille	N/A	N/A
2014/0160961	12/2013	Dragulescu et al.	N/A	N/A
2014/0201364	12/2013	Yoshikawa et al.	N/A	N/A
2014/0254647	12/2013	Stott et al.	N/A	N/A
2014/0286174	12/2013	Iizuka et al.	N/A	N/A
2015/0199207	12/2014	Lin et al.	N/A	N/A
2015/0215184	12/2014	Sundararajan et al.	N/A	N/A
2016/0080242	12/2015	Grosso	N/A	N/A

FOREIGN PATENT DOCUMENTS

Patent No.	**Application Date**	**Country**	**CPC**
2014047359	12/2013	WO	N/A

Primary Examiner: Ly; Anh Vu H

Background/Summary

CROSS REFERENCE TO RELATED APPLICATIONS (1) This application is a continuation of U.S. patent application Ser. No. 16/882,818, filed May 26, 2020, now U.S. U.S. Pat. No. 11,792,083, which is a continuation of U.S. patent application Ser. No. 16/369,098, filed Mar. 29, 2019, now U.S. Pat. No. 10,700,941, which is a continuation of U.S. patent application Ser. No. 15/784,339, filed Oct. 16, 2017, now U.S. Pat. No. 10,291,484, which is a continuation of U.S. patent application Ser. No. 15/047,031, filed Feb. 18, 2016, now U.S. Pat. No. 9,819,553, which claims priority to U.S. patent application Ser. No. 14/254,932, filed Apr. 17, 2014, now U.S. Pat. No. 9,300,565, all of which are hereby incorporated by reference herein in its entirety.

FIELD OF THE INVENTION
(1) The present disclosure relates to Ethernet based mobile and business Ethernet networks.
SUMMARY
(2) A system for out-of-line testing of performance of a network, comprising a multiplexer at an input to the network; a demultiplexer at an output from the network; said multiplexer further comprising a traffic generator to insert synthetic traffic, and a first switch to accept an incoming customer traffic stream and join said incoming customer traffic stream with a synthetic traffic stream to form a total traffic stream, said total traffic stream fed to said input to said network; and said demultiplexer comprising a second switch to receive said total traffic stream from said output of said network, and separate said total traffic stream into the synthetic traffic stream and the customer traffic stream, and a traffic analyzer to analyze said separated synthetic traffic stream.
(3) A method for out-of-line testing of performance of a network, comprising multiplexing, using a multiplexer at an input to the network, an incoming customer traffic stream and a synthetic traffic stream to form a total traffic stream; said multiplexing comprising accepting the incoming customer traffic stream, inserting generated synthetic traffic stream, joining said inserted synthetic traffic stream with said incoming customer traffic stream to form said total traffic stream, and feeding said total traffic stream to said input to said network; demultiplexing, using a demultiplexer at an output from the network, said total traffic stream into said customer traffic stream and said

synthetic traffic stream, said demultiplexing comprising receiving said total traffic stream from an output from said network, separating said total traffic stream into the synthetic traffic stream and customer traffic stream, and analyzing said separated synthetic traffic stream.
(4) The foregoing and additional aspects and embodiments of the present disclosure will be apparent to those of ordinary skill in the art in view of the detailed description of various embodiments and/or aspects, which is made with reference to the drawings, a brief description of which is provided next.

Description

BRIEF DESCRIPTION OF THE DRAWINGS
(1) The foregoing and other advantages of the disclosure will become apparent upon reading the following detailed description and upon reference to the drawings.
(2) FIG. **1** shows an existing network with a source and destination.
(3) FIG. **2** shows the customer traffic rate over time.
(4) FIG. **3** shows an example of insertion of synthetic traffic at a rate to bring the total traffic rate up to an intended test rate.
(5) FIG. **4** shows one embodiment of a system to insert synthetic traffic into and remove synthetic traffic from the network.
(6) FIG. **5** shows insertion of synthetic traffic using a switch and a generator.
(7) FIG. **6** shows one embodiment of the separation of the total traffic stream into the customer traffic and synthetic traffic streams.
(8) While the present disclosure is susceptible to various modifications and alternative forms, specific embodiments or implementations have been shown by way of example in the drawings and will be described in detail herein. It should be understood, however, that the disclosure is not intended to be limited to the particular forms disclosed. Rather, the disclosure is to cover all modifications, equivalents, and alternatives falling within the spirit and scope of an invention as defined by the appended claims.
DETAILED DESCRIPTION
(9) Currently, throughput testing techniques, such as those described in Request For Comments (RFC) 2544 “Benchmarking Methodology for Network Interconnect Devices” or International Telecommunications Union Telecommunication Standardization Sector (ITU-T) standard Y.1564 “Ethernet service activation test methodology”, are considered “disruptive” tests and can only be executed by the operators when customer services are not running. This means when a customer calls in to say their service is not performing as expected, the network operator or service provider needs to take the customer out of service to identify the issue. Furthermore, network operators are often hesitant in deploying devices in-line with their customer traffic in fear of disrupting existing customer services.
(10) The system and method described in this specification allows operators to measure available capacity without disrupting existing customer services. The system to be described below enables the network operator to measure real-time traffic throughput under operational conditions, thereby providing operators with accurate metrics including capacity and optionally throughput.
(11) The system and method described in this specification uses customer traffic combined with synthetic traffic generated in real-time to “fill out” the gaps in the customer traffic to form the total traffic for testing. This allows testing of customer service performance during normal business hours, without taking the customer out-of-service. The system and method described in this specification achieves this without network operators having to insert devices in line to perform throughput testing.
(12) FIG. **1** shows an existing network **101**, on which throughput and capacity testing is to be

performed. The existing network **101** could be any type of network such as, for example, an Ethernet network, a 3G/4G network. It could also be a Local Area Network (LAN), a Wide Area Network (WAN), any type of wired network, an optical network and so on. A customer sends traffic **104** from source **102** to destination **103**. This traffic could be, for example, packets or frames or any type of protocol data unit, depending on what network **101** is.

(13) FIG. **2** shows the customer traffic rate **104-**A over time. As can be seen in FIG. **2**, customer traffic rate **104-**A is less than an intended test rate **105** to be used to test existing network **101**. In one embodiment, network **101** is an Ethernet network, and customer traffic rate **104-**A is the customer traffic rate associated with a virtual circuit in network **101**, and intended test rate **105** is the Committed Information Rate (CIR) associated with the virtual circuit in network **101**.

(14) Therefore, synthetic traffic can be inserted into the existing network **101** so as to increase the total traffic rate (customer traffic rate plus synthetic traffic rate) to the level of the intended test rate **105**, and test the response of network **101** to the total traffic at intended test rate **105**. An example is shown in FIG. **3**. Synthetic traffic with a rate **106-**A is inserted into the network, so as to increase the total traffic rate to intended test rate **105**. In one embodiment, network **101** is an Ethernet network. In a further embodiment, the synthetic traffic is used to test a virtual circuit of Ethernet network **101**.

(15) FIG. **4** shows one embodiment of a system to insert synthetic traffic into and remove synthetic traffic from the network. Multiplexer **401** comprises traffic generator **402**, and switch **404**. Demultiplexer **411** comprises traffic analyzer/monitor **403** and switch **412**.

(16) Traffic generator **402** monitors, in real-time, the customer traffic **104** from source **102** that is being passed on to the network in one direction and adds the necessary amount of synthetic traffic **106** to bring the total traffic rate to the intended test rate **105**. The synthetic traffic stream **106** and the customer traffic stream **104** are joined at switch **404**.

(17) At the other end, the two streams are separated at switch **412** of demultiplexer **411** into synthetic traffic **106** and customer traffic **104**. Traffic analyzer **403** analyzes the synthetic traffic **106** that is received at the other end, computes and displays statistics. In one embodiment, the monitor **403** collects and computes statistics on the number of frames or packets received, error rate and delay measurements. The customer traffic **104** is directed to destination **103**.

(18) The synthetic traffic **106** should be inserted out-of-line so as to not disrupt customer traffic. FIG. **5** shows out-of-line insertion of synthetic traffic using switch **404** and generator **402**. Incoming customer traffic **104** from source **102** is routed from port **404-1** of switch **404** to port **404-4**. A copy of the customer traffic is routed to port **404-2** of switch **404** which is connected to traffic generator **402**.

(19) In one embodiment, the generator **402** monitors the copy of the incoming customer traffic from port **404-2** in real-time, determines the customer traffic rate **104-**A and determines the synthetic traffic rate **106-**A needed to bring the total traffic rate to the intended test rate.

(20) In a further embodiment, the generator **402** monitors the incoming customer traffic **104** for gaps such as inter-packet or inter-frame gaps, and inserts synthetic traffic **106** into these gaps as shown in FIG. **5**. In a further embodiment, the generator **402** determines whether the inter-packet or inter-frame gaps are greater than a threshold duration before inserting the synthetic packets or frames.

(21) This multiplexed traffic must then rejoin the main stream of customer traffic from port **404-1** to port **404-4**. So as to not interfere with the customer traffic, the copy of customer traffic must be removed.

(22) In one embodiment, the copy of customer traffic is removed by traffic generator **402**.

(23) In another embodiment, the removal is performed in the following manner: Generator **402** gives the synthetic traffic a different marking from the copy of the incoming customer traffic. For example, generator **402** gives the synthetic traffic the same flow identifier as the incoming customer traffic **104**. An example of a flow identifier is a Virtual Local Area Network identification

(VLAN ID). Generator **402** then modifies the flow identifier of the copy of the incoming customer traffic so that it is different from the flow identifier of the synthetic traffic. Port **404-3** of switch **404** is set to only accept traffic with the same flow identifier as the incoming customer traffic **104**. The copy of the incoming customer traffic is then dropped at port **404-3**. The synthetic traffic then goes on to join the main stream of customer traffic from port **404-1** to port **404-4**.
(24) Using flow identifiers is one way to mark the synthetic traffic differently from the copy of the customer traffic so as to enable removal of the copy of the customer traffic. Other methods to mark the synthetic traffic as different from the copy of the customer traffic so that the copy of the customer traffic can be removed are also possible, as would be known by those of skill in the art.
(25) In a further embodiment, in order to facilitate easy removal of the synthetic traffic at the output from the network, the synthetic traffic is marked differently from the customer traffic. Methods to mark the synthetic traffic as different are known by those of skill in the art.
(26) In a further embodiment, either the stream of synthetic traffic from **404-3** to **404-4** or the main stream of customer traffic from **404-1** to **404-4** is delayed so that the two streams can be multiplexed in time at port **404-4**. In another embodiment, both streams are delayed to ensure correct multiplexing at port **404-4** as shown in FIG. **5**. Techniques to achieve this are well known to those of skill in the art and will not be discussed further. The total traffic stream is then routed into existing network **101** as shown in FIG. **5**.
(27) FIG. **6** shows one embodiment of the separation of the total traffic stream into the customer traffic and synthetic traffic streams. At demultiplexer **411**, the total traffic stream is received from the output of network **101** at port **412-1** of switch **412**. The synthetic traffic **106** is routed to traffic analyzer **403** via port **412-2** of switch **412**, and the customer traffic **104** is forwarded without disruption to port **412-3** of switch **412**. The separation of the streams is achieved based on the different markings applied to the streams. This way the two streams are demultiplexed.
(28) The synthetic traffic **106** is then analyzed by analyzer **403** to collect and compute statistics on the number of packets or frames received, error rate and delay measurements. It uses these statistics to determine metrics such as latency, throughput, jitter, bit error rate (BER), packet error rate (PER), throughput and so on.
(29) It would be known by one having skill in the art that the out-of-line method described above could be generalized to networks having more than one input and more than one output, or networks having more than one source and more than one destination. Furthermore it would be known by one of skill in the art that while the foregoing example refers to Ethernet networks, the techniques described above are not limited to Ethernet networks.
(30) Although the algorithms described above including those with reference to the foregoing flow charts have been described separately, it should be understood that any two or more of the algorithms disclosed herein can be combined in any combination. Any of the methods, algorithms, implementations, or procedures described herein can include machine-readable instructions for execution by: (a) a processor, (b) a controller, and/or (c) any other suitable processing device. Any algorithm, software, or method disclosed herein can be embodied in software stored on a non-transitory tangible medium such as, for example, a flash memory, a CD-ROM, a floppy disk, a hard drive, a digital versatile disk (DVD), or other memory devices, but persons of ordinary skill in the art will readily appreciate that the entire algorithm and/or parts thereof could alternatively be executed by a device other than a controller and/or embodied in firmware or dedicated hardware in a well known manner (e.g., it may be implemented by an application specific integrated circuit (ASIC), a programmable logic device (PLD), a field programmable logic device (FPLD), discrete logic, etc.). Also, some or all of the machine-readable instructions represented in any flowchart depicted herein can be implemented manually as opposed to automatically by a controller, processor, or similar computing device or machine. Further, although specific algorithms are described with reference to flowcharts depicted herein, persons of ordinary skill in the art will readily appreciate that many other methods of implementing the example machine readable

instructions may alternatively be used. For example, the order of execution of the blocks may be changed, and/or some of the blocks described may be changed, eliminated, or combined.
(31) It should be noted that the algorithms illustrated and discussed herein as having various modules which perform particular functions and interact with one another. It should be understood that these modules are merely segregated based on their function for the sake of description and represent computer hardware and/or executable software code which is stored on a computer-readable medium for execution on appropriate computing hardware. The various functions of the different modules and units can be combined or segregated as hardware and/or software stored on a non-transitory computer-readable medium as above as modules in any manner, and can be used separately or in combination.
(32) While particular implementations and applications of the present disclosure have been illustrated and described, it is to be understood that the present disclosure is not limited to the precise construction and compositions disclosed herein and that various modifications, changes, and variations can be apparent from the foregoing descriptions without departing from the spirit and scope of an invention as defined in the appended claims.

Claims

1. A system for testing performance of a network comprising: a traffic monitor to determine an original packet rate at which a plurality of original packets related to one or more customer services are sent from a traffic source; and a traffic generator coupled to said traffic monitor, to generate one or more synthetic packets at a synthetic packets rate based on a copy of the plurality of original packets to bring a total traffic rate equivalent to a testing rate, wherein said traffic generator merges said one or more synthetic packets with the plurality of original packets being transmitted to a destination and removes the copy of the plurality of original packets from being transmitted into the network.
2. The system of claim 1, wherein said total traffic rate is a sum of the original packet rate and the synthetic packets rate.
3. The system of claim 1, wherein said one or more synthetic packets are merged when there is a delay between two original packets.
4. The system of claim 3, wherein said delay is greater than a predetermined threshold.
5. A method to test a performance of a network, the method comprising: estimating an original packet rate of a plurality of original packets related to one or more customer services transmitted by a traffic source; generating one or more synthetic packets at a synthetic packet rate based on a copy of the plurality of original packets such that a total traffic rate reaches a predetermined test rate; merging said one or more synthetic packets with the plurality of original packets being transmitted without disrupting said one or more customer services; and removing the copy of the plurality of original packets from being transmitted into the network.
6. The method of claim 5, wherein said total traffic rate is a sum of the original packet rate and the synthetic packet rate.
7. The method of claim 5, wherein said merging of said one or more synthetic packets is performed when there is a delay between two original packets.
8. The method of claim 5, wherein said merging of said one or more synthetic packets is performed when there is a delay between two original packets that is greater than a predetermined threshold.
9. The method of claim 5, wherein removing comprises: assigning a flow identifier to the copy of the plurality of original packets; and removing, based on the flow identifier, the copy of the plurality of original packets from being transmitted into the network.
10. The method of claim 9, wherein the flow identifier is a Virtual Local Area Network identification (VLAN ID).
11. The method of claim 5, wherein merging comprises: identifying one or more gaps in the

plurality of original packets; and merging, based on the one or more gaps, said one or more synthetic packets with the plurality of original packets being transmitted without disrupting said one or more customer services.

12. The system of claim 1, wherein said traffic generator removes the copy of the plurality of original packets from being transmitted into the network based on a flow identifier of the copy of the plurality of original packets.

13. The system of claim 12, wherein the flow identifier is a Virtual Local Area Network identification (VLAN ID).

14. The system of claim 1, wherein said traffic generator merges said one or more synthetic packets with the plurality of original packets based on one or more gaps in the plurality of original packets.

15. A non-transitory computer-readable medium encoded with instructions that, when executed by a processor, causes the processor to: estimate an original packet rate of a plurality of original packets related to one or more customer services transmitted by a traffic source; generate one or more synthetic packets at a synthetic packet rate based on a copy of the plurality of original packets such that a total traffic rate reaches a predetermined test rate; merge said one or more synthetic packets with the plurality of original packets being transmitted without disrupting the one or more customer services; and remove the copy of the plurality of original packets from being transmitted into a network.

16. The non-transitory computer-readable medium of claim 15, wherein said total traffic rate is a sum of the original packet rate and the synthetic packet rate.

17. The non-transitory computer-readable medium of claim 15, wherein the instructions are operable to cause the processor to merge said one or more synthetic packets with the plurality of original packets when there is a delay between two original packets.

18. The non-transitory computer-readable medium of claim 15, wherein the instructions are operable to cause the processor to merge said one or more synthetic packets with the plurality of original packets when there is a delay between two original packets that is greater than a predetermined threshold.

19. The non-transitory computer-readable medium of claim 15, wherein the instructions are operable to cause the processor to remove the copy of the plurality of original packets from being transmitted into the network by: assigning a flow identifier to the copy of the plurality of original packets; and removing, based on the flow identifier, the copy of the plurality of original packets from being transmitted into the network.

20. The non-transitory computer-readable medium of claim 19, wherein the flow identifier is a Virtual Local Area Network identification (VLAN ID).
