UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

CISCO SYSTEMS INC.,

Petitioner

IPR2023-00402

U.S. Patent No. 8,830,821

DECLARATION OF HENRY H. HOUH,
UNDER 37 C.F.R. § 1.68 IN SUPPORT OF PETITION
FOR *INTER PARTES* REVIEW

TABLE OF CONTENTS

I. INTRODUCTION 4
II. QUALIFICATIONS AND PROFESSIONAL EXPERIENCE 7
III. LEVEL OF ORDINARY SKILL IN THE ART 16
IV. RELEVANT LEGAL STANDARDS 17
V. OVERVIEW OF THE '821 PATENT 19
VI. PROSECUTION HISTORY OF THE '821 PATENT 20
VII. CLAIM CONSTRUCTION 21
A. "*entity*" 22
VIII. IDENTIFICATION OF HOW THE CLAIMS ARE UNPATENTABLE 22
A. Ground 1: The Combination of Doshi and Guichard renders obvious claims 1-7, 9-11, 13, and 17-20 24
1. Summary of Doshi 24
2. Summary of Guichard 27
3. Reasons to Combine Doshi and Guichard 28
4. Claim 1 32
5. Claim 2 67
6. Claim 3 69
7. Claim 4 71
8. Claim 5 73
9. Claim 6 78
10. Claim 7 80
11. Claim 9 80
12. Claim 10 88
13. Claim 11 89
14. Claim 13 89
15. Claim 17 93

16. Claim 18 96
17. Claim 19 99
18. Claim 20 99
B. Ground 2: The Combination of Doshi, Guichard, and Huang renders obvious claims 8 and 12. 100
1. Summary of Huang 100
2. Reasons to Combine Doshi and Huang 101
3. Claim 8 105
4. Claim 12 108
C. Ground 3: The Combination of Doshi, Guichard, and Xu renders obvious claims 14-16. 108
1. Summary of Xu 108
2. Reasons to Combine Doshi and Xu 109
3. Claim 14 112
4. Claim 15 126
5. Claim 16 127
IX. CONCLUSION 127

I, Henry H. Houh, Ph.D., do hereby declare as follows:

I. INTRODUCTION

1. I am making this declaration at the request of Cisco Systems, Inc. in the matter of the *Inter Partes* Review of U.S. Patent No. 8,830,821 ("the '821 patent") to Cohn et al.

2. I am being compensated for my work in this matter at my standard hourly rate. I am also being reimbursed for reasonable and customary expenses associated with my work and testimony in this proceeding. My compensation is not contingent on the outcome of this matter or the specifics of my testimony.

3. I have been asked to provide my opinions regarding whether the subject matter of claims 1-20 ("the Challenged Claims") of the '821 patent would have been obvious to a person having ordinary skill in the art ("POSITA") at the time of the alleged invention, in light of the prior art. It is my opinion that the Challenged Claims would have been obvious to a POSITA.

4. In the preparation of this declaration, I have studied:

a. the '821 patent, Ex.1001;

b. the prosecution history of the '821 patent ("'821 File History"), Ex.1002;

c. U.S. Patent Pub. No. 2004/0205239 to Doshi et al. ("Doshi"), Ex.1005;

d. "Definitive MPLS Network Designs, Field-proven MPLS designs covering MPLS VPNs, pseudowire, QoS, traffic engineering, IPv6, network recovery, and multicast," by Guichard et al., Cisco Press, 2005 ("Guichard"), excerpt, Ex.1006;

e. U.S. Patent Pub. No. 2003/0117950 to Huang ("Huang"), Ex.1007; and

f. U.S. Patent Pub. No 2011/0141877 to Xu et al. ("Xu"), Ex.1025.

5. In forming the opinions expressed below, I have considered: the documents listed above; the relevant legal standards, including the standard for obviousness; and my own knowledge and experience based upon my work in the field of network communications and security as described below, as well as portions of the following additional materials:

a. "MATE: MPLS Adaptive Traffic Engineering" by Elwalid et al., IEEE, 2001, ("MPLS Traffic Engineering Article"), Ex.1008;

b. "Traffic Engineering with MPLS," by Osborne et al., Cisco Press, 2003 ("Osborne"), excerpt, Ex.1009;

c. "Fast Network Re-optimization Schemes for MPLS and Optical Networks" by Bhatia et al., 2006, ("Bhatia"), Ex.1010;

d. "RSVP-TE: Extensions to RSVP for LSP Tunnels," by Awduche et al., IETF Request for Comments 3209 (Dec. 2001) ("RFC 3209"), Ex.1011;

e. U.S. Patent Pub. No. 2008/0155257 to Werner et al. ("Werner"), Ex.1016;

f. U.S. Patent No. 7,624,142 to Jungck ("Jungck"), Ex.1017;

g. U.S. Patent Pub. No. 2006/0015781 to Rothman et al. ("Rothman"), Ex.1018;

h. "Overview and Principles of Internet Traffic Engineering," by Awduche et al., IETF Request for Comments 3272 (May 2002) ("RFC 3272"), Ex.1022;

i. "Multiprotocol Label Switching (MPLS) Traffic Engineering," Cisco IOS Release 12.0(5)S ("Release 12.0(5)S"), Ex.1023;

j. "Multiple Metrics for Traffic Engineering with IS-IS and OSPF," by Fedyk et al., IETF Internet Draft (May 2001) ("Fedyk"), Ex.1024;

k. U.S. Patent No. 5,699,403 to Ronnen ("Ronnen"), Ex.1026;

l. "The Internet Standards Process – Revision 3," by Bradner, IETF Request for Comments 2026 (Oct. 1996) ("RFC 2026"), Ex.1027;

m. U.S. Patent No. 8,102,774 to Gerö et al. ("Gerö"), Ex.1028;

n. U.S. Patent Pub. No. 2007/0201513 to Anderson et al. ("Anderson"), Ex.1029;

o. "Framework for Multi-Protocol Label Switching (MPLS)-based Recovery," by Sharma et al., IETF Request for Comments 3469 (Feb. 2003) ("RFC 3469"), Ex.1030;

p. U.S. Patent No. 6,980,906 to Kao et al. ("Kao"), Ex.1031;

q. U.S. Patent No. 7,502,884 to Shah et al. ("Shah"), Ex.1032;

r. U.S. Patent No. 7,197,418 to Fuller, III et al. ("Fuller"), Ex.1033;

s. U.S. Patent Pub. No. 2006/0250948 to Zamfir et al. ("Zamfir"), Ex.1034;

t. U.S. Patent Pub. No. 2007/0047469 to Vasseur et al. ("Vasseur '469"), Ex.1035; and

u. U.S. Patent Pub. No. 2006/0291391 to Vasseur et al. ("Vasseur '391"), Ex.1036.

6. Unless otherwise noted, all **emphasis** in any quoted material has been added. Claim terms are *italicized*.

II. QUALIFICATIONS AND PROFESSIONAL EXPERIENCE

7. The details of my background and education, and a listing of all publications that I have authored, are provided in my *Curriculum Vitae*, a copy of which is submitted as Ex.1004. The following is a brief summary of my relevant qualifications and professional experience.

8. I received a Ph.D. in Electrical Engineering and Computer Science from the Massachusetts Institute of Technology ("MIT") in 1998. I also received a Master of Science degree in Electrical Engineering and Computer Science in 1991, a Bachelor of Science Degree in Electrical Engineering and Computer Science in 1989, and a Bachelor of Science Degree in Physics in 1990, all from MIT.

9. During my college studies, I focused on communications and data networking. I took specialized courses including graduate courses in telecommunications networks, optical communications, and data networking. I, along with other graduate students in a networking research group, maintained both the computer workstations and the networking devices in the research group.

10. I have worked in data networking and distributed networking systems on several occasions. As part of my doctoral research at MIT from 1991-1998, I worked as a research assistant in the Telemedia Network Systems ("TNS") group at the Laboratory for Computer Science. The TNS group built a high-speed gigabit ATM network and applications which ran over the network, such as remote video capture (including audio), processing and display on computer terminals. In addition to helping design the core network components (such as the ATM switch), designing and building the high-speed ATM links, and designing and writing the device drivers for the interface cards, I also set up the group's web server, which at the time was

one of the first several hundred web servers in existence. Our high-speed data network carried multimedia data including video and audio data within ATM cells.

11. Like all ATM networks, interconnection of various endpoints in our ATM system required the setup of end-to-end virtual circuits which required, for each ATM switch in the connection path, the switch's header remapping tables to be configured for each virtual circuit required. I developed and implemented our system's protocol of controlling the content of these tables and the overall circuit setup.

12. The TNS group was the first group to initiate a remote video display over the World Wide Web. Vice President Al Gore visited our group in 1996 and received a demonstration of—and remotely drove—a radio-controlled toy car with a wireless video camera mounted on it; the video was encoded by TNS-designed hardware, streamed over the TNS-designed network and displayed using TNS-designed software.

13. I authored or co-authored twelve papers and conference presentations on our group's research. I also co-edited the final report of the gigabit networking research effort with Professor David Tennenhouse and Senior Research Scientist David Clark. David Clark is generally considered to be one of the fathers of the Internet Protocol and served as Chief Protocol Architect for the Internet. With its focus on networking, the group, including myself, set up and maintained the network

and computer systems. These systems included the networking on the workstations and desktops, the distributed file system, desktops and workstations, setting up and maintaining the distributed file system (Network File System) and the authentication system (Network Information Service, formerly known as Yellow Pages).

14. I defended and submitted my Ph.D. thesis, titled "Designing Networks for Tomorrow's Traffic," in January 1998. As part of my thesis research, I analyzed local-area and wide-area flows to show a more efficient method for routing packets in a network, based on traffic patterns at the time. My thesis involved analyzing flows of data in the network and the routing efficiencies gained by labeling the flows at the edge of the network, and routing the data based on the label instead of the IP destination address. My analysis included different methods of setting up the granularity of the flows for the most efficient use of routing resources. This type of label switching became popular with Multiprotocol Label Switching (MPLS) later implemented in various commercial routers. I gathered a large amount of network traffic, used publicly available network traces, and broke the traffic into different types of flows using various parameters for classifying the flows. My flow analysis is applicable to the flow classification at the ingress to an MPLS domain. My thesis also addressed real-time streamed audio and video. The network traffic that I analyzed was IP protocol traffic, including UDP and TCP.

15. I have been involved in Voice over Internet Protocol ("VoIP") data streaming technologies since 1997 and have specific experience in designing and testing VoIP networks.

16. From 1997 to 1999, I worked at NBX Corporation, which was acquired by 3Com Corporation in 1999. During this time, I was a Senior Scientist and Engineer working in IP Telephony. NBX delivered the world's first fully featured business telephone system to run over a data network, the NBX100. NBX was one of the first business phone systems to be configurable via a web interface. Users and administrators had access to varying levels of configuration for the phone system.

17. As part of my work at NBX, I designed the core audio reconstruction algorithms for the telephones which depacketized the voice data and reconstructed the audio. In addition, I designed the voice data packet transmission algorithms. I created a system to capture and analyze network packets sent by devices in the NBX system for aid in testing and debugging. I also designed and validated the core packet transport protocol used by the phone system. In addition, I designed and oversaw the development of the underlying transport protocol used by the NBX100 phone system for reliable packet transport. That transport protocol is still used by the NBX100 system and its successors and is estimated to be used hundreds of millions of times daily. I wrote NBX's first demonstration IP software stack, which added the capability for utilizing the NBX100 phone system on an IP network. I

also specified and prototyped the phone system's support for Differentiated Services. NBX first demonstrated a phone in the NBX100 system working over the Internet in 1998 at a trade show in Las Vegas, for which I acquired, set up, and configured Virtual Private Network tunnels to carry the traffic. I was later the lead architect in designing NBX's next-generation highly scalable system. After NBX was acquired by 3Com, I did some work with 3Com's cable equipment division, including demonstrating a working NBX IP phone system over 3Com's cable equipment infrastructure using an early version of DOCSIS at a trade show in 1999. The NBX100 was the market's leading business phone system to run on a data network for several years following its introduction. During that time, I became more familiar with the various standards relevant to Internet telephony as well as the problems which designers of commercial telephony operations were faced with in implementing VoIP.

18. I, along with two of NBX's founders, was awarded U.S. Patent No. 6,967,963 titled "Telecommunication Method for Ensuring On-Time Delivery of Packets Containing Time-Sensitive Data," for some of the work we did while at NBX.

19. After NBX, I worked at Teradyne, a test tool company primarily focused on semiconductors. Teradyne had recently acquired Hammer, a company that specialized in load and functional testing for telecommunications systems. The

Hammer product is well known as a telecom test tool. Teradyne spun out Hammer and several other internal divisions into an independent company called Empirix. I became Chief Technologist of the Hammer division of Empirix. Empirix was a leader in VoIP network testing and monitoring.

20. At Empirix, I laid out a new multi-year product vision for data network testing, secured internal funding for the effort, and led a team to deliver a new technology platform to the market in February 2001. This new product, PacketSphere, initially emulated network behavior so that wide-area VoIP connections could be tested in a lab. A later release allowed PacketSphere to generate high volumes of VoIP calls, including media streams, and to monitor the quality of VoIP voice streams. PacketSphere was also used in the Storage Area Network area. Later, the core technology was added to other Empirix products such as Empirix's Hammer XMS to monitor thousands of VoIP media streams in real time to determine their quality. PacketSphere was Empirix's most successful new platform introduction at the time. Companies purchased the PacketSphere product to emulate an Internet Protocol network to see the effects of deploying their product on the Internet prior to launch. PacketSphere received several industry awards.

21. During my time at Empirix, I presented lectures on VoIP and data network testing to companies including Lucent Labs (formerly AT&T Bell Labs). I was also invited to present several guest lectures in a software engineering course at

MIT. Since then, I have also participated twice as a unit lecturer in an experimental course that was taught by an Institute Professor (the highest award that a MIT Professor can achieve) and sponsored by the Chairman of the MIT Corporation (MIT's board of trustees).

22. From 2004 to 2008, I was employed by BBN Technologies Corp., a technology research and development company located in Cambridge, Massachusetts. BBN Technologies is a world-renowned company with expertise in acoustics, speech recognition, and communications technology. BBN Technologies staff have pioneered many internetworking technologies and Internet applications and have built some of the world's largest government and commercial data networks.

23. My duties and responsibilities at BBN Technologies generally included commercialization of the technologies developed by BBN Technologies, which included spinning off companies and growing commercial businesses in-house. More particularly, I was involved in utilizing the award-winning AVOKE STX speech recognition technology to create the public audio/video search engine EveryZing (formerly known as PodZinger) which was spun out into a stand-alone company now known as RAMP, Inc. PodZinger won the 2006 MITX Technology Award for best Web 2.0 Application and was also named the 2006 Forbes Favorite Video & Audio Search Engine, beating out Google, Yahoo, and other companies.

After managing the creation of the initial prototype system, PodZinger built out a full streaming audio and video search solution when I was the Vice President of Operations and Technology there. I was also involved in the Boomerang Mobile Shooter Detection project as the Vice President of Engineering for the program. The Boomerang system was deployed to Iraq and Afghanistan and was credited with saving many lives.

24. In 2012, I opened Einstein's Workshop, a 7,000 square foot facility for teaching science, technology, engineering, and math to children. I installed and configured the telephone system, designed and programmed the web site, and designed and configured the network, which has grown to roughly 100 computers, multiple WiFi access points, firewalls, multiple wireless networks, and multiple facilities. We also created an educational 3D Computer-Aided Design program, which we spun-out into a separate company, BlocksCAD. BlocksCAD has received grants from DARPA DSO and the USDA and has participated in the LearnLaunch, MIT Play Labs, and MassChallenge accelerator programs. BlocksCAD currently has over 150,000 registered users and is used in schools throughout the US.

25. From 1989 to 1990, I worked at AT&T Bell Laboratories on optical computers. This work generated six peer-reviewed papers, and multiple U.S. and European patent applications in which I was named as a co-author or inventor. I

also interned at AT&T Bell Laboratories in 1987 and 1988. Additional relevant experience in the field of computer networking is listed in my *Curriculum Vitae*.

26. I am a named inventor on several patents and published patent applications that are related to VoIP technology including: U.S. Patent No. 6,967,963, entitled "Telecommunication Method for Ensuring On-Time Delivery of Packets Containing Time-Sensitive Data"; U.S. Patent Application Publication No. 20020015387, entitled "Voice Traffic Packet Capture and Analysis Tool for a Data Network"; U.S. Patent Application Publication No. 20020016708, entitled "Method and Apparatus for Utilizing a Network Processor as Part of a Test System"; U.S. Patent Application Publication No. 20020016937, entitled "Method and Apparatus for Utilizing a Network Processor as Part of a Test System"; and U.S. Patent No. 7,590,542, entitled "Method of Generating Test Scripts Using a Voice-Capable Markup Language."

III. LEVEL OF ORDINARY SKILL IN THE ART

27. I understand there are multiple factors relevant to determining the level of ordinary skill in the pertinent art, including (1) the levels of education and experience of persons working in the field at the time of the invention; (2) the sophistication of the technology; (3) the types of problems encountered in the field; and (4) the prior art solutions to those problems.

28. A POSITA in the field of the ’821 patent, as of its earliest claimed priority date of June 22, 2011, would have been someone knowledgeable and familiar with network communications and multiprotocol label switching-transport (MPLS) techniques available at the time. Such a POSITA would have a bachelor’s degree in computer science, computer engineering, electrical engineering, or equivalent training, and approximately two years of experience working in the field of network communications and would be knowledgeable regarding MPLS techniques. Additional work experience can substitute for specific educational background, and vice versa.

29. For purposes of this Declaration, in general, and unless otherwise noted, my statements and opinions, such as those regarding my own experience and what a POSITA would have understood or known generally (and specifically related to the references I consulted herein), reflect the knowledge that existed in the relevant field as of the priority date of the ’821 patent.

IV. RELEVANT LEGAL STANDARDS

30. I am not an attorney. In preparing and expressing my opinions and considering the subject matter of the ’821 patent, I am relying on certain basic legal principles that Cisco’s counsel has explained to me.

31. I understand that prior art to the ’821 patent includes patents and printed publications in the relevant art that predate the earliest claimed priority date of the

'821 patent. For purposes of this Declaration, I am applying June 22, 2011 as the priority date of the '821 patent.

32. I have been informed by Cisco's counsel that a claimed invention is unpatentable under 35 U.S.C. § 103 if the differences between the claimed invention and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a POSITA. I have also been informed by Cisco's counsel that the obviousness analysis considers factual inquiries, including the level of ordinary skill in the art, the scope and content of the prior art, and the differences between the prior art and the claimed subject matter.

33. I have been further informed by Cisco's counsel that there are several recognized rationales for combining references or modifying a reference to show obviousness. These rationales include: (a) combining prior art elements according to known methods to yield predictable results; (b) simple substitution of one known element for another to obtain predictable results; (c) use of a known technique to improve a similar device (method, or product) in the same way; (d) applying a known technique to a known device (method, or product) ready for improvement to yield predictable results; (e) choosing from a finite number of identified, predictable solutions, with a reasonable expectation of success; and (f) some teaching, suggestion, or motivation in the prior art that would have led a POSITA to modify the prior art or to combine prior art teachings to arrive at the claimed invention.

V. OVERVIEW OF THE '821 PATENT

34. The '821 patent generally describes a "system and method for selecting MPLS network transport entities between a first endpoint and a second endpoint." '821 patent, Abstract. According to the '821 patent, the noted system and method provide "protection against multiple span or node failures" by utilizing transport entity pairs that are "fully or partially resource disjoint." '821 patent, 3:15-18. "[A]n overall cost for each pair of the plurality of transport entities is determined" and an entity pair, comprising a "working entity" and a "protection entity," is "selected to minimize an overall cost." '821 patent, 4:13-16, 29:-33. For communication, "[a]n active entity may be selected from the working entity and the protection entity." '821 patent, 4:57-58.

35. The '821 patent also discloses that an "event" triggers "[e]ntity pair reselection." '821 patent, 4:34-35. The reselection may result in selecting new entity pairs or selecting the same pairs. '821 patent, 4:63-5:8. The trigger may correspond to "when the operational status changes for an entity in the plurality of transport entities, when an entity is added to the plurality of transport entities, when an entity is removed from the plurality of transport entities, or due to combined cost changes in one or more entities." '821 patent, 4:35-41.

36. For reference, representative claim 1 is provided below:

> **1**. An entity selection method performed by a network device, comprising the steps of:
>
> providing a plurality of multi protocol label switching (MPLS) transport entities between a first endpoint and a second endpoint;
>
> determining an overall cost for each entity pair of said plurality of entities;
>
> selecting an entity pair from said plurality of transport entities based at least in part upon said overall cost; and
>
> if an entity pair reselection event occurs, reselecting said entity pair from the group consisting of said entity pair and a replacement entity pair comprising at least one entity distinct from the entities of said entity pair,
>
> wherein said entity pair reselection event is selected from a group consisting of adding an entity to said plurality of transport entities, removing an entity from said plurality of transport entities, an operational status change for one of said plurality of transport entities, and a change in overall cost for
>
> one of said plurality of transport entities.

37. As I will explain below, these concepts were well known as of the priority date of the '821 patent.

VI. PROSECUTION HISTORY OF THE '821 PATENT

38. The '821 patent claims priority to Provisional application No. 61/499,943 filed on June 22, 2011.

39. In response to an Office Action, the Applicants amended the independent claims "to clarify that 'reselecting said entity pair' means reselecting said entity pair from the group consisting of said entity pair and a replacement entity pair comprising at least one entity distinct from the entities of said entity pair." File History, 64. Applicants then argued that the prior art does not teach the amended

claim because it only addresses a failover scenario and not reselecting the path pair. File History, 63.

40. Notably, as I demonstrate below, there is nothing novel about the Challenged Claims of the '821 patent because all of the claimed features, including the reselection features forming the basis of Applicants' arguments and amendments, were known in the art and it would have been obvious to combine the relevant prior art teachings as claimed.

VII. CLAIM CONSTRUCTION

41. It is my understanding that in order to properly evaluate the '821 patent, the terms of the claims must first be interpreted. It is my understanding that for the purposes of this *inter partes* review, the claims are to be construed under the so-called *Phillips* standard, under which claim terms are given their ordinary and customary meaning as would have been understood by a POSITA in light of the specification and prosecution history, unless the inventor has set forth a special meaning for a term. I have also been informed that claim terms only need to be construed to the extent necessary to resolve the obviousness inquiry.

42. I have reviewed the entirety of the '821 patent, as well as its prosecution history. It is my opinion that, for purposes of applying the prior art presented herein to evaluate the patentability of the claims, a majority of the claim terms do not require express construction.

43. I have applied the so-called *Phillips* standard, which requires that claim terms are given their ordinary and customary meaning as would have been understood by a POSITA in light of the specification and prosecution history, unless the inventor has set forth a special meaning for a term. Below, for clarity, I discuss examples to provide context and understanding of the term "*entity*."

A. "*entity*"

44. The term "*entity*" is described in the specification as including a multiprotocol label switching ("MPLS") label switched path ("LSP") or a pseudo wire ("PW"):

> An exemplary embodiment may include a network with a set of **MPLS-TP transport <u>entities</u>, for example, <u>label switch paths (LSPs) or pseudo wires (PWs)</u> between a first endpoint and a second endpoint**. It is preferable that an operator will define the entity set such that, with high probability, at least one entity will stay up in the event of multiple span or node failures.

'821 patent, 3:22-29.

45. Accordingly, in view of the '821 patent's disclosure, a POSITA would have understood that the term "*entity*" includes an MPLS LSP or a PW.

VIII. IDENTIFICATION OF HOW THE CLAIMS ARE UNPATENTABLE

46. The discussion in this Declaration provides a detailed analysis of how the asserted prior art references teach each limitation of the Challenged Claims.

47. As part of my analysis, I have considered, and discuss in detail, the scope and content of the prior art and any differences between the alleged invention and the prior art. In addition to various U.S. Patents, Patent Publications, articles, and textbooks, I have reviewed and appropriately cite to a number of "Request for Comments" (RFC) documents published by the Internet Engineering Task Force (IETF). RFCs were well-known as of the priority date of the '821 patent, and it is my opinion that they were "printed publications" for the purposes for which they are cited throughout. *See, e.g.*, RFC 2026, 6, ("RFCs can be obtained from a number of Internet hosts…"), 8 (IETF documents are "readily available to a wide audience"), 26 (IETF participants "shall publicly announce…every activity" relating to the standardization process). Exhibits 1011, 1022, and 1030 are documents published by the IETF describing MPLS networking standards and improvements, and a POSITA in the field of network communications, including mesh networks and path calculation methodologies, would have reasonably relied upon such documents. POSITAs frequently cited these exact documents before the priority date of the '821 patent. Doshi, [0056], [00265] (citing RFC 3209); Vasseur '469, [0013], [0040]-[0041] (citing RFC 3209); Vasseur '391, [0011], [0040]-[0041] (citing RFC 3209); Gerö, 2:22-24 (citing RFC 3469); Anderson, [0047] (citing RFC 3272).

48. It is my opinion that the alleged invention recited in the Challenged Claims would have been obvious in view of the teachings of the asserted prior art and the knowledge of a POSITA.

A. Ground 1: The Combination of Doshi and Guichard renders obvious claims 1-7, 9-11, 13, and 17-20.

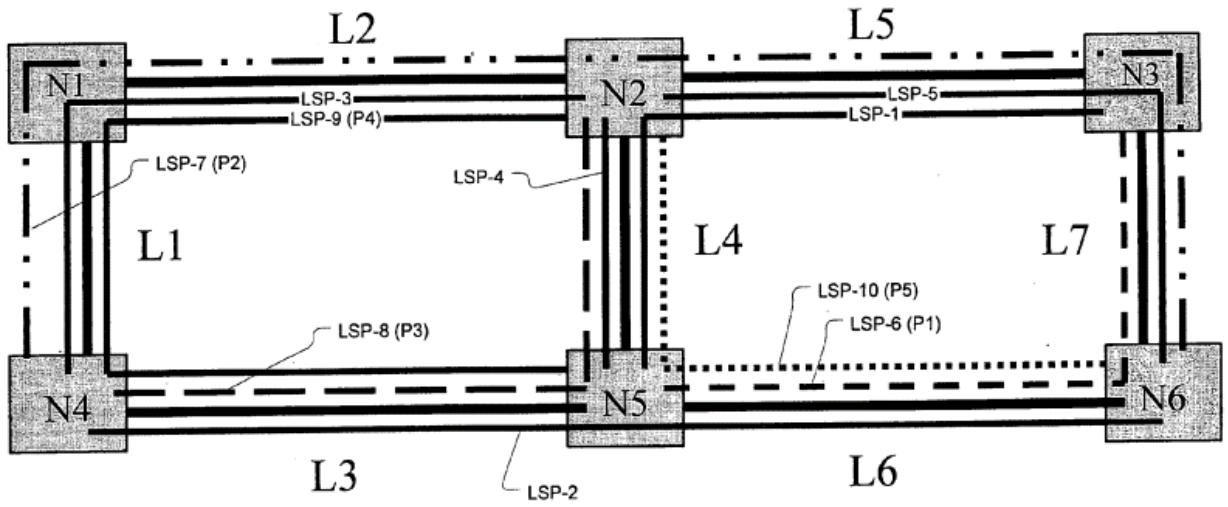
1. Summary of Doshi

49. U.S. Patent Publication No. 2004/0205239 to Doshi et al. (Ex.1005, "Doshi") was published October 14, 2004. Doshi is titled "Primary/restoration path calculation in mesh networks based on multiple-cost criteria." Doshi, Title. Doshi was not before the Examiner during prosecution of the '821 patent.

50. Like the '821 patent, Doshi recognized deficiencies in path "restoration and recovery schemes in the prior art." Doshi, [0008]; *see* '821 patent, 1:66-2:4. To address these deficiencies, Doshi describes a method performed by a "network manager" for determining and selecting an overall minimum-cost primary and protection path pair. Doshi, [0014], [0142]. Note that Doshi's "restoration path" is also referred to as a "protection path." Doshi, [0076] ("…a protection path (also known as 'restoration path')"). The determination of the primary and protection path pair considers several factors, including "link utilization, utilization threshold, administrative weight, and sharing degree." Doshi, [0142]. Another factor considered is whether the primary and protection paths are disjoint. Doshi, [0144].

Doshi explains that the benefit of disjoint paths is that "a failure affecting one of them will not affect the other." Doshi, [0040].

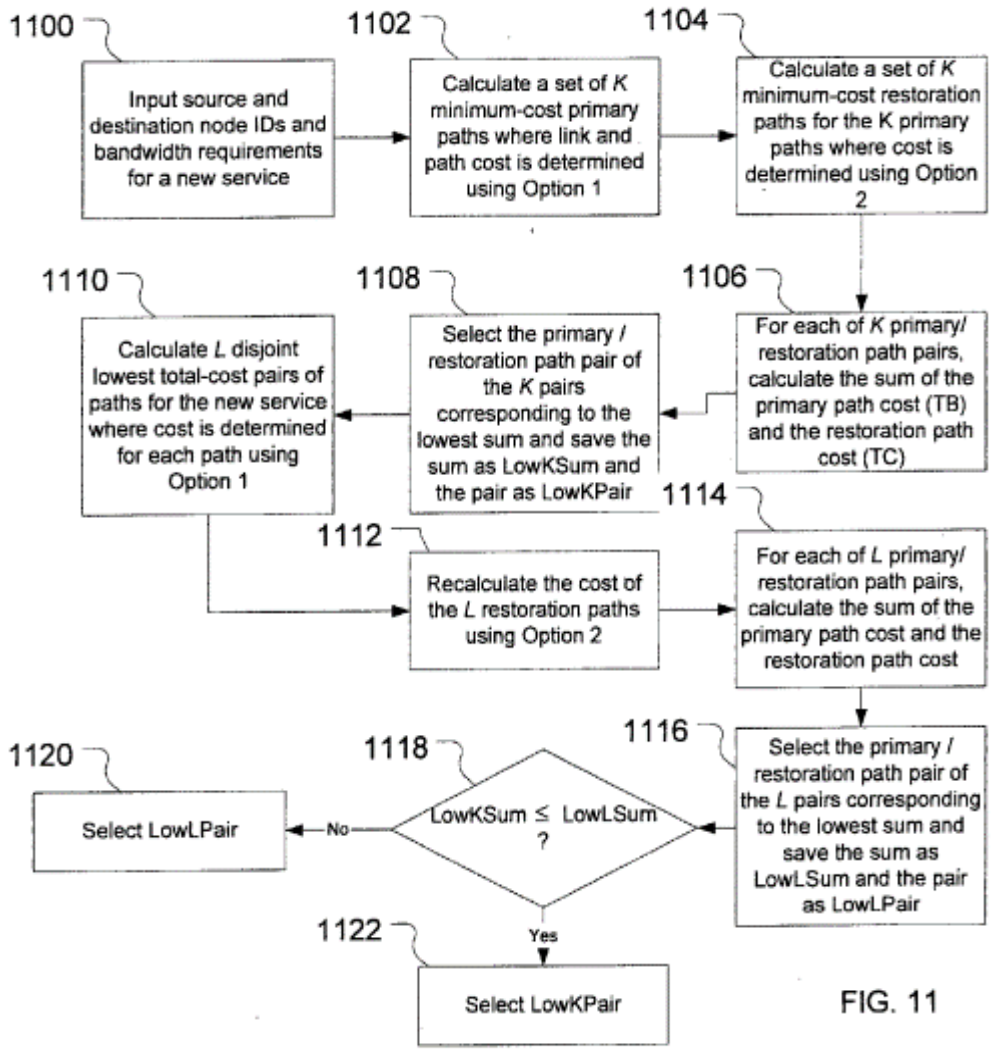
51. Doshi's concepts "are discussed in the context of exemplary shared mesh data networks" such as an "MPLS" network. Doshi, [0049]. "A mesh network is a network topology (or a subset thereof) in which there are at least two nodes having two or more paths between them." Doshi, [0048]. Figure 1, reproduced below, illustrates such an exemplary shared mesh data network where the paths connecting nodes of the mesh are MPLS Label Switched Paths ("LSPs"):



Doshi, Fig. 1

52. In Doshi's Figure 1, nodes within the mesh are connected by pairs of primary and protection LSPs. Doshi, [0052]-[0054]. For example, Node 4 is connected to Node 6 by primary LSP-2 and by protection LSP-7. Doshi, [0052]-[0054]. The protection LSPs serve as a backup path in the event of a failure of their

primary LSP. Doshi, [0051]-[0054], [0266]-[0267]. Doshi's LSPs form a plurality of candidate pairs considered in the cost analysis, an example of which is illustrated in Doshi's Figure 11:



Doshi, Fig. 11

53. Each of the plurality of candidate pairs is considered, and the overall minimum-cost pair is selected. Doshi, [0142]-[0145].

54. Accordingly, Doshi shows that selecting an overall minimal-cost MPLS LSP pair (comprising a primary and a protection path), from a plurality of LSPs between two nodes, was known before the filing of the '821 patent. Additional teachings of Doshi, and how they apply to the claims of the '821 patent, are discussed below.

2. Summary of Guichard

55. The textbook titled "Definitive MPLS Network Designs, Field-proven MPLS designs covering MPLS VPNs, pseudowire, QoS, traffic engineering, IPv6, network recovery, and multicast" is authored by Guichard et al. (Ex.1006, "Guichard"). The textbook was published by Cisco Press in 2005.

56. Like the '821 patent and Doshi, Guichard describes LSP selection within an MPLS network. Guichard, 79-81; *see* '821 patent, Abstract, 3:23-26; *see also* Doshi, [0050]-[0054], [0142]-[0145]. Guichard recognizes that network state may change, including nodes and links failing and coming back online and LSPs being added or removed. Guichard, 79. Guichard notes the advisability of detecting such changes and attempting to reoptimize LSPs along better (or more optimal) paths when they become available. Guichard, 79. Several reoptimization schemes are provided, including: manual reoptimization, timer-based reoptimization, and event-driven reoptimization. Guichard, 81. An event-driven reoptimization evaluation is triggered by, for example, by restoration or addition of a link. Guichard, 80-81.

3. Reasons to Combine Doshi and Guichard

57. A POSITA would have been motivated to combine the teachings of Doshi and Guichard, as discussed below. It would have been obvious to a POSITA after selecting MPLS LSPs for communication between two nodes, as described by Doshi, to evaluate if the selection should be reoptimized to select a different LSP or pair of LSPs in response to network changes. Guichard describes such a reoptimization evaluation of MPLS LSPs.

58. **First**, Doshi and Guichard are analogous art to the '821 patent. For example, just like the '821 patent, both references generally pertain to MPLS networks. '821 patent, 3:15-18, Abstract; Doshi, [0014], [0049]; Guichard, 79-81. Additionally, both references address the problem of selecting an MPLS LSP from available paths. '821 patent, 4:63-5:8; Doshi, [0014]; Guichard, 79-81. Thus, given the similarities between the references and the fact that they address the problem of selecting LSPs, a POSITA considering Doshi would have naturally considered the teachings of Guichard.

59. **Second**, a POSITA would have been motivated to combine the teachings of Doshi and Guichard to produce numerous predictable and beneficial results.

60. Doshi teaches performing cost calculations and selecting an MPLS LSPs pair that has an overall minimum-cost. Doshi, [0014], [0049]. Guichard

complements Doshi by teaching how to reoptimize an LSP selection. According to Guichard, the "[n]etwork state keeps changing." Guichard, 79. As a consequence, Guichard teaches that "[i]t is then **highly desirable** to detect the existence of such [changes] and reoptimize a TE LSP along a better path when it becomes available." Guichard, 79. Reoptimization evaluation, for a potentially better path, may be manually triggered by an administrator, time-based, or event-driven (e.g., change in the network such as LSP set up and tear down, restoration of a link, or new link). Guichard, 79-81.

61. A POSITA would have been motivated to perform reoptimization evaluations over time, as taught by Guichard, when implementing Doshi's pair calculation and selection techniques in order to identify a potentially better LSP (e.g., for primary or protection LSP paths) with an even lower overall minimum-cost. Doshi's initial LSP path selection identifies and selects the lowest cost pair of paths at a point in time; however, a POSITA would have recognized that it would be advantageous—indeed "highly desirable" per Guichard—to evaluate the potential for reoptimization on an ongoing basis as taught by Guichard to determine whether the selected LSP pair remains the overall minimum-cost pair or whether there is a better pair available for reoptimization. Performing repeated reoptimization evaluations as taught by Guichard furthers Doshi's stated objectives of optimization and optimal path choice. Doshi, [0172] ("In general, path selection is a powerful tool

that can be used to achieve several objectives such as … **optimization**."), [0200] ("…making an **optimal** restoration path choice…"). For instance, reoptimization would allow for selecting a shorter path if one becomes available. Doshi expressly describes attempting to identify the shortest primary and protection paths as part of optimization. Doshi, [0151], [0165], [0172].

62. The combination represents the use of a known technique (Guichard's reoptimization evaluation in response to various triggers) to improve a similar system or device (Doshi's MPLS architecture) in the same way with predictable results (ongoing optimization). The combination represents what would have been common sense to a POSITA—that it is desirable to evaluate whether a selected LSP remains optimal and to reoptimize the selection if a better option is available.

63. I also note that the results would have been predictable and there would have been a reasonable expectation of success in the combination given that Doshi and Guichard are analogous art and in the same field of endeavor. Also, a POSITA would have had a reasonable expectation of success in reoptimizing Doshi's initial LSP pair selection by performing a reoptimization evaluation because Guichard provides significant implementation details, including when to trigger reoptimization evaluations and how to "avoid network instability." Guichard, 81.

64. Making the combination would be within a POSITA's skillset since Doshi already sets forth the calculation and selection techniques to be utilized during

the reoptimization evaluation. Additionally, the combination would be a simple matter as it pertains to event-driven reoptimization evaluations since Doshi already monitors the network. Doshi, [0270]-[0281] (describing, for example, "monitoring and updating of network topology."). Doshi is already equipped to detect the event-based reoptimization evaluation triggers described in Guichard, and the reoptimization evaluation itself would utilize Doshi's own calculation and selection techniques. I also note that Doshi acknowledges that a POSITA would have had the necessary skill set to make "[v]arious modifications of the described embodiments." Doshi, [300]. It would have been obvious for such modifications to include a reoptimization evaluation as taught by Guichard.

65. Moreover, the reasonable expectation of success is evidenced by other skilled artisans that had reoptimized similar systems. Ex.1009, 136 ("Event-Drive Reoptimization…If [a] link comes up, should RtrD reoptimize its D→H tunnel so that this tunnel flows over the directly connected link?"); Ex.1010, 1 ("[T]he reoptimizer constantly monitors the network to determine if re-optimization will lead to sufficient network efficiency benefits. When sufficient benefits can be obtained, the re-optimizer computes the least cost set of connections."); Ex.1023, 34 ("To control the frequency at which tunnels with established LSPs are checked for better LSPs, use the mpls traffic-eng reoptimize timers frequency command."); Ex.1034, [0027]-[0033], claim 19 (describing that "a re-optimization operation is triggered

(for example in the case where there is a change in topology or when an LSP needs to be re-routed due to a failure)" where the re-optimization operation uses "an optimization that treats the two LSPs as a set" and where the LSP set includes a "working path" and a "protecting path."); Ex.1035, [0024] ("triggering the optimization of TE-LSPs when an event occurs.").

66. Lastly, the proposed combination relies on the teachings of the references and does not require physical incorporation of elements from Guichard into Doshi.

4. Claim 1

a. [1.0] *An entity selection method performed by a network device, comprising the steps of:*

67. As an initial matter, as analyzed in the claim construction section, the term "*entity*" includes a multiprotocol label switching ("MPLS") label switched path ("LSP") or a pseudo wire ("PW").

68. Doshi teaches a path selection method performed by a network manager:

> In one embodiment, the invention is **a <u>method</u> for determining primary and restoration paths**[1] **for a new service** in a mesh network

[1] I note that Doshi's "restoration path" is also referred to as a "protection path." Doshi, [0046] ("Unless otherwise clear from the context, the process of 'restoration'

> having a plurality of nodes interconnected by a plurality of links. For each of a plurality of candidate path pairs for the new service, each candidate path pair comprising a candidate primary path and a candidate restoration path for the new service, a path cost associated with said each candidate path pair is generated, wherein the path cost for a candidate path pair is a function of sharability of links within the corresponding candidate restoration path. **The primary and restoration <u>paths</u> for the new service are <u>selected</u> from the plurality of candidate path pairs based on the path cost of each candidate path pair**. **The <u>method may be implemented by a network manager</u> for the mesh network**.

Doshi, [0014]; *see also* Doshi, [0142] ("In the first flow, illustrated by steps 1102, 1104, 1106, and 1106, one minimum-cost path pair is **selected** from K candidate pairs by a first method."), claim 11 (…**the network manager is adapted to select** the primary and restoration paths for the new service from the plurality of candidate path pairs based on the path cost of each candidate path pair."), [0141]-[0145], [0156], [0270]-[0281], Abstract, Figs. 10, 11, 17, 18.

69. Doshi's network manager is a network device, such as a server or a node within a shared mesh data network "SMDN":

---

is synonymous in this specification with the process of 'protection.'"), [0076] ("…a protection path (also known as 'restoration path')"). In this declaration, for consistency, I will generally refer to these paths as protection paths.

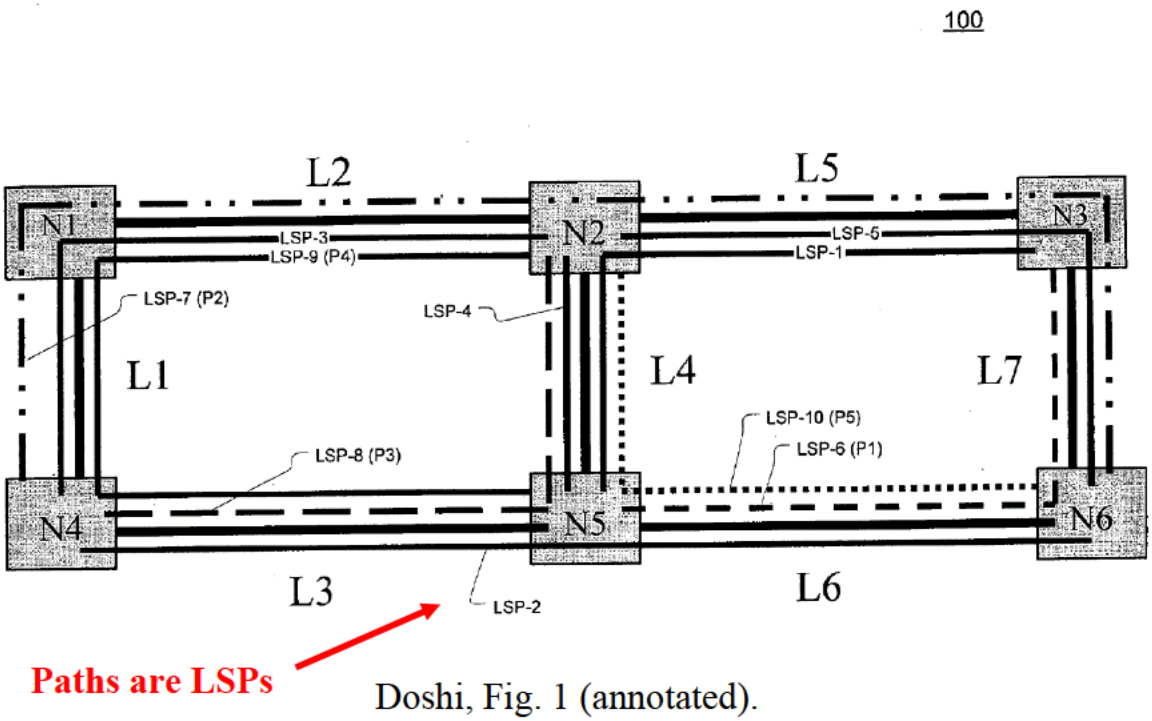
> An **SMDN can be partitioned into a variety of modules. These modules are part of a network manager that may be implemented at a centralized server or distributed partially or fully to the nodes**. Some exemplary modules, representing one specific functional partitioning of the components of an SMDN, which provide functionality such as fault detection and notification, protection switching, signaling and control, routing, and OAM, are listed below:

Doshi, [0271]; *see also* Doshi, [0082], claim 11.

70. Furthermore, Doshi discloses that its SMDN supports an MPLS architecture:

> The concepts of the present invention are discussed in the context of exemplary shared mesh data networks. However, without loss of generality, they may be applied generally to any mesh network (e.g., Sonet, ATM, frame relay, **MPLS**) that provides traffic-switching control at the nodes incident to the links in the network. A mesh data network is a network in which data is communicated asynchronously between nodes in packetized form and it is a specific case of the more general mesh network. In the present example, **SMDN 100 supports the <u>multiprotocol label switching (MPLS)</u> architecture** standard per Rosen, E., et al., "RFC3031—Multiprotocol Label Switching Architecture," Network Working Group, The Internet Society, January 2001, incorporated herein by reference in its entirety; however, as would be understood by one skilled in the art, other data technologies that offer similar data traffic engineering features (e.g., ATM and frame relay) could be substituted for MPLS.

Doshi, [0049]. Doshi's Figure 1, reproduced below, illustrates how the SMDN 100 supports an exemplary MPLS architecture provisioned with multiple LSPs. Doshi, [0016], [0052]-[0054].



Doshi, Fig. 1 (annotated).

71. Since Doshi's SMDN 100 has an MPLS architecture, it would have been obvious to a POSITA that Doshi's "paths" correspond to LSPs because LSPs are the paths used in MPLS networks.

72. Therefore, Doshi teaches a path (e.g., MPLS LSP) selection method performed by a network manager, which discloses "*[a]n entity selection method performed by a network device*," as recited in the preamble.

73. As I further explain below, the prior art renders obvious all of the claimed method steps ("*comprising the steps of:*").

b. [1.1] *providing a plurality of multi protocol label switching (MPLS) transport entities between a first endpoint and a second endpoint;*

74. Consistent with the analysis in the claim construction section, "*multi protocol label switching (MPLS) transport entities*" include LSPs of an MPLS network:

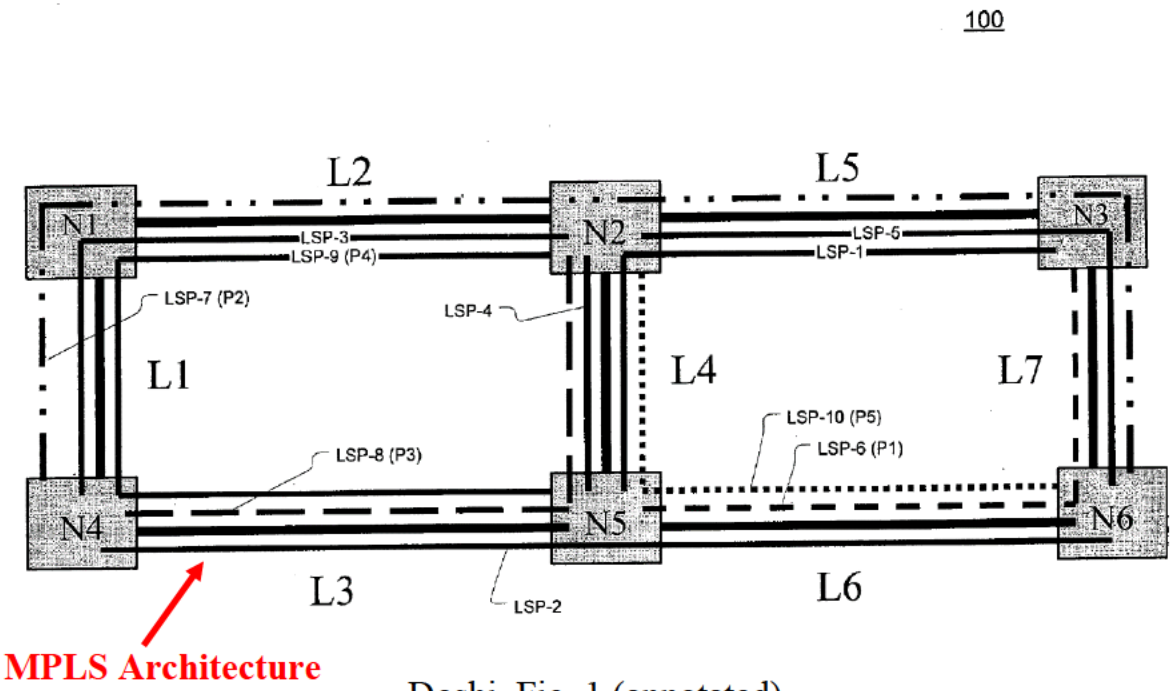
> An exemplary embodiment may include a network with a set of **MPLS-TP transport entities, for example, label switch paths (LSPs)** or pseudo wires (PWs) between a first endpoint and a second endpoint. It is preferable that an operator will define the entity set such that, with high probability, at least one entity will stay up in the event of multiple span or node failures.

'821 patent, 3:22-29.

75. **First**, as already analyzed above (see preamble [1.0]), Doshi at Figure 1 discloses an SMDN that supports an MPLS architecture:

> **In the present example, SMDN 100 supports the <u>multiprotocol label switching (MPLS) architecture</u>** standard per Rosen, E., et al., "RFC3031—Multiprotocol Label Switching Architecture," Network Working Group, The Internet Society, January 2001, incorporated herein by reference in its entirety.

Doshi, [0049].



Doshi, Fig. 1 (annotated).

76. In the MPLS architecture shown above, five primary LSPs (1-5) and five protection LSPs (6-10) are provisioned:

> FIG. 1 illustrates the **five primary LSPs** as follows: LSP-1 carries traffic along path N3-N2-N5, LSP-2 carries traffic along path N4-N5-N6, LSP-3 carries traffic along path N2-N1-N4, LSP-4 carries traffic along path N2-N5, and LSP-5 carries traffic along path N2-N3-N6.

Doshi, [0052].

> FIG. 1 also illustrates **five exemplary protection paths** LSP- 6 to LSP-10 (also of unit bandwidth in this example) that have been reserved to use for protection of services given a failure of one of the primary paths. LSP-6 (along path N3-N6-N5) protects LSP-1, LSP-7 (along path N4-N1-N2-N3-N6) protects LSP-2, LSP-8 (along path N2-N5-N4) protects

> LSP-3, LSP-9 (along path N2-N1-N4-N5) protects LSP-4, and LSP-10 (along path N2-N5-N6) protects LSP-5.

Doshi, [0054]; *see also* [0050]-[0053].

77. Doshi teaches that the LSPs are formed "**between two nodes**" that correspond to "**source and destination nodes**" or "**end nodes** of an LSP." Doshi, [0050], [0154], [0267].

> In general, each connection or service in the network is carried by a primary path that (1) **connects the source and destination nodes** and (2) provides the necessary bandwidth for the service.

Doshi, [0154]; *see also* Doshi, [0156].

> As discussed above, after detecting a failure, **end nodes of an LSP** switch traffic from a primary (i.e., working) LSP to its corresponding protection LSP. Note that both working and protection LSPs typically enter the ingress node and exit the egress node using different ports.

Doshi, [0267].

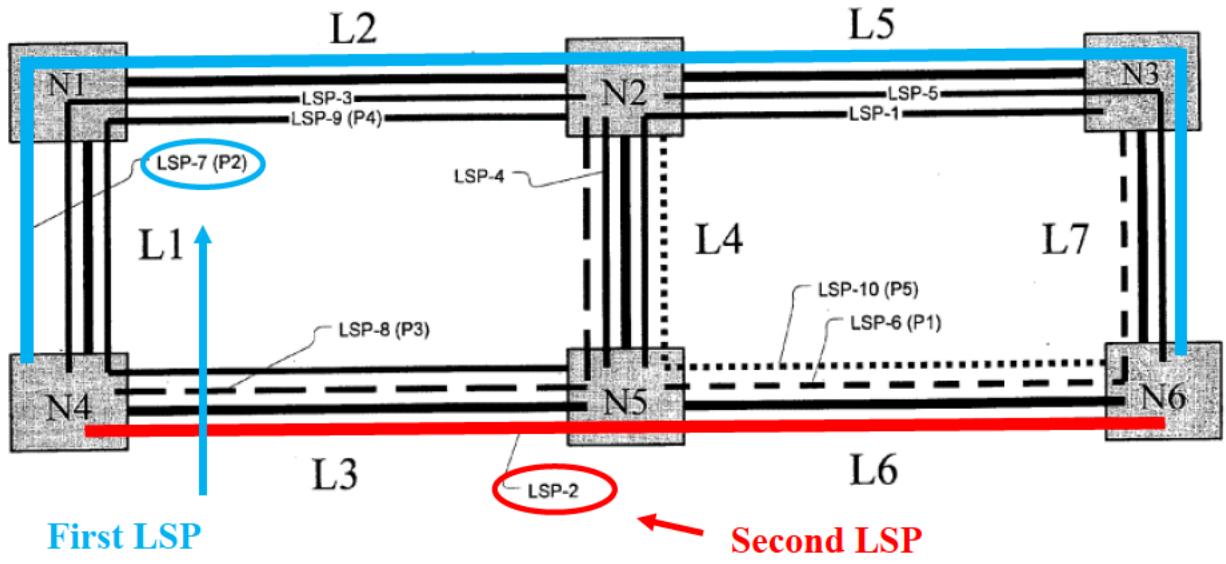
> As illustrated, SMDN 100 has been provisioned with five primary label-switched paths (LSPs) LSP-1 to LSP-5. **A label-switched path is effectively a tunnel between two nodes** that carries service traffic according to a predetermined route.

Doshi, [0050]; *see also* Doshi, [0056], [0142], [0188], [0190], [0192], [0195], Fig. 5, Fig. 8, Fig. 11, Fig. 17, Fig. 18, and Fig. 20.

78. Since Doshi's LSPs are provided "*between*" and terminate at the two end nodes, a POSITA would have understood that one end node (e.g., the source

node) corresponds to "*a first endpoint*" and that the other end node (e.g., the destination node) corresponds to "*a second end point*." Any two LSPs (or more as discussed below) that are provided between two end nodes correspond to "*a plurality of multi protocol label switching (MPLS) transport entities.*"

79. For reference, I have highlighted an example at Figure 1 of providing a primary LSP-2 and a protection LSP-7 between end node N4 and end node N6:



Doshi, Fig. 1 (annotated).

80. Importantly, although Figure 1 only illustrates two LSPs between a node pair, Doshi makes clear that this figure is merely "**exemplary**" and expressly teaches "**two or more paths between**" the nodes:

> FIG. 1 illustrates an **exemplary** shared mesh data network (SMDN) according to one embodiment of this invention.

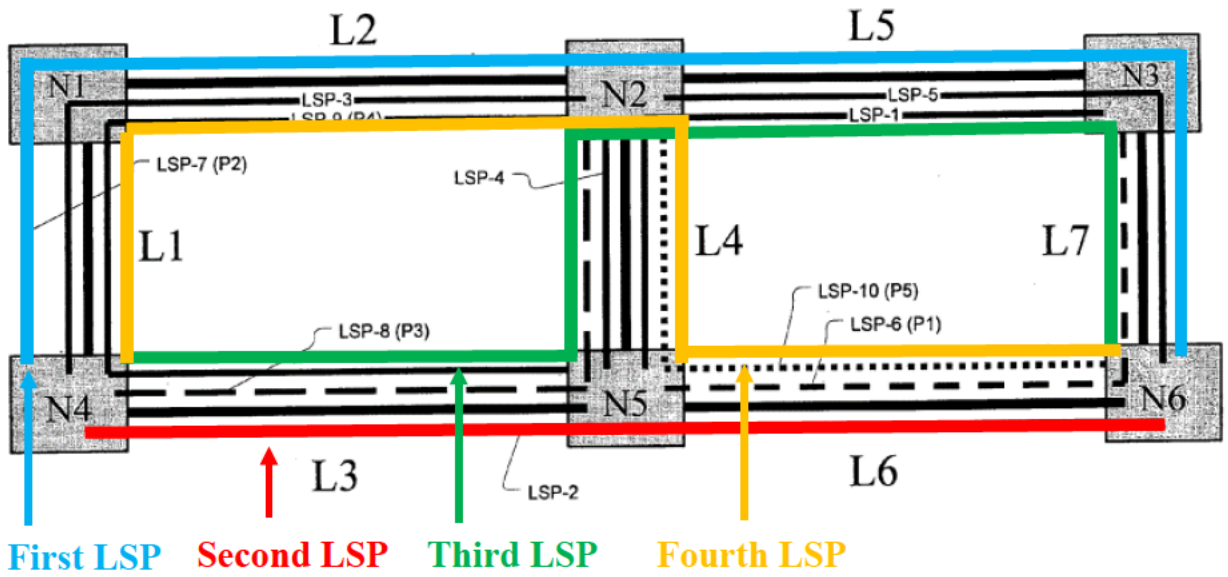
Doshi, [0016]; *see also* Doshi, [0048]-[0055] (describing Fig.1 as exemplary)

> FIG. 1 illustrates an exemplary shared mesh data network (SMDN) 100 according to one embodiment of this invention. A mesh network is a network topology (or a subset thereof) in which there are at least two nodes having **two or more paths between** them.

Doshi, [0048]. Doshi further contemplates making "**[v]arious modifications**" to the exemplary embodiments. Doshi, [0300].

81. Since Figure 1 is merely exemplary and subject to modification, and since Doshi expressly teaches "**two or more paths between**," a POSITA would have found it obvious to provide additional LSPs between each node pair. Doshi, [0048]. That is, it would have been obvious to a POSITA to provide a sufficient number of LSPs between two nodes to meet bandwidth requirements for communication between the nodes. Doshi contemplates more than two LSPs between two nodes by describing "two or more paths between" the end nodes and that there are a "plurality of candidate path pairs" from which to select. Doshi, [0048], Abstract; *see also* Doshi, Abstract ("[S]electing the primary and restoration paths for the new service from the plurality of candidate path pairs"), [0014]. This would mean that each node pair has at least two primary LSPs and two protection LSPs, with different primary and protection LSP combinations corresponding to candidate pairs. *See* Doshi, [0014], ("…each candidate path pair comprising a candidate primary path and a candidate restoration path…"); *see also* analysis at sections [1.2]-[1.5].

82. Below in modified Figure 1, I illustrate an example of providing two primary LSPs and two protection LSPs between nodes N4 and N6:



Doshi, Fig. 1 (modified and annotated).

83. A POSITA would have recognized that providing more than two LSPs between each node pair (e.g., at least two primary LSPs and two protection LSPs) would be beneficial since doing so facilitates path diversity to reduce the probability of service interruption between the nodes. For example, the illustrated third and fourth LSPs between N4 and N6 offers additional protection against failures along different paths, at least in part. A concurrent failure of, for example, L2 and L6 would cause both the first and second LSP (the original two LSPs illustrated in Doshi) to go down; however, such a failure would not bring down the third LSP. Having the third LSP available would allow traffic to be switched over with minimal interruption of service as compared to if the first and second LSPs were the only

LSPs connecting N4 and N6. Having the fourth LSP available in the event of concurrent failure of, for example, L3 and L7 (which would affect both the first and second LSP without affecting the fourth LSP) would similarly allow traffic to be switched over to the fourth LSP with minimal interruption of service.

84. Another advantage of having more than two LSPs (e.g., more than just one primary and one protection LSP) between each node pair is that it allows traffic with different quality of service (QoS) requirements to be allocated and aggregated on different LSPs. *See* Doshi, [0265] (describing assigning QoS characteristics to LSPs), [0272] (describing provisioning paths with different QoS). For example, traffic with a high QoS requirement, e.g., video traffic, would be assigned to an LSP configured to handle such high QoS traffic while traffic with a lower QoS requirement would be assigned to a different LSP.

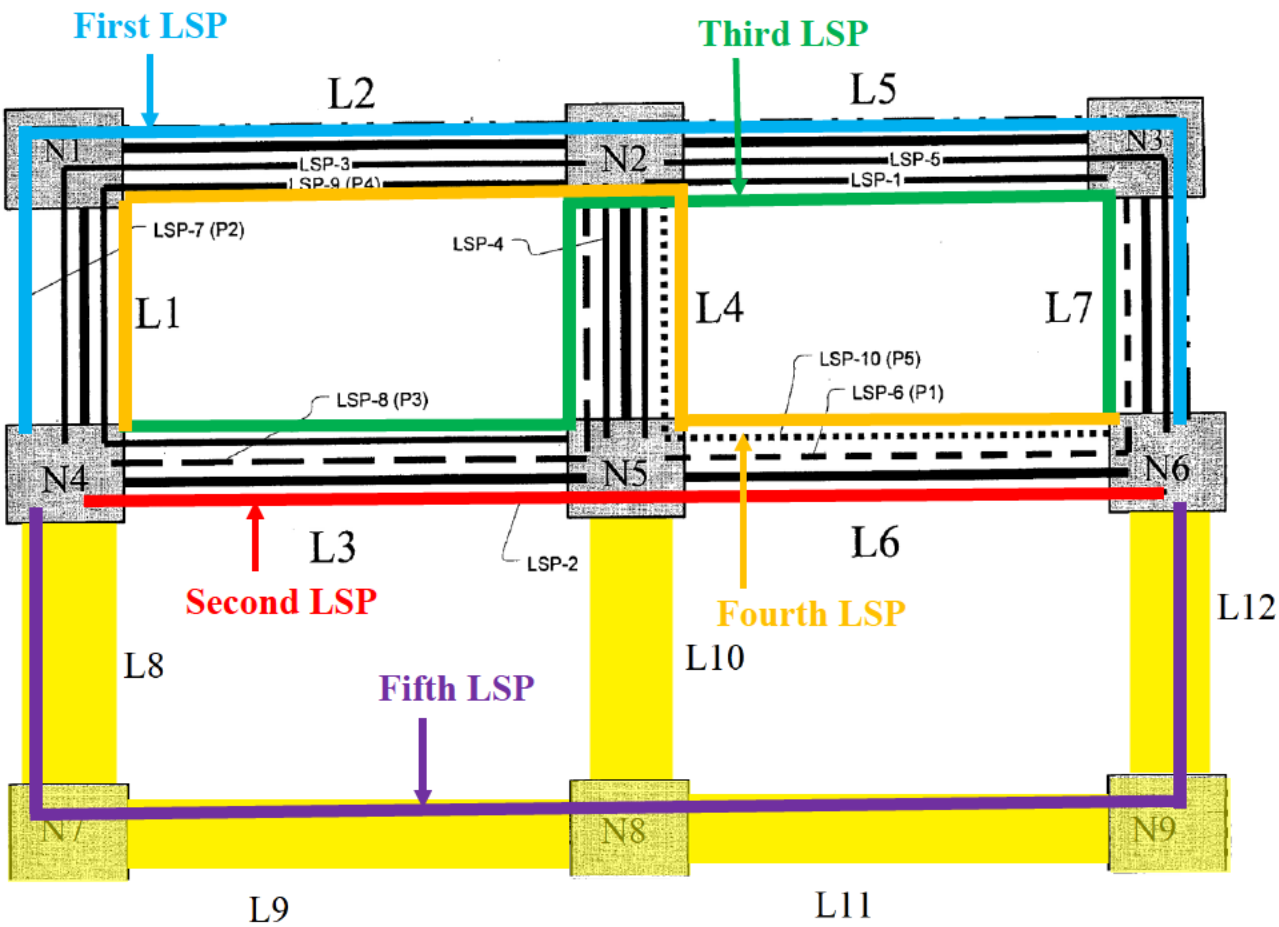
85. Moreover, I note that providing more than two LSPs between each node pair also facilitates reoptimization. Consistent with the below analysis at elements [1.3]-[1.5], it was known in the art that it is "highly desirable" to reoptimize LSP selection over time (e.g., based on a manual trigger, time-based, or event-driven). Guichard, 79-81. When reoptimizing for the best LSPs between two end nodes, a POSITA would have understood to consider all possible LSP paths and choose the optimal LSP pair (e.g., primary and protection LSPs). Doshi, [0172] ("In general, path selection is a powerful tool that can be used to achieve several objectives such

as … optimization."), [0200] ("…making an optimal restoration path choice…"). If only one primary LSP and one protection LSP were provided between two end nodes, then reoptimization would not be possible in some failure situations. Thus, facilitating reoptimization is another benefit of providing more than two LSPs between each node pair in Doshi.

86. I note that the modified example of Figure 1 is not intended to be limiting and it would be understood that there may be additional nodes with corresponding LSPs not illustrated. *See e.g.,* Doshi, [0055] (Describing that the example is not limiting and that there are cases where other paths and nodes may exist). Also, per Guichard, it would have been understood that the network may change such that new nodes with new LSPs may be added. *See e.g.,* Guichard, 79 ("**Network state keeps changing**. Links and nodes fail and recover … new TE LSPs are set up…"). A POSITA would have been motivated to provide, as the network changes, additional LSPs along new links between new nodes in Doshi so that the paths between a node pair are completely disjointed, thereby further reducing the probability of service interruption. Further support for the combination is provided in Section VIII.A.3 above.

87. I have provided an example of such an implementation below, illustrating additional nodes N7-N9 with new links L8-L12 that provide

corresponding a new Fifth LSP between nodes N4 and N6 that is completely disjoint with respect to the First and Second LSPs:



Doshi, Fig. 1 (modified and annotated).

88. Providing more than two LSPs between a node pair would have been consistent with other prior art implementations. *See* MPLS Traffic Engineering Article, 1305 (disclosing six LSPs between an ingress node and an egress node). Accordingly, a POSITA would have had a reasonable expectation of success in implementing more than two LSPs between two end nodes (as shown at the modified

Figure 1) since Doshi itself suggests this implementation and others had successfully provided up to six LSPs between node pairs.

89. Therefore, Doshi in combination with Guichard discloses providing a plurality of MPLS LSPs (e.g., at least two primary LSPs and at least two protection LSPs) between a first end node (e.g., source node) and a second end node (e.g., destination node), which renders obvious "*providing a plurality of multi protocol label switching (MPLS) transport entities between a first endpoint and a second endpoint*," as claimed.

c. [1.2] *determining an overall cost for each entity pair of said plurality of entities;*

90. As an initial matter, the '821 patent describes that the overall cost for an entity pair may be determined based on various factors, including entity cost resulting from the sum of link metrics along the entity:

> An overall cost for each pair of the plurality of transport entities is determined, as shown by block 220. **The overall cost may be determined based on** the probability of simultaneous failure of the working and protection entities (SFP), as described below, the entity cost, or a combination of SFP and entity cost. The entity cost may be defined as the sum of the link metrics, for example IGP or TE metrics, along the entity.

'821 patent, 4:14-21.

91. **First**, as discussed above at [1.0]-[1.1], Doshi in combination with Guichard discloses providing a plurality of MPLS LSPs (e.g., at least two primary and at least two protection LSPs), which corresponds to "*said plurality of entities*."

92. **Second**, Doshi discloses determining the cost of each path (e.g., each primary or protection LSP) by calculating the cost for each link along the path:

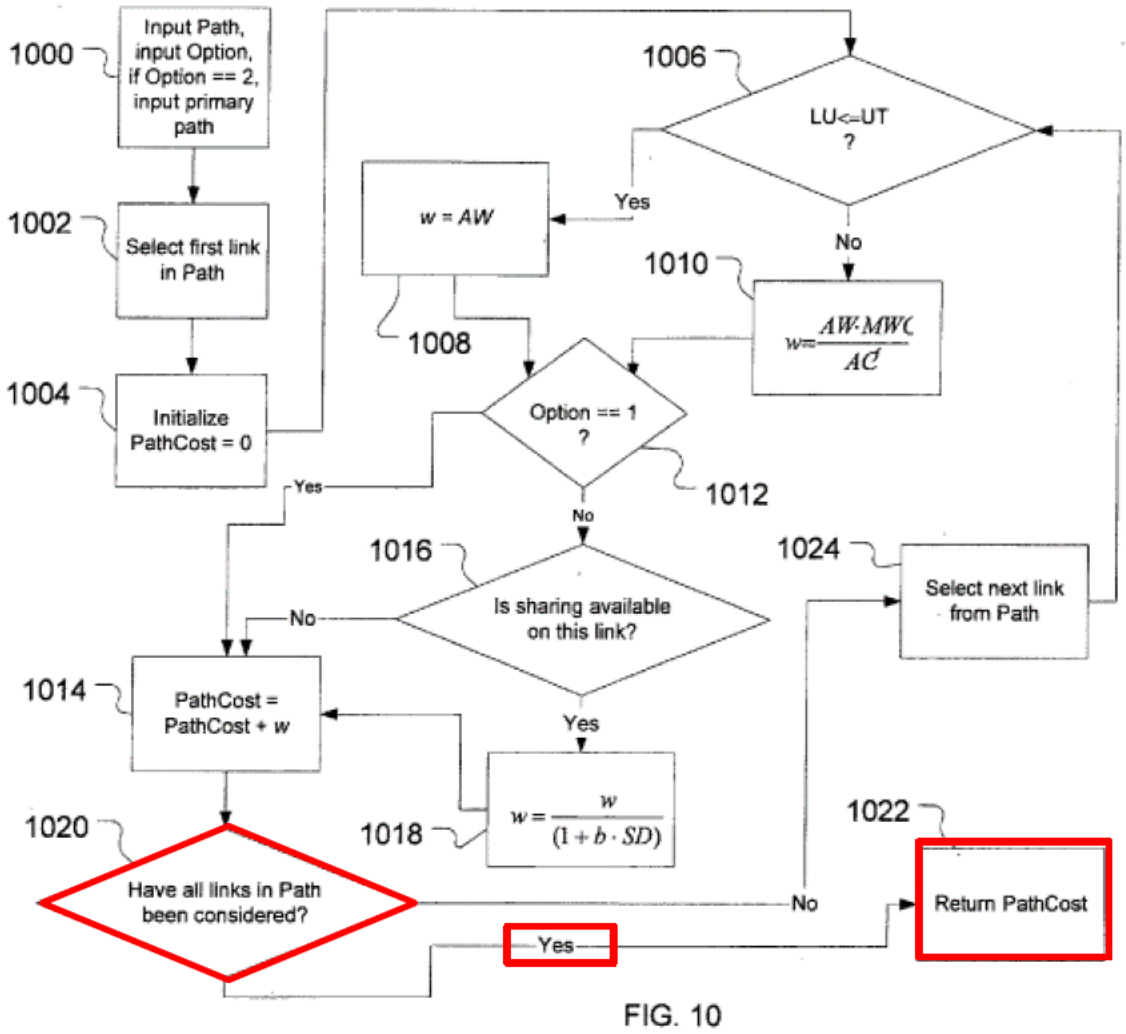


FIG. 10

Doshi, Fig. 10 (annotated).

93. Doshi explains that "path cost is considered to be the sum of the link costs." Doshi, [0125]. This is consistent with the '821 patent's definition of the "entity cost" on which determination of "*overall cost*" is based. Since Doshi's paths are LSPs (see analysis at [1.0]-[1.1]) and since LSPs correspond to "*entities*" in the '821 patent, a POSITA would have understood that the path cost described in Doshi corresponds to an "entity cost" in the '821 patent. As discussed below, Doshi's path cost is used to determine the cost of each candidate pair. In other words, the cost of each of Doshi's pairs is based on path cost (just like the '821 patent).

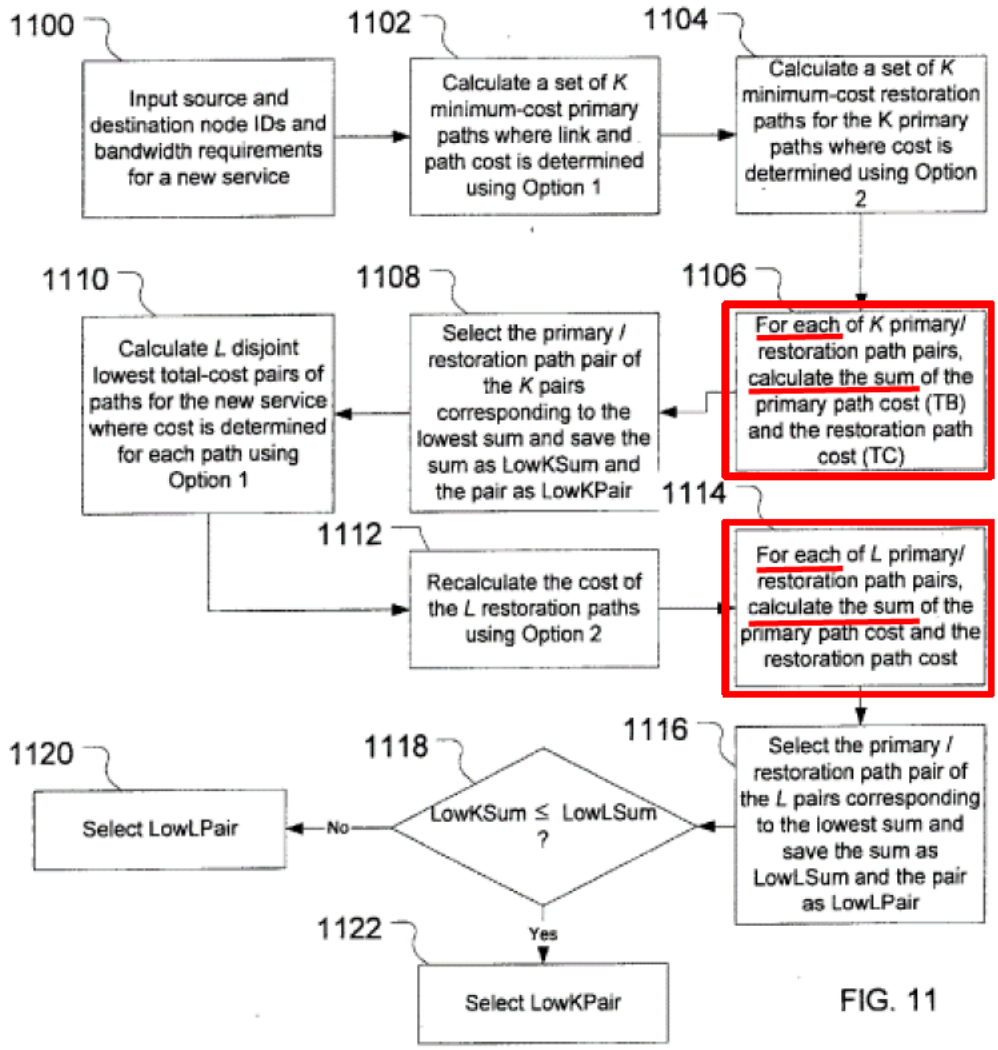
94. Further, Doshi discloses determining a cost associated with each candidate path pair (e.g., a primary path and a protection path):

> **<u>For each of a plurality of candidate path pairs</u> for the new service, each candidate path pair comprising a candidate primary path and a candidate restoration path for the new service, a path <u>cost associated with said each candidate path pair is generated</u>**, wherein the path cost for a candidate path pair is a function of sharability of links within the corresponding candidate restoration path.

Doshi, [0014]; *see also* Doshi, Abstract, [0026], [0046], [0050]-[0054], [0076], [0082], [0130], [0124]-[0181], Fig. 1, Fig. 10, Fig. 11, Fig. 12, and Fig. 13.

95. In the context of Figure 11 (building upon Figure 10's example for determining individual path cost), Doshi provides an example where the cost of each

candidate pair is determined by summing a primary path cost and a protection path cost to identify an overall-minimum cost pair:



Doshi, Fig. 11 (annotated).

The method involves two major flows. In the first flow, illustrated by steps 1102, 1104, 1106, and 1106, one minimum-cost path pair is selected from K candidate pairs by a first method. In the second flow, illustrated by 1110, 1112, 1114, and 1116, one minimum-cost path pair is selected from L candidate pairs by a second method. **Then in step**

> **1118, and steps 1120 or 1122, the lowest-cost pair of these two lowest-cost results is selected as <u>the overall minimum-cost pair</u>.**

Doshi, [0142].

96. Summing the cost of a primary path and a protection path determines the cost of the pair as a whole. As discussed above, Doshi's cost calculation is performed for each possible pair. A POSITA would have understood that this corresponds to the '821 patent's determination of "*overall cost for each entity pair of said plurality of entities*" based on "entity cost."

97. As discussed above, the '821 patent's "*overall cost*" can also be determined based on probability of simultaneous failure. Doshi's cost determination takes into account whether paths are "**disjoint**,*"* where disjoint paths would be understood as lowering the probability of simultaneous failure. Doshi, [0146]-[0156], Fig. 11 (step 1110 "Calculate L **disjoint** lowest total-cost pairs of paths"). The cost determination also "consider[s]…link utilization, utilization threshold, administrative weight, and sharing degree." Doshi, [0142]. A POSITA would have understood that Doshi's cost determination—which considers a myriad of factors such as link utilization, sharing degree, and path disjointedness—represents the "*overall cost"* for each candidate path pair.

98. Consistent with the analysis at [1.0]-[1.1], it would have been obvious to a POSITA for the candidate path pairs to be implemented as LSPs, since Figure 1

supports an MPLS architecture and provides a plurality of LSPs. Also, a POSITA would have found it obvious for the candidate path pairs to be selected from the provided plurality of MPLS LSPs (at least two primary and at least two protection LSPs) between the end nodes (see analysis at [1.1]), because these LSPs would have already been in existence and available for immediate use for the new service.

99. By determining the cost for the provided LSPs, assessment can be made whether the provided LSPs have sufficient resources for the new service—without having to allocate additional resources. Considering the provided LSPs for the new service would allow such LSPs to be more fully utilized. MPLS Traffic Engineering Article at 1306 (describing the known problem of underutilized LSPs and solution of allocating traffic to such LSPs).

100. Consistent with the analysis at [1.3]-[1.5] below, in view of Guichard's teaching of ongoing reoptimization efforts, it would have been an obvious aspect of path selection to perform Doshi's cost calculations repeatedly to attempt to determine whether the original LSP pair still has the overall minimum-cost or if there is another LSP pair that is more optimal. A POSITA would have understood that the cost calculations for such reoptimization efforts would consider existing paths, including, for example, an originally selected LSP pair that is currently being utilized.

101. Therefore, Doshi in combination with Guichard discloses determining an overall cost for each LSP pair (e.g., a primary and a protection LSP) of the plurality of MPLS LSPs, which renders obvious "*determining an overall cost for each entity pair of said plurality of entities*," as claimed.

d. [1.3] *selecting an entity pair from said plurality of transport entities based at least in part upon said overall cost; and*

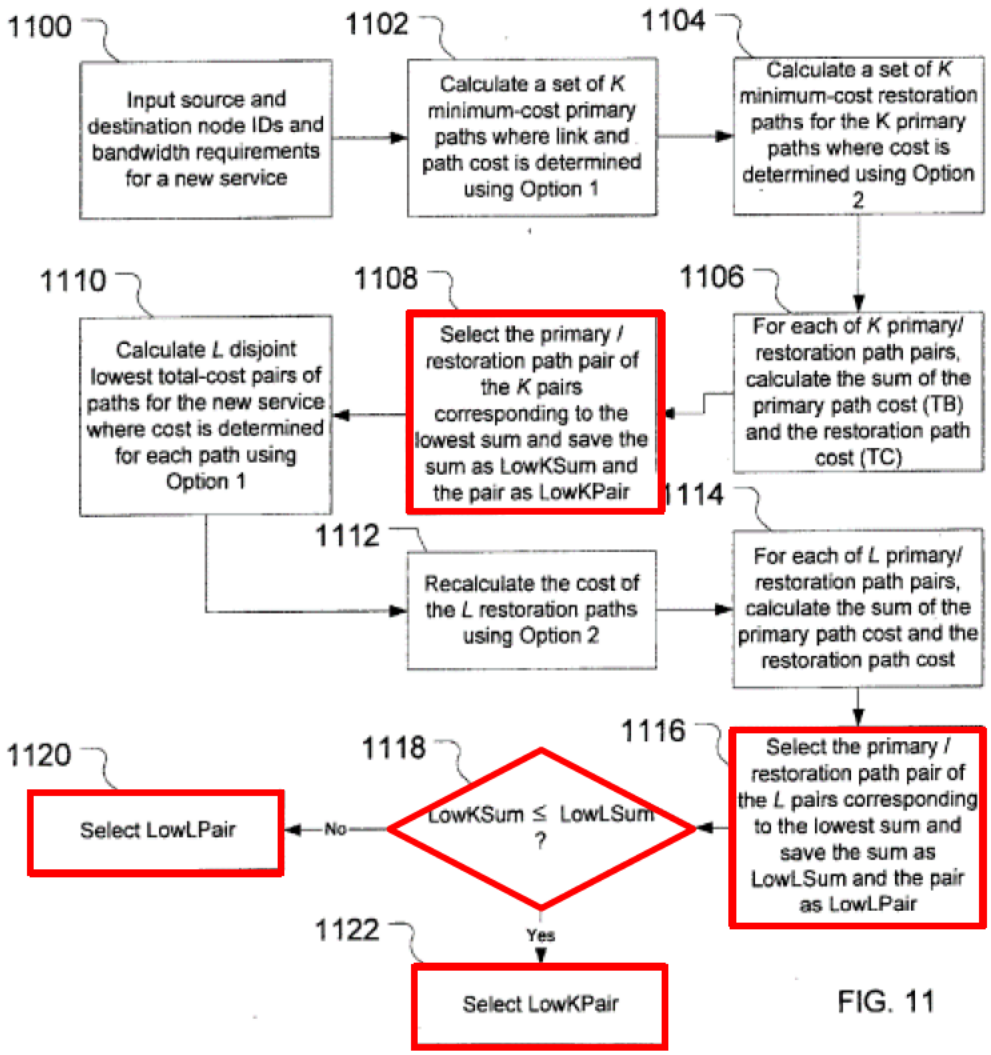
102. First, as discussed above at [1.1]-[1.2], Doshi in combination with Guichard determines the "*overall cost*" for each LSP pair (e.g., a primary and a protection LSP) of the plurality of MPLS LSPs ("*said plurality of transport entities*").

103. Second, Doshi teaches that a primary and a protection pair are selected from the plurality of paths of the MPLS network:

> For each of a plurality of candidate path pairs for the new service, each candidate path pair comprising a candidate primary path and a candidate restoration path for the new service, a path cost associated with said each candidate path pair is generated, wherein the path cost for a candidate path pair is a function of sharability of links within the corresponding candidate restoration path. **The <u>primary and restoration paths</u> for the new service are <u>selected from the plurality of candidate path pairs</u> based on the path cost of each candidate path pair. The method may be implemented by a network manager for the <u>mesh [MPLS] network</u>**.

Doshi, [0014]; *see also* Doshi, Abstract, [0046], [0076], [0142]-[0145], Claim 1, Claim 11. Consistent with the analysis at [1.1]-[1.2], it would have been obvious for the "plurality of candidate path pairs" (pairs of primary and protection LSPs) to correspond to the provided plurality of LSPs of the MPLS architecture. Doshi, [0014]. Accordingly, Doshi renders obvious "*selecting an entity pair from said plurality of transport entities*."

104. Doshi further teaches that the primary and protection path pair is selected based, at least in part, on the overall cost. More particularly, Doshi's Figure 11 and accompanying description teach selecting the overall minimum-cost pair:



Doshi, Fig. 11 (annotated).

> FIG. 11 depicts and exemplary **method for calculating a minimum-cost primary/restoration path pair** for a new service in consideration of link utilization, utilization threshold, administrative weight, and sharing degree.

Doshi, [0142].

> The method involves two major flows. In the first flow, illustrated by steps 1102, 1104, 1106, and 1106, one minimum-cost path pair is selected from K candidate pairs by a first method. In the second flow,

> illustrated by 1110, 1112, 1114, and 1116, one minimum-cost path pair is selected from L candidate pairs by a second method. **Then in step 1118, and steps 1120 or 1122, the lowest-cost pair of these two lowest-cost results is <u>selected as the overall minimum-cost pair</u>.**

Doshi, [0142]; *see also* Doshi, [0026], [0125]-[0139], [0143]-[0145].

105. Accordingly, Doshi renders obvious that the selection is "*based at least in part upon said overall cost*."

106. Third, in view of Guichard's teaching of ongoing reoptimization efforts, it would have been obvious to perform Doshi's cost calculations repeatedly over time to attempt to determine whether the originally selected LSP pair still has the overall minimum-cost or if another LSP pair should be selected. A POSITA would have understood that the cost calculations for such reoptimization efforts would consider existing LSP paths, including, the originally selected LSP pair that had the overall minimum-cost. If the originally selected LSP pair still has the overall-minimum cost, a POSITA would have understood that Doshi's selection techniques identified above would reselect the original pair. On the other hand, if another LSP pair is determined to have the overall-minimum cost, then that LSP pair would be selected. Each such selection is an example of "*selecting an entity pair from said plurality of transport entities based at least in part upon said overall cost*."

107. In more detail, Guichard teaches attempting to reoptimize LSP selection on an ongoing basis. Specifically, Guichard teaches it is highly desirable

to detect the existence of better a path that is available and reoptimize the LSP selection accordingly:

> Network state keeps changing. Links and nodes fail and recover; new TE LSPs are set up while others may be torn down. Consequently, for any TE LSP, a more optimal path may appear in the network. **It is then highly desirable to detect the existence of such a path and reoptimize a TE LSP along a better path when it becomes available**.

Guichard, 79.

108. Guichard teaches that reoptimization evaluation may be, for example, manual and timer-based:

> When a reoptimization evaluation process is triggered by R6, it determines that a **more optimal (shortest) path** exists between R6 and R5 (the path R6-R2-R3-R4-R5), and it reroutes the TE LSP along this more optimal path.

Guichard, 80-81.

- **Manual reoptimization**—A command is issued on the headend router that triggers a reoptimization evaluation for each TE LSP the router is the headend for. If a **better path** is found, the TE LSP is reoptimized according to the make-before-break procedure.
- **Timer-based reoptimization**—A reoptimization timer is configured on the headend router. Upon expiration, the headend tries to reoptimize each TE LSP it is the headend for.

Guichard, 81; *see also* Guichard, 79.

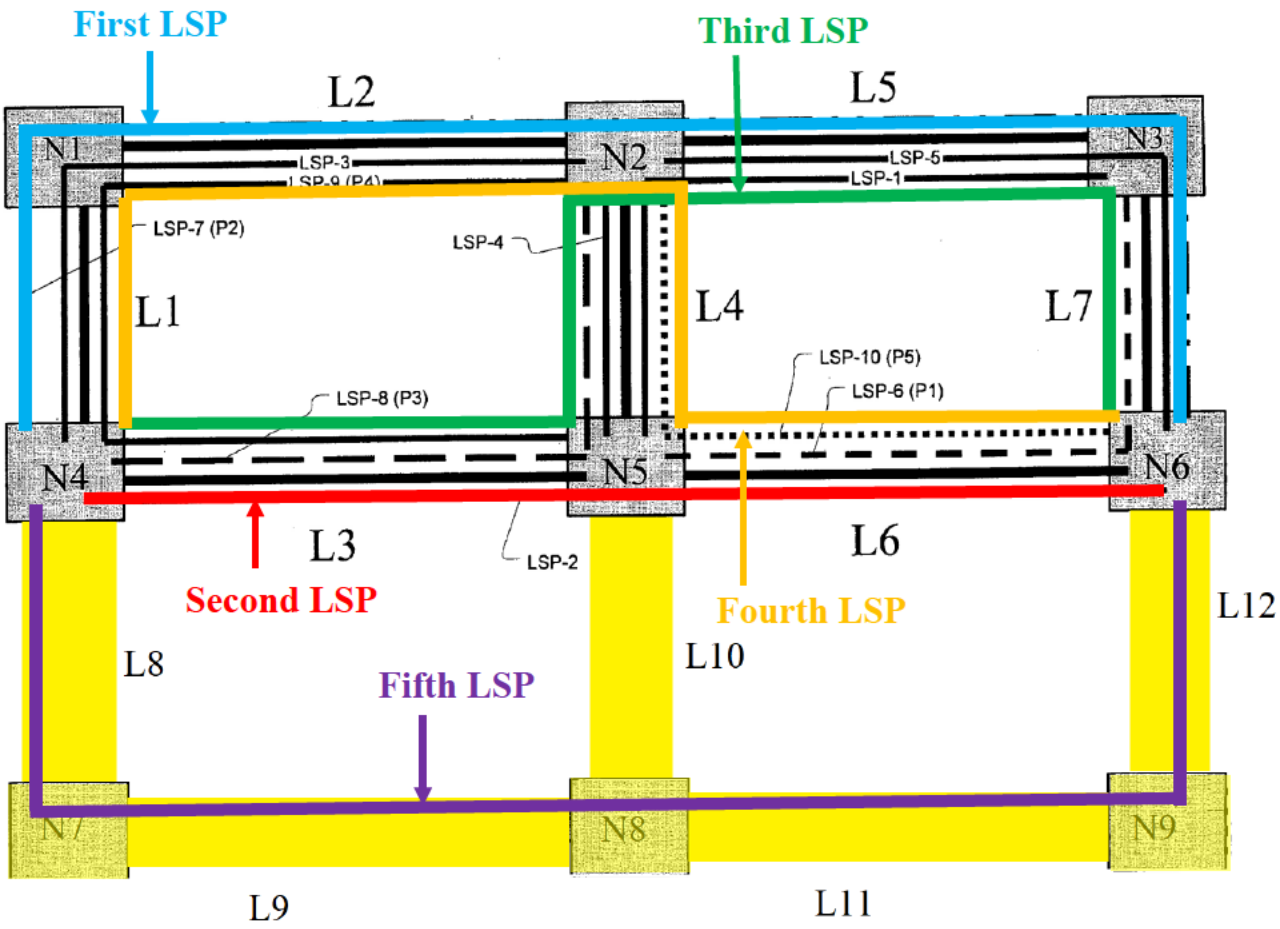
109. Guichard merely describes what would have been common sense to a POSITA—reoptimizing a system such that the LSP or LSP pair used is the best available. In other words, a POSITA would have understood that Guichard teaches that original path selections (e.g., LSP pair selection) should be evaluated on an ongoing basis to determine whether better or more optimal paths exist for the LSP pair.

110. It would have been obvious to a POSITA, in view of Guichard, to perform ongoing reoptimization attempts, when implementing Doshi's MPLS architecture, so that an optimum LSP pair (e.g., an overall minimum-cost pair) is selected over time for communication between two end nodes. In implementing the combination, a POSITA would have evaluated all existing available LSPs (including previously selected LSPs) by repeating Doshi's candidate pair cost determination (see analysis at [1.2]) and selection techniques in response to a manual trigger or expiration of a timer to thereby select the LSP pair with an overall minimum-cost.

111. A POSITA would have been motivated to perform ongoing reoptimization to identify potentially better primary and protection LSP paths (for the LSP pair between two end nodes) that minimize the overall cost. *See e.g.,* Doshi, [0172] ("In general, path selection is a powerful tool that can be used to achieve several objectives such as … **optimization**."), [0200] ("…making an **optimal** restoration path choice…"). For instance, this would allow for selecting the shortest

path during reoptimization if available. Doshi expressly describes attempting to identify the shortest primary and protection paths as part of optimization. Doshi, [0151], [0165], [0172]. Guichard also recognizes that a shorter path is more optimal than a longer path. Guichard, 80. Other potential benefits would have been apparent and motivating to a POSITA as well, such as finding a path with more available bandwidth or finding a path with fewer nodes or links in common with the other path in Doshi's pair of paths. *See* Doshi, [0158] (describing the advantages of disjoint paths). Further support for the combination is provided in Section VIII.A.3 above.

112. Below, in the context of Doshi's modified Figure 1, I provide an example of selecting an LSP pair, based on the combined teachings:



Doshi, Fig. 1 (modified and annotated).

113. Take, for instance, where an original LSP pair (e.g., First LSP and Second LSP) was being utilized for communications between end nodes N4 and N6. In response to a manual trigger or expiration of a timer, as Guichard teaches, potential reoptimization of the LSPs would take place to determine if there is a better or more optimal LSP. All things being equal, in the circumstance where the links of the First LSP are heavily utilized as compared to the links of the Fifth LSP, it would have been obvious for Doshi's cost determination to result in a new pair (e.g., the Fifth LSP as the primary and the Second LSP as the protection) as having the overall

minimum cost. In such a circumstance, these LSPs would be selected for further communications between end nodes N4 and N6 while the First LSP would be replaced by the Fifth LSP as the primary. Of course, this is just an example, and given network status it would be understood that other LSPs may be selected as the overall minimum cost pair.

114. Therefore, Doshi in combination with Guichard discloses selecting an LSP pair (e.g., a primary and a protection LSP) from the plurality of existing MPLS LSPs based on the overall cost, which renders obvious "*selecting an entity pair from said plurality of transport entities based at least in part upon said overall cost*," as claimed.

e. [1.4] *if an entity pair reselection event occurs, reselecting said entity pair from the group consisting of said entity pair and a replacement entity pair comprising at least one entity distinct from the entities of said entity pair,*

115. **First**, as discussed above at [1.0]-[1.3], Doshi teaches selecting an LSP pair (e.g., a primary and a protection LSP) ("*said entity pair*").

116. **Second**, Guichard teaches attempting to reoptimize LSP selection based on various trigger events that impact the network:

> **<u>Network state keeps changing</u>. Links and nodes fail and recover; new TE LSPs are set up while others may be torn down**. Consequently, for any TE LSP, a more optimal path may appear in the

> network. It is then highly desirable to detect the existence of such a path and reoptimize a TE LSP along a better path when it becomes available.

Guichard, 79; *see also* Guichard, 80.

> **<u>Event-driven reoptimization</u>—In some cases, it may be desirable to trigger a reoptimization upon the occurrence of a particular event, such as the restoration of a link in the network**. Indeed, if a link is restored in the network, some TE LSPs may benefit from that new link to follow a more optimal path. When configured, **each time a new link is advertised as newly operational, this triggers the evaluation of any potential reoptimization**.

Guichard, 81; *see also* Guichard, 79-80.

117. A POSITA would have understood that Guichard generally teaches that the current LSP path selection should be evaluated following a trigger event (e.g., change in the network such as LSP set up and tear down, restoration of a link, or new link) to determine whether a more optimal LSP path is available. Also, a POSITA would have recognized that this evaluation may or may not result in a reoptimization. *See* Guichard, 81 (referring to "evaluation of any **potential** reoptimization."); *see also* Vasseur '469, [0057]-[0059] (describing "determining whether a network event could create a more optimal path for one or more TE-LSPs" and comparing a "cost of the [new path] to the old cost of the current TE-LSP path" while recognizing that if the "new cost exceeds the old cost … no optimization of the TE-LSP is performed."). A person of skill would have understood that there is

only a potential for reoptimization since if the evaluation indicates that the currently utilized path is still the best option, then no reoptimization would occur and the previously utilized path is reselected.

118. It would have been obvious to a POSITA, in view of Guichard, to perform event-driven reoptimization on an ongoing basis, when implementing Doshi's MPLS architecture, to determine if there are better LSP paths available or if the currently selected LSP pair represents the overall minimal cost. Further support for the combination is provided in Section VIII.A.3 above.

119. In implementing the combination with Doshi, a POSITA would have evaluated any restored and new LSPs along with all existing LSPs (including the currently utilized LSP pair) by repeating Doshi's candidate pair cost determination and selection techniques (see analysis at [1.2]-[1.3]) in response to a trigger event to potentially reoptimize the selection.

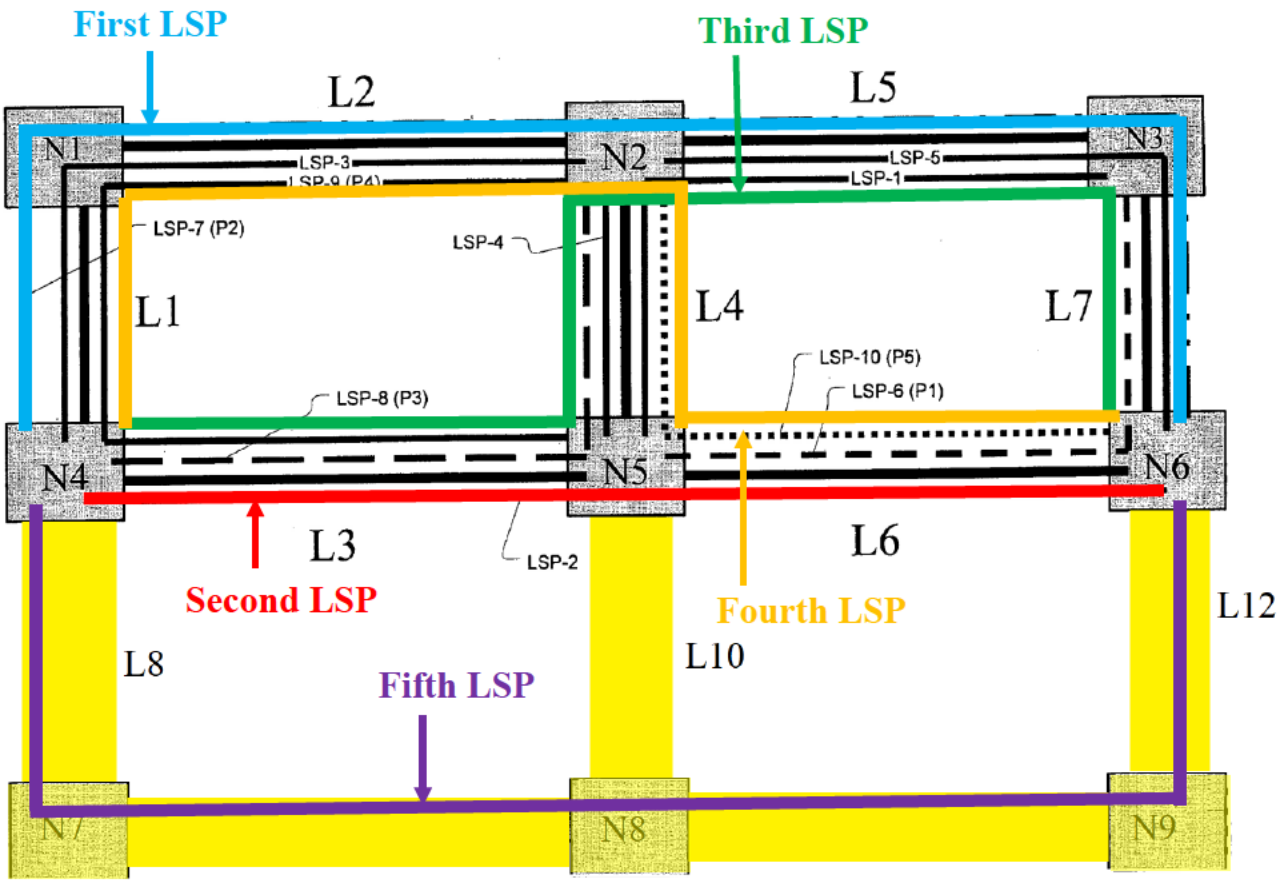
120. It would have been obvious to a POSITA that repeating Doshi's candidate path pair cost determination and selection would—in certain instances—result in the reselection of the currently utilized LSP pair. This would be the case where the currently utilized LSP pair still provides the overall minimum-cost, despite an LSP path being restored, a new LSP path being set up, or some other change in the network. In the combination, Doshi's currently utilized LSP pair would be repeatedly evaluated and reselected, per Guichard's event-driven reoptimization

technique, if it has the overall-minimum cost per Doshi's cost determination (see analysis at [1.2]).

121. Thus, Doshi in combination with Guichard discloses an event (e.g., change in the network such as LSP set up and tear down, restoration of a link, or new link) triggering candidate pair cost determination and, if the cost determination indicates that the currently used LSP pair still has the overall-minimum cost, reselecting the LSP pair, which renders obvious "*if an entity pair reselection event occurs, reselecting said entity pair*." *See also* analysis at element [1.5], below, analyzing further the reselection event.

122. **Third**, as discussed above at [1.0]-[1.3], Doshi's plurality of candidate path pairs include a plurality of LSPs (e.g., at least two primary and at least two protection LSPs) between a pair of end nodes. The currently used primary and protection LSP pair would be initially selected based on the overall minimum cost (see analysis above at [1.3]) and the remaining LSPs would be available (with any restored or new LSPs in the network that generate the trigger event, per Guichard) as potential replacements during reoptimization. Thus, in the proposed combination, the reselection is "*from the group consisting of said entity pair and a replacement entity pair comprising at least one entity distinct from the entities of said entity pair*."

123. Below, in the context of Doshi's modified Figure 1, I continue with my example, based on the combined teachings:



Doshi, Fig. 1 (modified and annotated).

124. As noted at [1.3], the selection step selected the Fifth LSP as the primary path and the Second LSP as the protection path. When performing event-based reoptimization, per Guichard, an overall cost for each LSP pair permutation would be determined, per Doshi. Doshi, [0014] ("…a path cost associated with said each candidate path pair is generated…"). If the cost determination indicates that the currently used LSP pair (e.g., Fifth LSP and Second LSP) still has the overall-minimum cost, then the same LSP pair would be reselected. Doshi, [0142] ("…the lowest-cost pair…is selected as the overall minimum-cost pair…").

125. Note that it would have been obvious to a POSITA that the reselection is from the group consisting of the currently used LSP pair (e.g., Fifth LSP and Second LSP) and a replacement LSP pair comprising at least one LSP distinct from the currently used LSP pair since, as discussed at [1.2] above, Doshi considers all possible pairs in the cost calculation. In my example, the replacement LSP pair is made from the different permutations of LSP pairs, including:

1) First LSP and Second LSP;
2) First LSP and Third LSP;
3) First LSP and Fourth LSP;
4) First LSP and Fifth LSP;
5) Fifth LSP and Third LSP;
6) Fifth LSP and Fourth LSP; and
7) Fifth LSP and Fifth LSP.

I note that there are additional permutations of possible LSP pairs with entirely different LSPs (i.e., both the primary and protection LSPs are distinct from the currently used LSP pair), but for brevity these permutations are not shown here. In the present example, the reselected LSP pair (e.g., Fifth LSP and Second LSP) would correspond to "*said entity pair*" while each remaining permutation of LSP pairs not resected would correspond to "*a replacement entity pair comprising at least one entity distinct from the entities of said entity pair*."

126. Fourth, to the extent Patent Owner contends that the claimed "*reselecting said entity pair*" refers to selecting a different LSP pair (e.g., replacing

one or both of the currently utilized primary and protection LSPs) instead of reselecting the same LSP pair, the proposed combination also renders that obvious. A POSITA would have found it obvious to select a different LSP pair, if the different LSP pair is determined to have a lower overall-minimum cost than the currently used LSP pair. To illustrate, in the current example, if the First LSP and Second LSP are now determined to have the lowest overall-minimum cost, then that LSP pair would be reselected. Of course, this is just an example, and given network status it would be understood that other LSPs may be reselected as the overall minimum cost pair.

127. Therefore, Doshi in combination with Guichard renders obvious "*if an entity pair reselection event occurs, reselecting said entity pair from the group consisting of said entity pair and a replacement entity pair comprising at least one entity distinct from the entities of said entity pair*," as claimed.

f. [1.5] *wherein said entity pair reselection event is selected from a group consisting of adding an entity to said plurality of transport entities, removing an entity from said plurality of transport entities, an operational status change for one of said plurality of transport entities, and a change in overall cost for one of said plurality of transport entities.*

128. First, as discussed at [1.4], Doshi in combination with Guichard discloses an event triggering candidate pair cost determination and pair selection, which corresponds to "*an entity pair reselection event*."

129. Second, Guichard teaches that a reoptimization evaluation is triggered by various events representative of changes in the network:

> **Event-driven reoptimization**—In some cases, it may be desirable to trigger a reoptimization upon the occurrence of a particular event, such as the restoration of a link in the network. Indeed, if a **link is restored in the network, some TE LSPs may benefit from that new link to follow a more optimal path**. When configured, each time a new link is advertised as **newly operational, this triggers the evaluation of any potential reoptimization**.

Guichard, 81. A POSITA would have understood that restoring a link restores LSPs of the link, which may provide a more optimal path.

130. As discussed above at [1.0]-[1.4], the plurality of MPLS LSPs between end nodes correspond to "*said plurality of transport entities*." It follows that restoring a link that previously went down to "newly operational," means that the operational status of the link's LSPs has also changed to operational, which discloses "*an operational status change for one of said plurality of transport entities*."

131. An additional network change contemplated by Guichard is the setting up and tearing down LSPs:

> **Network state keeps changing. Links and nodes fail and recover; new TE LSPs are set up while others may be torn down.** Consequently, for any TE LSP, a more optimal path may appear in the network. It is then highly desirable to detect the existence of such a path and reoptimize a TE LSP along a better path when it becomes available.

Guichard, 79.

132. It follows that a new LSP set up between end nodes would be added to the plurality of MPLS LSPs between the end nodes, which corresponds to "*adding an entity to said plurality of transport entities*." Conversely, tearing down an LSP between the end nodes would remove the LSP from the plurality of MPLS LSPs between the end nodes, which corresponds to "*removing an entity from said plurality of transport entities*."

133. Therefore, Doshi in combination with Guichard discloses that the triggering event includes the setting up an LSP, tearing down an LSP, and restoring the operational status of an LSP of the plurality of LSPs between the end nodes, which renders obvious "*wherein said entity pair reselection event is selected from a group consisting of adding an entity to said plurality of transport entities, removing an entity from said plurality of transport entities, an operational status change for one of said plurality of transport entities, and a change in overall cost for one of said plurality of transport entities*," as claimed.

5. Claim 2

a. [2.0] *The method of claim 1,*

134. See analysis at [1.0]-[1.5].

b. [2.1] *wherein said step of selecting an entity pair further comprises: selecting a working entity from said*

entity pair; and selecting a protection entity from said entity pair.

135. As discussed above at [1.3], Doshi discloses selecting an LSP pair from the plurality of MPLS LSPs. The selection includes selecting a primary LSP and selecting a protection LSP, which corresponds to "*selecting a working entity*" and "*selecting a protection entity*":

> For each of a plurality of candidate path pairs for the new service, each candidate path pair comprising a candidate primary path and a candidate restoration path for the new service, a path cost associated with said each candidate path pair is generated, wherein the path cost for a candidate path pair is a function of sharability of links within the corresponding candidate restoration path. **The <u>primary and restoration paths</u> for the new service <u>are selected</u>** from the plurality of candidate path pairs based on the path cost of each candidate path pair. The method may be implemented by a network manager for the <u>mesh [MPLS] network</u>.

Doshi, [0014]; *see also* Doshi, Abstract, [0046], [0142]-[0145], Claim 1, Claim 11.

136. Doshi's "primary path" is also referred to as a "**working path**" (e.g., working LSP), and Doshi's "restoration path" is also referred to as a "protection path" (e.g., protection LSP).[2] Note that the '821 patent describes switching traffic

[2] Doshi, [0046] ("Unless otherwise clear from the context, the process of 'restoration' is synonymous in this specification with the process of 'protection.'"), [0076] ("…

over to a protection entity after failure of a working entity. '821 patent, 1:51-43; *see also* '821 patent, 2:35-36 ("an entity protection switch configured to switch between a working entity and a protection entity"). Doshi similarly describes "Switching Between Working and Protection LSPs" and explains that "after detecting a failure, end nodes of an LSP switch traffic from a primary (i.e., working) LSP to its corresponding protection LSP." Doshi, [0266]-[0267]. Given the similarity in functionality and description, a POSITA would have understood that Doshi's working LSP corresponds to the claimed "*working entity*" while Doshi's protection LSP corresponds to the claimed "*protection entity*." Since Doshi's working and protection LSPs are part of the selected LSP pair, a POSITA would have understood that the selection of the working and protection LSPs are "*from said entity pair*."

137. Therefore, Doshi renders obvious "*wherein said step of selecting an entity pair further comprises: selecting a working entity from said entity pair; and selecting a protection entity from said entity pair*," as claimed.

6. Claim 3

a. [3.0] *The method of claim 2,*

138. See analysis at [2.0]-[2.1].

---

a primary path (also known as "**working path**") … a **protection path** (also known as "restoration path")").

b. [3.1] *further comprising the step of selecting an active entity from the set consisting of said working entity and said protection entity.*

139. As discussed at [2.1], Doshi describes selecting a working LSP and a protection LSP, which respectively correspond to the claimed "*working entity*" and "*protection entity*." Doshi further describes "Switching Between Working and Protection LSPs" and explains that "after detecting a failure, end nodes of an LSP switch traffic from a primary (i.e., working) LSP to its corresponding protection LSP." Doshi, [0266]-[0267]. Doshi's working LSP is the path actively carrying traffic absent a failure while Doshi's protection LSP becomes active when traffic is switched over following a failure. These switching decisions are consistent with the disclosure of the '821 patent. The '821 patent explains that "in the absence of entity failures, the working entity will be active" ('821 patent, 5:54-55) and that traffic is switched over to the '821 patent's protection entity in the event of a failure of the working entity ('821 patent, 1:51-43), which is to say that the protection entity then becomes active. Thus, a POSITA would have understood that Doshi's switching decisions correspond to "*selecting an active entity from the set consisting of said working entity and said protection entity*" because such decisions select which of the working LSP and protection LSP is in use.

140. Therefore, Doshi renders obvious "*further comprising the step of selecting an active entity from the set consisting of said working entity and said protection entity*," as claimed.

7. Claim 4

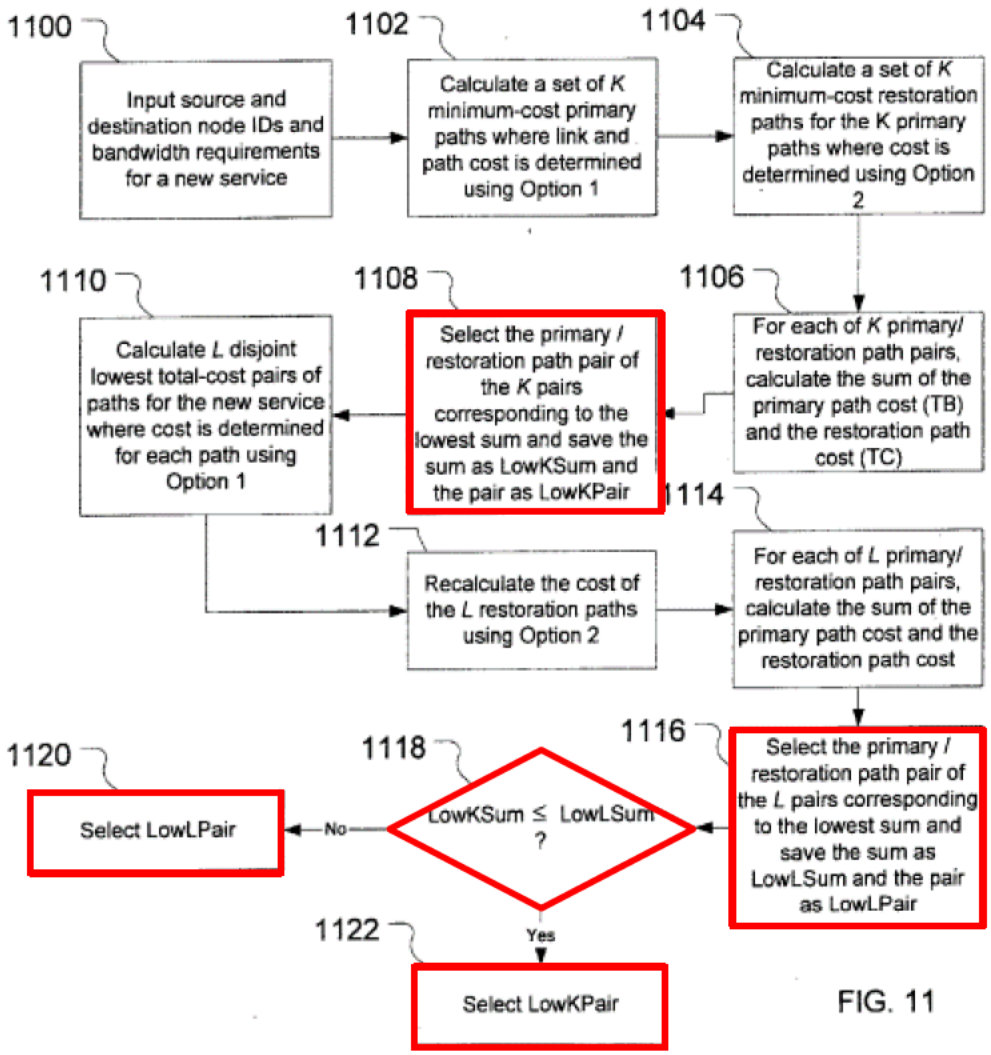
a. [4.0] *The method of claim 2,*

141. See analysis at [2.0]-[2.1].

b. [4.1] *wherein selecting an entity pair further comprises minimizing an overall cost function.*

142. **First**, as discussed at [1.2]-[1.3], Doshi calculates the path cost of each path (e.g., each primary and protection LSP) in a candidate path pair and sums the path costs to find the cost of the pair together. The cost calculation considers various factors, including link utilization, utilization threshold, administrative weight, sharing degree, and disjointedness, and which represent the "*overall cost"* for each candidate path pair. The calculation used to determine the "*overall cost*" corresponds to the claimed "*overall cost function*."

143. **Second**, as also discussed at [1.2]-[1.3], Doshi's Figure 11 and accompanying description teach calculating and selecting an "**overall minimum-cost"** pair:



Doshi, Fig. 11 (annotated).

> FIG. 11 depicts and exemplary **method for calculating a minimum-cost primary/restoration path pair** for a new service in consideration of link utilization, utilization threshold, administrative weight, and sharing degree.

Doshi, [0142].

> The method involves two major flows. In the first flow, illustrated by steps 1102, 1104, 1106, and 1106, one **minimum**-cost path pair is selected from K candidate pairs by a first method. In the second flow,

> illustrated by 1110, 1112, 1114, and 1116, one minimum-cost path pair is selected from L candidate pairs by a second method. **Then in step 1118, and steps 1120 or 1122, the lowest-cost pair of these two lowest-cost results is <u>selected as the overall minimum-cost pair</u>.**

Doshi, [0142]; *see also* Doshi, [0026], [0125]-[0139], [0143]-[0145]. Doshi's cost determination also takes into account whether paths are "disjoint," where disjoint paths would be understood as lowering the probability of simultaneous failure. Doshi, [0146]-[0156], Fig. 11 (step 1110 "Calculate L disjoint lowest total-cost pairs of paths").

144. A POSITA would have understood that the selected overall minimum-cost pair is the pair having the lowest overall cost function in consideration of various factors, including link utilization, utilization threshold, administrative weight, sharing degree, and disjointedness.

145. Therefore, Doshi renders obvious "*wherein selecting an entity pair further comprises minimizing an overall cost function*," as claimed.

8. Claim 5

a. [5.0] *The method of claim 4,*

146. See analysis at [4.0]-[4.1].

b. [5.1] *wherein said overall cost function comprises substantially minimizing a probability of concurrent failure of said protection entity and said working entity.*

147. First, as discussed at [1.2]-[1.3], and [4.1], Doshi calculates the path cost of each path (e.g., each primary and protection LSP) in a candidate path pair and sums the path costs to find the cost of the pair together. The cost calculation considers a myriad of factors such as link utilization, sharing degree, and **path disjointedness** and which represent the "*overall cost"* for each candidate path pair. The calculation used to determine the "*overall cost*" corresponds to the claimed "*overall cost function*."

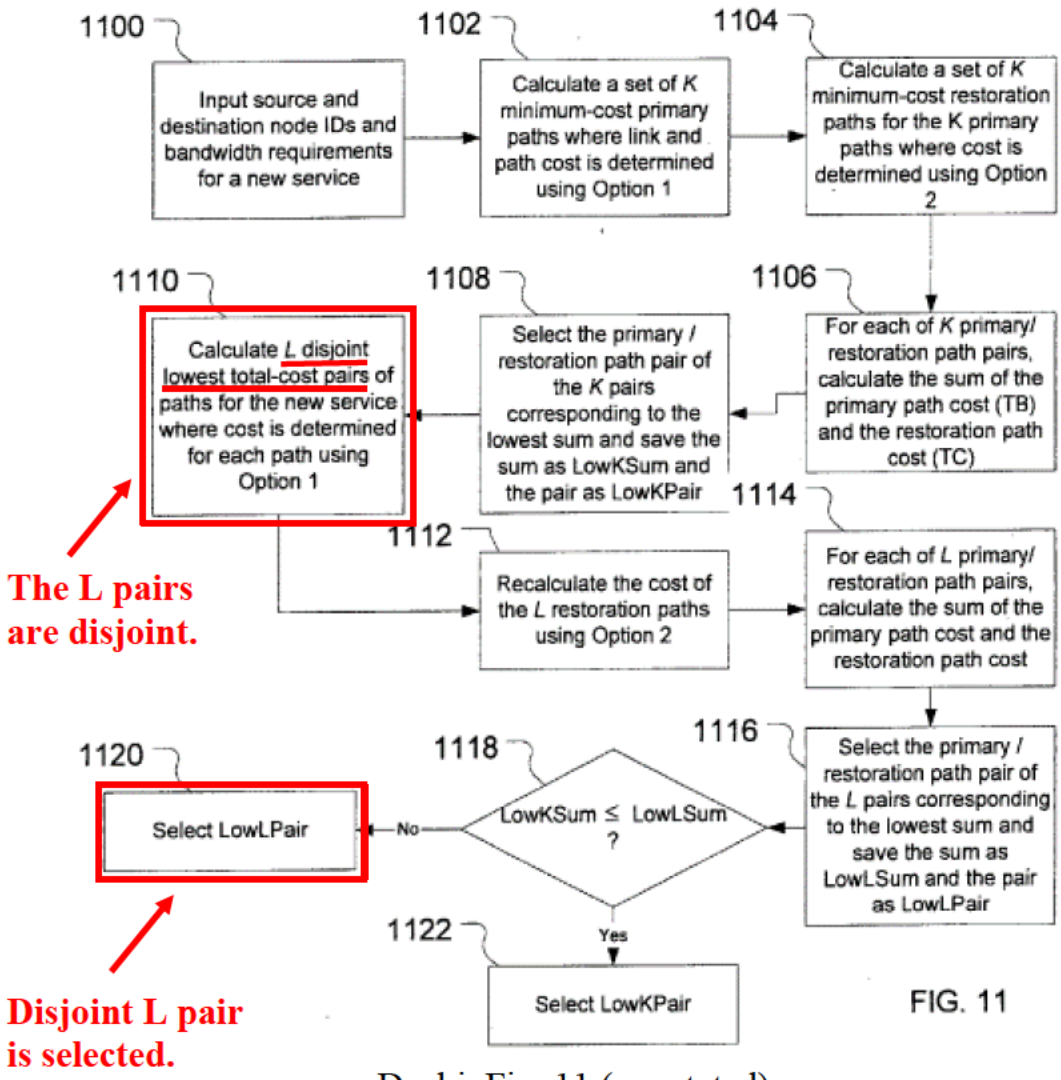
148. Second, regarding path disjointedness, Doshi describes selecting a primary path and a protection path (e.g., primary and protection LSPs) that are strictly disjoint from each other, having only source and destination nodes (e.g., end nodes) in common:

> **Disjoint Primary and Protection Paths**
>
> …
>
> Note that the protection paths in the example of FIG. 1 are strictly disjoint from the primary paths that they protect. In this context, **strictly disjoint means that the two paths in each primary/protection path pair share no common links or nodes other than their common ingress and egress nodes**.

Doshi, [0053], [0055].

> Primary/Restoration Paths: **Disjoint for Protection**
>
> Some of the services can be protected against a single transit node or link failure. This protection is made possible by **selecting a protection or restoration path for the service that <u>is node and link disjoint</u> from the primary path <u>except for the source and destination nodes</u>**. Whenever one of the primary path elements fails, the connection is switched automatically to the restoration path.

Doshi, [0155]-[0156]; *see also* Doshi, [0056], [0062], [0080], [0131], [0142]-[0152], [0287], Fig. 1.



Doshi, Fig. 11 (annotated).

149. Doshi explains that when two paths are strictly disjoint, e.g., by being node and link disjoint as above, "then a failure affecting one of them will not affect the other." Doshi, [0040]. By considering path disjointedness in its cost calculation and selecting a pair of primary and protection paths that are node and link disjoint such that the failure of one does not affect the other, Doshi's calculation "*substantially minimize[es] a probability of concurrent failure*" of the primary and

protection paths. I should also point out that Doshi's selection of disjoint paths is consistent with exemplary concurrent failure minimization described in the '821 patent. *See* '821 patent, 3:47-50 (recognizing that "a pair having multiple spans and/or nodes in common may be more likely to simultaneously fail than a pair having no spans or nodes in common.").

150. Although the discussion above refers to primary and protection or restoration paths, I again point out that as discussed in [2.0]-[2.1], Doshi clarifies that its "primary path" is also referred to as a "working path" (e.g., working LSP), and its "restoration path" is also referred to as a "protection path" (e.g., protection LSP).[3] These working and protection LSPs correspond to the claimed "*working entity*" and "*protection entity*" in the '821 patent. Accordingly, the discussion above applies equally to considering path disjointedness in cost calculation and selection of a pair of working and protection paths that are node and link disjoint such that the failure of one does not affect the other.

---

[3] Doshi, [0046] ("Unless otherwise clear from the context, the process of 'restoration' is synonymous in this specification with the process of 'protection.'"), [0076] ("… a primary path (also known as '**working path**') … a **protection path** (also known as 'restoration path')").

151. Therefore, Doshi renders obvious "*wherein said overall cost function comprises substantially minimizing a probability of concurrent failure of said protection entity and said working entity*," as claimed.

9. Claim 6

a. [6.0] *The method of claim 4,*

152. See analysis at [4.0]-[4.1].

b. [6.1] *wherein said overall cost function comprises a predefined entity cost metric.*

153. As discussed at [1.2]-[1.3] and [4.1], Doshi calculates the path cost of each path (e.g., each primary and protection LSP) in a candidate path pair and sums the path costs to find the cost of the pair together. The cost calculation considers various factors—including link utilization, utilization threshold, administrative weight, sharing degree, and disjointness—which represent the "*overall cost"* for each candidate path pair. The calculation used to determine the "*overall cost*" corresponds to the claimed "*overall cost function*."

154. Each of these factors, considered in the cost calculation, is predefined in the sense that the cost calculation is pre-configured to consider each factor. For example, before a calculation is performed, it is understood that the calculation will consider link utilization, sharing degree, and path disjointedness. Additionally, the utilization threshold (e.g., nominally set to 80%) and administrative weight (e.g., a coefficient for scaling) are predefined metrics. Doshi, [0128], [0141]. Each of these

metrics has an impact on the cost of the path being evaluated. Doshi, [0142]. Thus, each of link utilization, utilization threshold, administrative weight, sharing degree, and disjointedness correspond to "*a predefined entity cost metric*."

155. Additionally, I note that traffic engineering metrics is an exemplary "*predefined entity cost metric*" described in the '821 patent. '821 patent, 5:49-51. A POSITA would have understood that Doshi's link utilization, utilization threshold, and administrative weight each correspond to traffic engineering data since it was well known that traffic engineering metrics includes such factors. *See* RFC 3272, 56 (Entitled "Overview and Principles of Internet Traffic Engineering" describing use of "administrative weights" to implement traffic engineering decisions.); Release 12.0(5)S, 28, 30 (Entitled "Multiprotocol Label Switching (MPLS) Traffic Engineering" describing setting an administrative weight with an "mpls traffic-eng administrative-weight" command, setting a bandwidth threshold level of reserved bandwidth with an "mpls traffic-eng flooding thresholds" command, and identifying traffic engineering output fields that include administrative weight, reservable bandwidth, and bandwidth thresholds.); Fedyk, 4, 6 (Entitled "Multiple Metrics for Traffic Engineering with IS-IS and OSPF" describing several metrics, including "administrative weight" and "bandwidth available.").

156. A POSITA's understanding that Doshi's link utilization, utilization threshold, and administrative weight each correspond to traffic engineering metrics

would have been further confirmed by the fact that Doshi's disclosure is in the context of MPLS traffic engineering:

> [T]his invention has been described in one or more embodiments in the context of mesh data networks running one or more of the **multiprotocol-label switched (MPLS) traffic engineering extension protocols** …

Doshi, [0298]; see also Doshi, [0010], [0093], [0191], and [0257].

157. Therefore, Doshi renders obvious "*wherein said overall cost function comprises a predefined entity cost metric*," as claimed.

10. Claim 7

a. [7.0] *The method of claim 6,*

158. See analysis at [6.0]-[6.1].

b. [7.1] *wherein said predefined entity cost metric is selected from the group consisting of interior gateway protocol (IGP) and traffic engineering (TE).*

159. See analysis at [6.1].

11. Claim 9

a. [9.0] *The method of claim 4,*

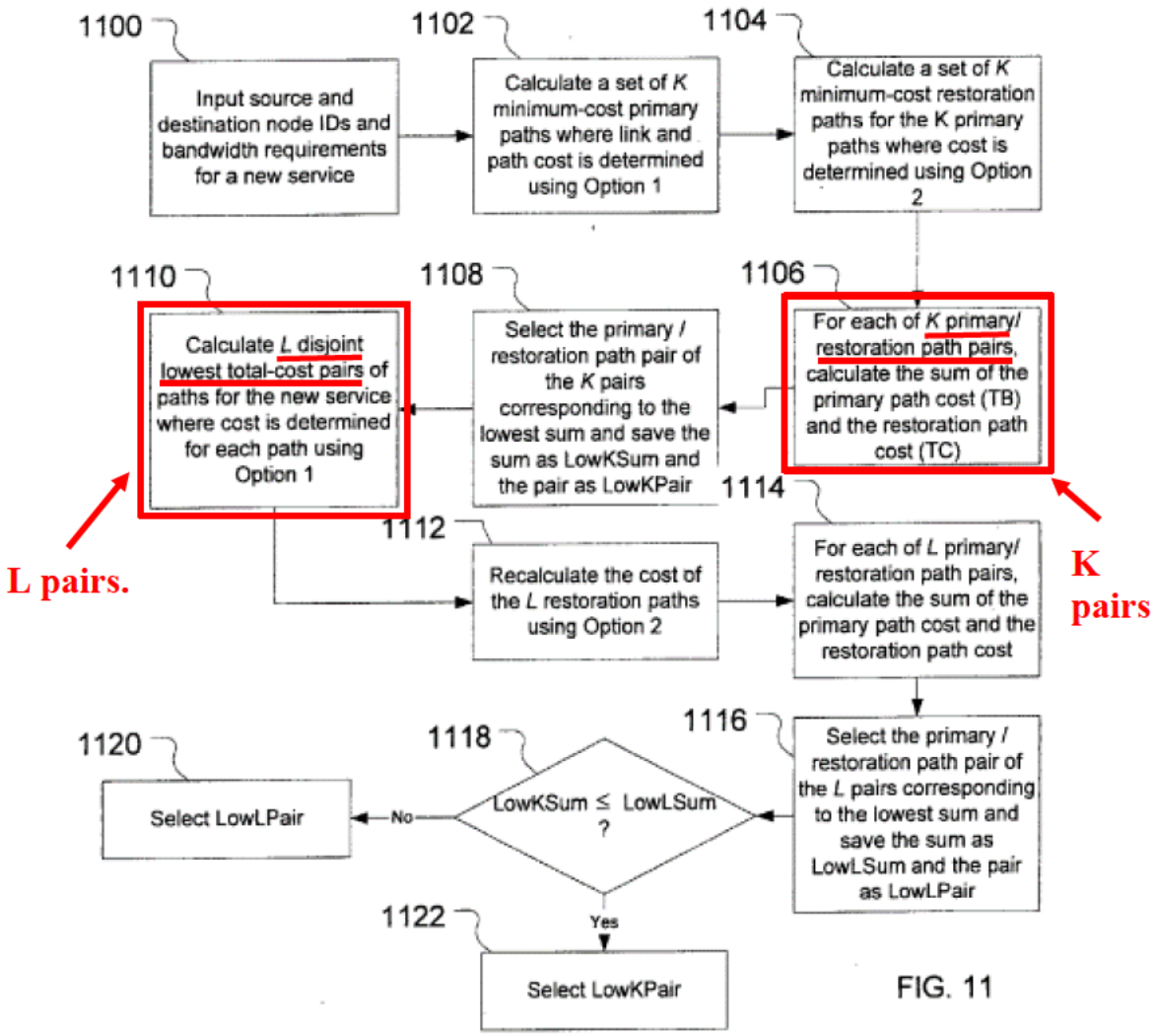
160. See analysis at 4.0]-[4.1].

b. [9.1] *wherein said overall cost function comprises: selecting a subset of entity pairs wherein each entity pair of said subset has substantially minimum*

probability of a concurrent failure of said protection entity and said working entity; and

161. As discussed at [1.2]-[1.3], Doshi calculates the path cost of each path (e.g., each primary and protection LSP) in a candidate path pair and sums the path costs to find the cost of the pair together. The cost calculation considers various factors, including link utilization, utilization threshold, administrative weight, sharing degree, and **disjointedness**, and which represent the "*overall cost"* for each candidate path pair. The calculation used to determine the "*overall cost*" corresponds to the claimed "*overall cost function*."

162. As shown at Doshi's Figure 11 and disclosed in the accompanying description, the cost calculation includes calculating and comparing two sets of candidate path pairs—K candidate pairs and L candidate pairs, where the L pairs are disjoint pairs:



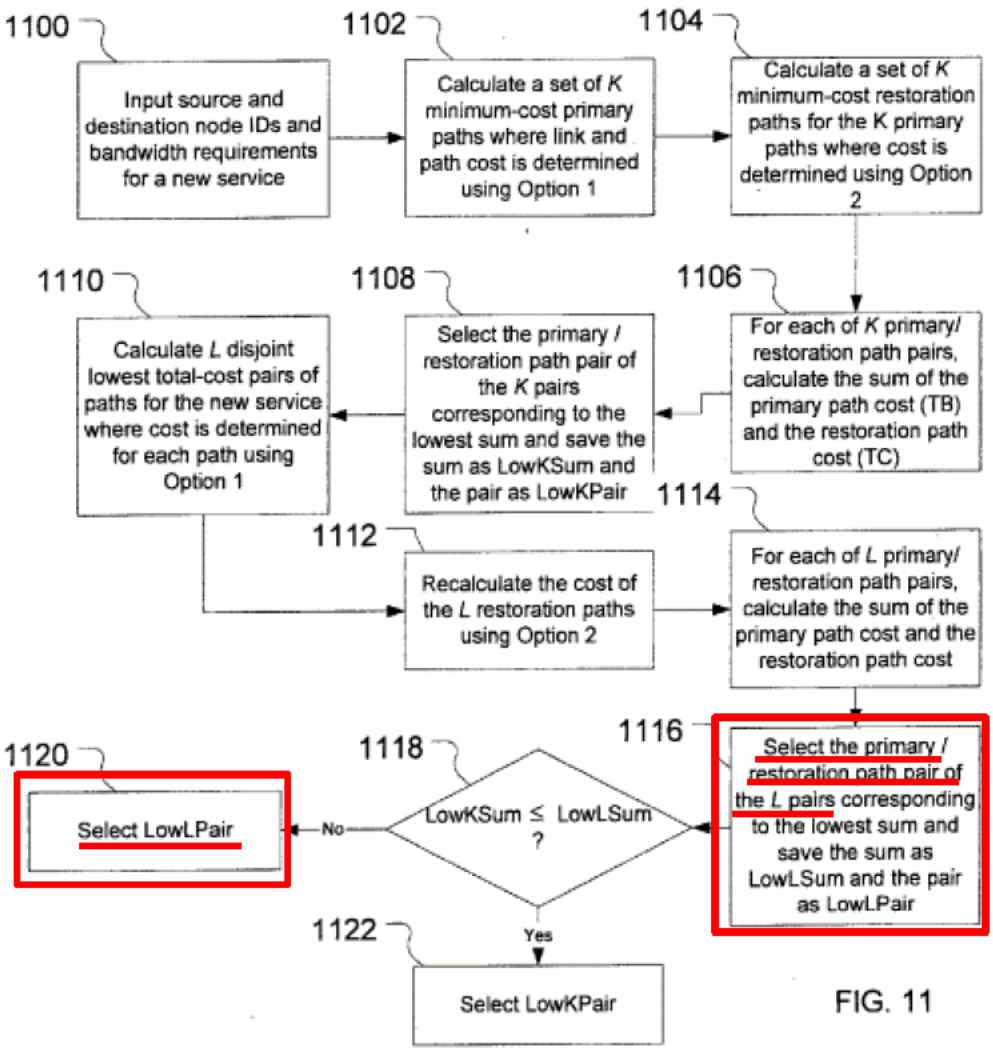
Doshi, Fig. 11 (annotated).

In the first flow, illustrated by steps 1102, 1104, 1106, and 1106, one minimum-cost path pair is selected from **K candidate pairs** by a first method. In the second flow, illustrated by 1110, 1112, 1114, and 1116, one minimum-cost path pair is selected from **L candidate pairs** by a second method.

Doshi, [0142]; *see also* Doshi, [0143]-[0145].

163. The L candidate pairs and K candidate pairs are both subsets of the total number of candidate pairs available. In the case of the L candidate pairs, the subset consists of pairs that are disjoint. As I discussed at [5.1], the pairs include both a working path (e.g., working LSP) and a protection path (e.g., protection LSP) that are disjoint from each other, which substantially minimizes the probability of concurrent failure. Thus, Doshi's L candidate pairs correspond to "*a subset of entity pairs wherein each entity pair of said subset has a substantially minimum probability of a concurrent failure of said protection entity and said working entity*." A POSITA would have understood selecting the set of L candidate pairs for calculations corresponds to "*selecting*" as claimed.

164. Moreover, since the claim does not specify any particular number of "*entity pairs*" within the claimed "*subset*," I note that the subset would be understood to consist of a single pair or multiple pairs. Doshi also discloses selection of a single L disjoint candidate pair as illustrated in annotated Figure 11 below:



Doshi, Fig. 11 (annotated).

165. Therefore, Doshi renders obvious "*wherein said overall cost function comprises: selecting a subset of entity pairs wherein each entity pair of said subset has substantially minimum probability of a concurrent failure of said protection entity and said working entity.*"

c. [9.2] *if said subset comprises at least two entity pairs, selecting an entity pair from said subset that minimizes an entity cost function.*

166. Regarding a subset including at least two pairs, Doshi discloses calculating L "disjoint lowest total-cost pairs," plural, as discussed at [9.1] above. Thus, Doshi's L candidate pairs include at least two pairs. Doshi goes on to explain that the L candidate pairs are assessed for cost, which as discussed at [1.2] includes assessing the cost of each path in the pair consistent with the '821 patent's description of "*entity cost*." As discussed at [1.2]-[1.3] and [4.1], Doshi calculates the path cost of each path (e.g., each primary and protection LSP) in a candidate path pair. The cost calculation considers various factors, including link utilization, utilization threshold, administrative weight, sharing degree, and disjointedness. Doshi, [0124]-[0141]. The calculation used to determine the path cost corresponds to the claimed "*entity cost function*":

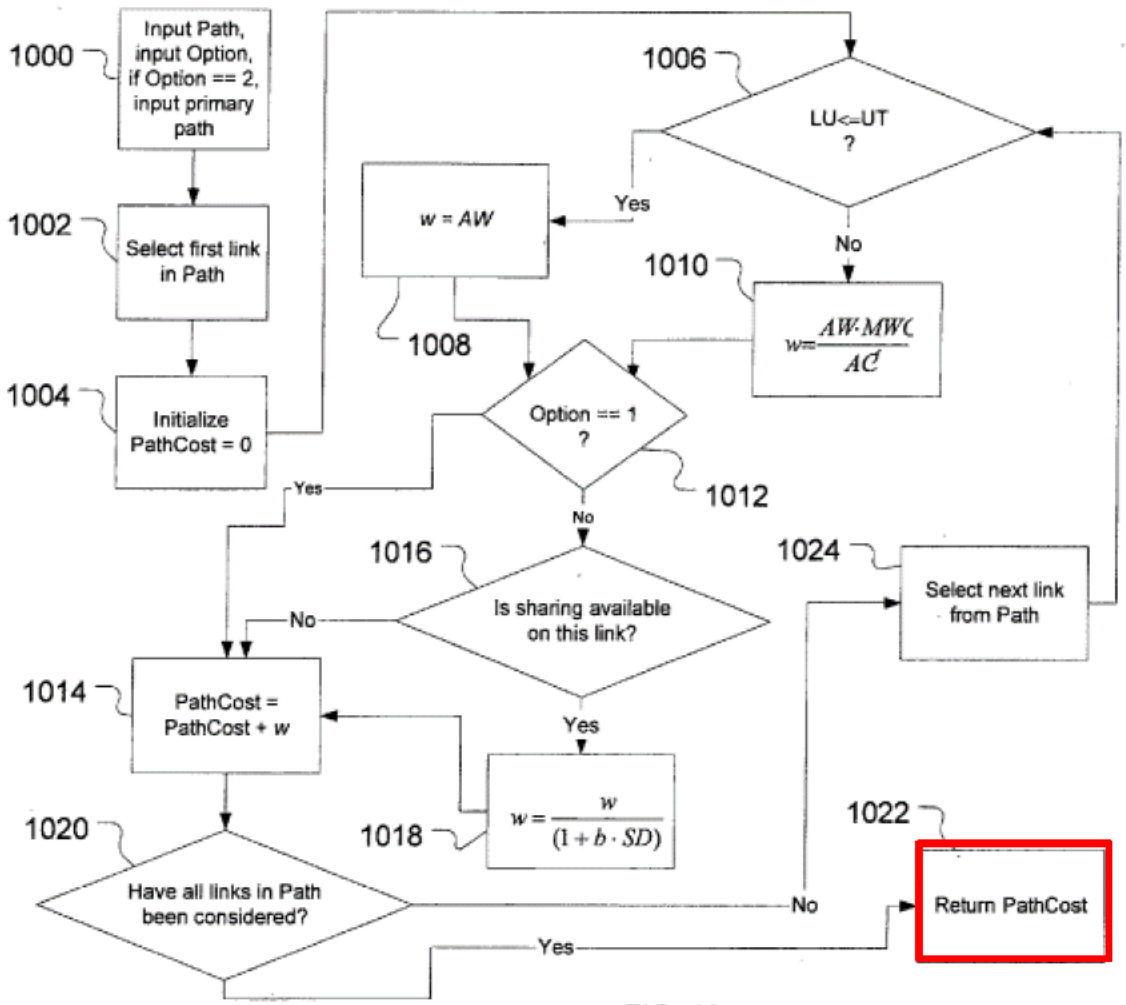
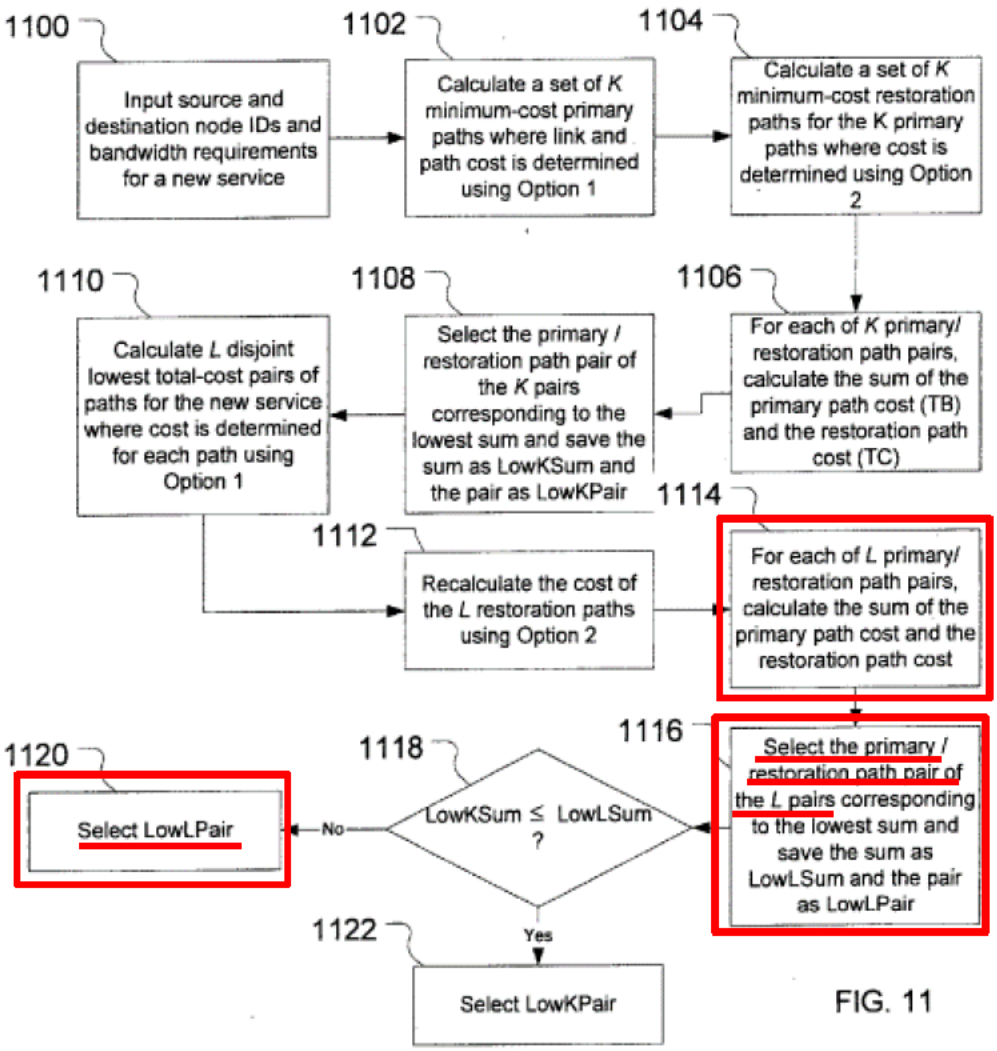


FIG. 10

Doshi, Fig. 10 (annotated).

167. Doshi describes selecting the lowest cost pair of the L candidate pairs after assessing for cost:



Doshi, Fig. 11 (annotated).

Next in step 1114, the sum is calculated of the cost of the primary and the restoration path for each of the L pairs and in step 1116, the lowest sum is stored in the variable LowLSum and the **lowest-cost pair is stored in the data structure LowLPair**.

Finally, in the test of step 1118, LowKSum is compared to LowLSum. If LowKSum is lower than or equal to LowLSum, then in step 1122, LowKPair is selected as the minimum-cost path pair. If it is not lower, then in step 1118, **LowLPair is selected**.

Doshi, [0144]-[0145].

168. Not only is the lowest cost pair of the L candidate pairs selected, its selection marks it as the lowest cost pair out of all pairs, including K candidate pairs. It would have been obvious to a POSITA for the lowest cost pair to include the lowest cost path. Accordingly, it would have been obvious to a POSITA that selection of the lowest cost L pair—the lowest cost pair out of all the pairs—"*minimizes an entity cost function*."

169. Therefore, Doshi renders obvious "*if said subset comprises at least two entity pairs, selecting an entity pair from said subset that minimizes an entity cost function*," as claimed.

12. Claim 10

a. [10.0] *The method of claim 9,*

170. See analysis at [9.0]-[9.2].

b. [10.1] *wherein said entity cost function comprises a predefined metric.*

171. As discussed at [1.2]-[1.3] and [4.1], Doshi calculates the path cost of each path (e.g., each primary and protection LSP) in a candidate path pair. As discussed at [9.2], the cost calculation considers various factors, including link utilization, utilization threshold, administrative weight, sharing degree, and disjointedness and corresponds to the claimed "*entity cost function*." Further, as

discussed at [6.1], these factors, along with traffic engineering, correspond to a "*predefined entity cost metric*," which renders obvious "*a predefined metric.*"

13. Claim 11

a. [11.0] *The method of claim 10,*

172. See analysis at [10.0]-[10.1].

b. [11.1] *wherein said predefined metric is selected from the group consisting of interior gateway protocol (IGP) and traffic engineering (TE).*

173. See analysis at [10.1].

14. Claim 13

a. [13.0] *The method of claim 1,*

174. See analysis at [1.0]-[1.5].

b. [13.1] *further comprising the step of: if said entity pair reselection results in both working and protection entities being replaced, sequentially replacing said working entity and said protection entity.*

175. Regarding replacement of the claimed "*working entity*" and "*protection entity*," I first note that as discussed at [2.1] and [5.1], Doshi's "primary path" is also referred to as a "working path" (e.g., working LSP), and its "restoration path" is also

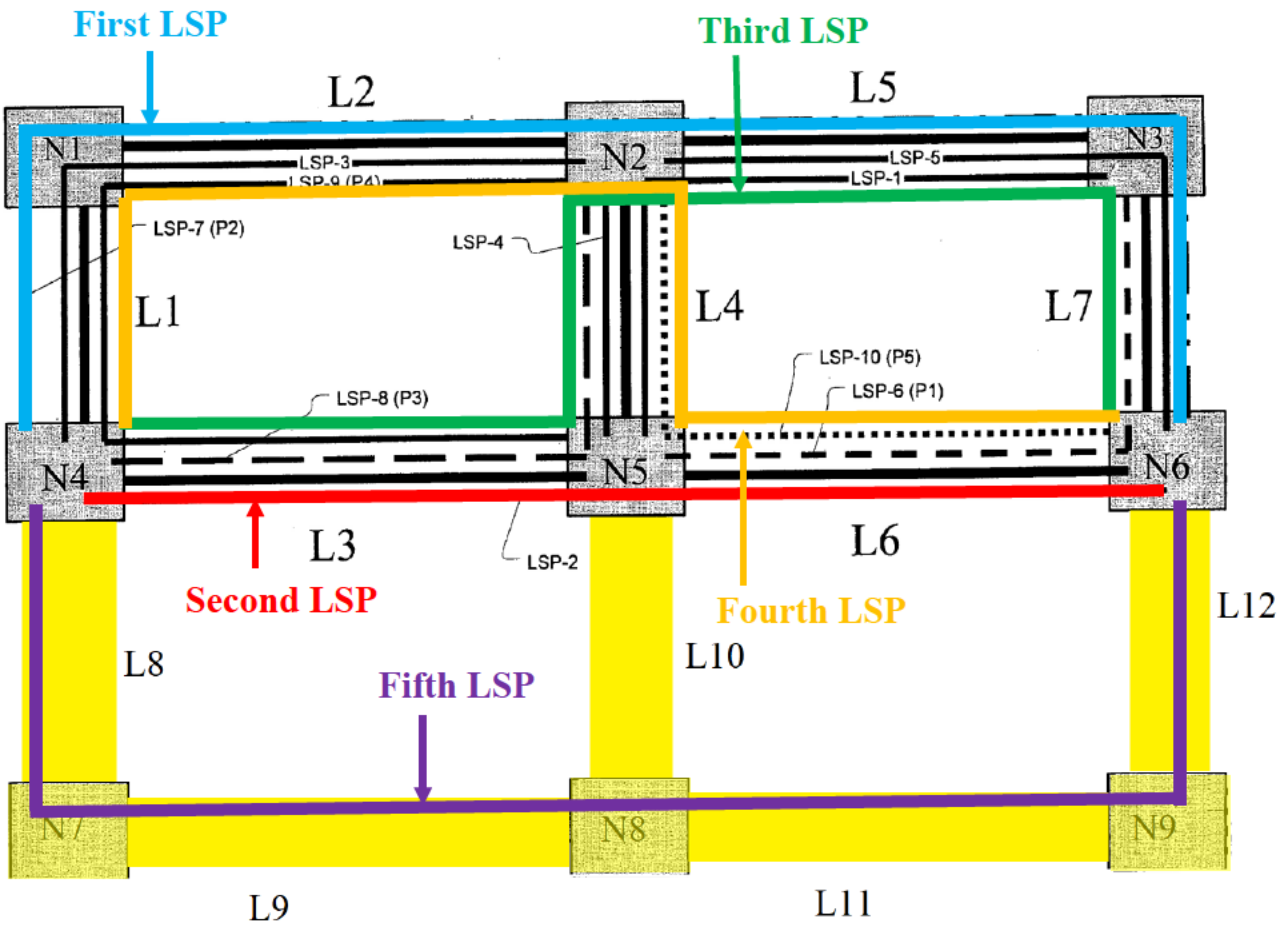
referred to as a "protection path" (e.g., protection LSP). [4] Doshi's primary path (aka working path) corresponds to the claimed "*working entity*" and Doshi's restoration path (aka protection path) corresponds to the claimed and "*protection entity.*"

176. As discussed at [1.3]-[1.4], Doshi teaches that a primary and a protection pair (e.g., LSP pair) are selected from the plurality of paths of the MPLS network and that, in view of Guichard's teaching of ongoing reoptimization efforts, it would have been an obvious to perform Doshi's cost calculations repeatedly to attempt to determine whether the originally selected LSP pair still has the overall minimum-cost or if another LSP pair should be selected.

177. It would have been obvious for the LSPs of the originally selected pair (e.g., a primary LSP and a protection LSP) to be sequentially replaced as the ongoing reoptimization efforts identify lower cost LSPs, which corresponds to "*if said entity pair reselection results in both working and protection entities being replaced, sequentially replacing said working entity and said protection entity*." To build upon an earlier example, consider Doshi's modified and annotated Figure 1 below:

---

[4] Doshi, [0046] ("Unless otherwise clear from the context, the process of 'restoration' is synonymous in this specification with the process of 'protection.'"), [0076] ("… a primary path (also known as '**working path**') … a **protection path** (also known as 'restoration path')").



Doshi, Fig. 1 (modified and annotated).

178. As previously discussed at [1.3]-[1.4], the Fifth LSP was selected as the primary path and the Second LSP was selected as the protection path. Upon an event trigger as taught by Guichard, attempted reoptimization of the LSPs would take place to determine if there is a better or more optimal LSP. If, for example, the reoptimization evaluation determined that the links of the Fifth LSP are heavily utilized as compared to the links of the First LSP, it would have been obvious for Doshi's cost determination to result in a new pair (e.g., the First LSP as the primary while the Second LSP remains as the protection) as having the overall minimum

cost. Thus, the Fifth LSP would be replaced. A subsequent such reoptimization evaluation would find that, for example, the Second LSP has become suboptimal as a protection path and replace the Second LSP with another LSP (e.g., a newly added Sixth LSP (not shown)). Of course, this is just an example, and given network status it would be understood that other LSPs may be selected as the overall minimum cost pair, but I note that the LSPs have been replaced sequentially in this example—first the primary LSP followed by the protection LSP. It would have been obvious for the sequence of replacement to be reversed in other examples where the protection LSP becomes suboptimal before the primary LSP.

179. It would have been similarly obvious for a single event-triggered reoptimization evaluation to find that both the Fifth and Second LSPs are suboptimal. When both LSPs are determined to be suboptimal, a POSITA would have found it obvious to replace the LSPs sequentially in order to minimize interruption in service. In that regard, a POSITA would have appreciated the desirability of maintaining a working and protection path throughout the replacement process. In replacing the Fifth and Second LSPs with the First and Sixth LSPs, for example, a POSITA would have found it obvious to first designate the First LSP (determined to be the optimal primary LSP) as a new protection LSP temporarily, thereby replacing the Second LSP. Next, traffic on the Fifth LSP would be switched over to the First LSP with the Fifth LSP temporarily serving as a

protection LSP. The First LSP would thus become the new primary LSP as was determined to be optimal. The Fifth LSP, now acting as a protection LSP, would then be replaced by the Sixth LSP, which was determined to be the optimal protection LSP. A POSITA would have recognized that such sequential replacement would advantageously maintain two LSPs (e.g., one primary and one protection) throughout the replacement.

180. Therefore, Doshi in combination with Guichard renders obvious "*if said entity pair reselection results in both working and protection entities being replaced, sequentially replacing said working entity and said protection entity*," as claimed.

181.

15. Claim 17

a. [17.0] *Non-transitory computer readable media configured to perform a method comprising the steps of:*

182. Claim 17 is substantially similar to claim 1 and is therefore obvious for the reasons stated above. The below analysis addresses the obviousness of the preamble's different recitations, to the extent limiting.

183. As discussed at [1.0], Doshi discloses that the claimed method steps are performed by a network manager, that may be a server or other computing device. It was common knowledge that computing devices, such as servers, would include

a processor and memory. *See* Xu, [0092], Fig. 10 (describing "exemplary computer system 1000 that can be used as … a server" where the computer system 1000 includes "one or more processors," "a main memory," "and may also include a secondary memory"); *see also* Kao, 8:30-31 (describing that "server 502 includes a processor 504 operably coupled to memory 506"); Shah, 7:46-47 ("Each server may include a processor and memory."); Fuller, 17:60-61 ("The server computer system 103 may comprise a CPU and a memory medium."). Indeed, Doshi expressly discloses storing, for example, protection information and information about path pairs which would have informed a POSITA that Doshi's network manager includes a memory. Doshi, [0143]-[0144], [0196], [0214].

184. It was similarly common knowledge for such memory to be non-volatile and computer readable by the processor ("*[n]on-transitory computer readable media*") while storing instructions for execution by the processor. *See* Xu, [0092] (describing that "[c]omputer programs (also referred to as computer control logic) are stored in main memory 1008 and/or secondary memory 1010" and that "[c]omputer programs, when executed, enable the computer system 1000 to perform the features of the present disclosure."); *see also* Werner, [0072] (describing "computer-readable" hard disk drives and flash memory cards "provide nonvolatile storage of data, data structures, computer-executable instructions, and so forth."); Jungck, 17:29-31, 41:31-32 (describing "software and/or firmware provided in a

read only memory … for execution" by a processor, where the memory includes "non-volatile memory 812" comprising "Flash memory or Static RAM"); Rothman, [0021], [0003] (describing a processor "executing sequences of instructions contained in a memory" and explaining that "[n]on-volatile memory retains the stored data after the power is removed for a period of time.").

185. Accordingly, a POSITA would have found it obvious for Doshi's network manager to include a memory that corresponds to a "*non-transitory computer readable medium*." As discussed above at [1.0]-[1.5], Doshi's network manager performs the method steps. Since, as evidenced by the citations above, it was common knowledge to store executable instructions in memory, and since Doshi's network manager performs the method steps, a POSITA would have considered it obvious to store instructions to perform each step of the method in the memory of Doshi's network manager. Accordingly, a POSITA would have found it obvious for Doshi's network manager to include "*[n]on-transitory computer readable media configured to perform a method comprising the steps of:*"

186. Therefore, Doshi renders obvious "*Non-transitory computer readable media configured to perform a method comprising the steps of:*"

b. [17.1] *providing a plurality of MPLS transport entities between a first endpoint and a second endpoint;*

187. See analysis at [1.1].

c. [17.2] ***determining an overall cost for each entity pair of said plurality of entities;***

188. See analysis at [1.2].

d. [17.3] ***selecting an entity pair from said plurality of transport entities based at least in part upon said overall cost; and***

189. See analysis at [1.3].

e. [17.4] ***if an entity pair reselection event occurs, reselecting said entity pair from the group consisting of said entity pair and a replacement entity pair comprising at least one entity distinct from the entities of said entity pair,***

190. See analysis at [1.4].

f. [17.5] ***wherein said entity pair reselection event is selected from a group consisting of adding an entity to said plurality of transport entities, removing an entity from said plurality of transport entities, an operational status change for one of said plurality of transport entities, and a change in overall cost for one of said plurality of transport entities.***

191. See analysis at [1.5].

16. Claim 18

a. [18.0] ***The non-transitory computer readable media of claim 17,***

192. See analysis at [17.0]-[17.5].

b. [18.1] ***wherein said step of selecting an entity pair further comprises: selecting a working entity from said***

plurality of transport entities; and selecting a protection entity from said plurality of transport entities; and

193. As discussed above at [1.3], Doshi discloses selecting an LSP pair from the plurality of MPLS LSPs. The selection includes selecting a primary LSP and selecting a protection LSP, which corresponds to "*selecting a working entity from said plurality of transport entities; and selecting a protection entity from said plurality of transport entities*":

> For each of a plurality of candidate path pairs for the new service, each candidate path pair comprising a candidate primary path and a candidate restoration path for the new service, a path cost associated with said each candidate path pair is generated, wherein the path cost for a candidate path pair is a function of sharability of links within the corresponding candidate restoration path. **The <u>primary and restoration paths</u> for the new service <u>are selected</u>** from the plurality of candidate path pairs based on the path cost of each candidate path pair. The method may be implemented by a network manager for the <u>mesh [MPLS] network</u>.

Doshi, [0014]; *see also* Doshi, Abstract, [0046], [0142]-[0145], Claim 1, Claim 11.

Doshi's "primary path" is also referred to as a "**working path**" (e.g., working LSP), and Doshi's "restoration path" is also referred to as a "protection path" (e.g.,

protection LSP).[5] Note that the '821 patent describes switching traffic over to a protection entity after failure of a working entity. '821 patent, 1:51-43; *see also* '821 patent, 2:35-36 ("an entity protection switch configured to switch between a working entity and a protection entity"). Doshi similarly describes "Switching Between Working and Protection LSPs" and explains that "after detecting a failure, end nodes of an LSP switch traffic from a primary (i.e., working) LSP to its corresponding protection LSP." Doshi, [0266]-[0267]. Given the similarity in functionality and description, a POSITA would have understood that Doshi's working LSP corresponds to the claimed "*working entity*" while Doshi's protection LSP corresponds to the claimed "*protection entity*." Since Doshi's working and protection LSPs are part of the selected LSP pair, a POSITA would have understood that the working and protection LSPs are "*from said entity pair*."

194. Therefore, Doshi renders obvious "*wherein said step of selecting an entity pair further comprises: selecting a working entity from said plurality of*

[5] Doshi, [0046] ("Unless otherwise clear from the context, the process of 'restoration' is synonymous in this specification with the process of 'protection.'"), [0076] ("… a primary path (also known as '**working path**') … a **protection path** (also known as 'restoration path')").

transport entities; and selecting a protection entity from said plurality of transport entities," as claimed.

c. [18.2] *selecting an active entity from the set consisting of said working entity and said protection entity.*

195. See analysis at [3.1].

17. Claim 19

a. [19.0] *The non-transitory readable media of claim 18,*

196. See analysis at [17.0]-[17.5], and [18.0]-[18.2].

b. [19.1] *wherein said step of selecting an entity pair further comprises minimizing an overall cost function.*

197. See analysis at [4.1].

18. Claim 20

a. [20.0] *The non-transitory readable media of claim 19,*

198. See analysis at [17.0]-[17.5], [18.0]-[18.2], and [19.0]-[19.1].

b. [20.1] *wherein said overall cost function comprises: minimizing a probability of concurrent failure of said protection entity and said working entity; and*

199. See analysis at [5.1].

c. [20.2] *a predefined metric selected from the group consisting of interior gateway protocol (IGP) and traffic engineering (TE).*

200. See analysis at [6.1] and [7.1].

B. Ground 2: The Combination of Doshi, Guichard, and Huang renders obvious claims 8 and 12.

1. Summary of Huang

201. U.S. Patent Pub. No. 2003/0117950 to Huang (Ex.1007, "Huang") was published June 26, 2003. Huang is titled "Link redial for mesh protection." Huang at Title. Huang was not before the Examiner during prosecution of the '821 patent.

202. Like the '821 patent and Doshi, Huang describes LSP selection within an MPLS network. Huang, [0004], [0032], [0035], [0040]; *see* '821 patent, Abstract, 3:23-26; *see also* Doshi, [0050]-[0054], [0142]-[0145]. Huang's LSP selection is described in the context of "protection of connections formed through a mesh-type communication network." Huang, [0001]. In Huang, a request is received to set up a label switched path segment between a head end node and a tail end node, where the request further identifies a backup route to the tail end node. Huang, [0016]. The backup routes are "backup LSP[s]" that serve to "protect each of the working links" such that in the event of a failure traffic "may be switched to corresponding individual backup LSPs." Huang, [0032]. Huang's "working links" are also LSPs. *See* Huang, [0030]-[0031] (explaining that each "working link" is associated with an LSP and that a link is a "working link" when traffic is flowing over the LSP). After traffic has been switched over to a backup LSP following a failure, Huang describes "revertive switching" such that "the connection will be switched back from

the backup LSP to the working link once the working link has cleared the failure that caused the switchover." Huang, [0053].

2. Reasons to Combine Doshi and Huang

203. A POSITA would have been motivated to combine the teachings of Doshi and Huang, as discussed below.

204. **First**, Doshi and Huang are analogous art to the '821 patent. For example, just like the '821 patent, both references generally pertain to MPLS networks. '821 patent, 3:15-18, Abstract; Doshi, [0049]; Huang, [0004]-[0007]. Additionally, both references address the problem of selecting a pair of MPLS paths from available paths. '821 patent, 4:63-5:8; Doshi, [0014]; Huang, [0004], [0032], [0035], [0040], [0048]. Thus, given the similarities between the references and the fact that they address the problem of selecting LSPs, a POSITA considering Doshi would have naturally considered the teachings of Huang.

205. **Second**, a POSITA would have been motivated to combine the teachings of Doshi and Huang to produce numerous predictable and beneficial results.

206. Doshi teaches "Switching Between Working and Protection LSPs" and explains that "after detecting a failure, end nodes of an LSP switch traffic from a primary (i.e., working) LSP to its corresponding protection LSP." Doshi, [0266]-[0267]. Huang complements Doshi by teaching what to do when the failure affecting

the working LSP has been cleared. Huang teaches "revertive switching" where "the connection will be switched back from the backup LSP to the working link once the working link has cleared the failure that caused the switchover." Huang, [0053].

207. A POSITA would have been motivated to perform revertive switching, as taught by Huang, when implementing Doshi's switching between working and protection LSPs in order to avoid the need to calculate and select a replacement working LSP when the previously selected working LSP (which as discussed at [1.2]-[1.4] is part of the overall minimum-cost pair) has been quickly restored. In that regard, Doshi recognizes the known problem that LSPs are sometimes incorrectly declared to have failed as a result of packet loss or delay. Doshi, [0261]. A working LSP that was incorrectly declared to have failed would quickly be identified as being "restored" when, for example, the state of the LSP can be checked using a fast failure detection function. *See* Doshi, [0256]-[0263].

208. A POSITA would have recognized that in instances of quick restoration, e.g., in the case of a mistaken failure diagnosis that is quickly resolved, it would be simpler to revert back to the working LSP instead of calculating the costs for all LSPs (e.g., by the network manager) to ultimately reach the same result of selecting the working LSP. Reverting allows for the previously selected working LSP (which as discussed at [1.2]-[1.4] is part of the overall minimum-cost pair) to

be utilized without any additional computations or selections. Thus, Huang's teachings obviate unnecessary efforts.

209. The combination is merely the use of a known technique (Huang's technique of configuring the working LSP as revertive) to improve a similar method (Doshi's method for selecting working and protection LSPs) in the same way with predictable results (reverting back to Doshi's working LSP once the failure has been cleared without performing needless calculations). The combination represents what would have been common sense to a POSITA—reverting back to a preferred low cost path when that path becomes operational again.

210. I also note that the results would have been predictable and there would have been a reasonable expectation of success in the combination given that Doshi and Huang are analogous art and in the same field of endeavor. Also, a POSITA would have had a reasonable expectation of success in reverting back to Doshi's working LSP as taught by Huang because Huang provides implementation details such as waiting to revert back until the working LSP has been restored. Huang, [0053].

211. Implementing the combination would have been within a POSITA's skillset since Doshi already sets forth techniques for switching between working and protection LSPs. Additionally, the combination would be a simple matter since Doshi already describes how to switch between LSPs. Doshi, [0266]-[0268]

(describing, for example, switching between LSPs by "activating one or the other of [the next-hop label-forwarding entry (NHLFE) entries] will serve to direct packets coming into an ingress node to the working or the protection path, effectively switching between working and protection LSPs."). Accordingly, Doshi is already equipped to switch back and forth between working and protection LSPs. Building upon Doshi to configure working LSPs as revertive, per Huang, would utilize already existing switching techniques. I also note that Doshi acknowledges that a POSITA would have had the necessary skill set to make "[v]arious modifications of the described embodiments." Doshi, [0300]. It would have been obvious for such modifications to include reverting back to the working LSP after a failure of the working LSP has been cleared as taught by Huang.

212. Moreover, the reasonable expectation of success is evidenced by the fact that revertive switching was described in multiple RFCs discussing MPLS operation. RFC 3209, 12 (describing returning to an "original path when the failed resource becomes re-activated."); RFC 3469, 16 (describing a "Revertive Mode" in which "traffic is automatically switched back from the recovery path to the original working path upon the restoration of the working path to a fault-free condition."). As explained above, RFCs are intended for and familiar to POSITAs. *See, e.g.*, RFC 2026, 6, ("RFCs can be obtained from a number of Internet hosts…"), 8 (IETF documents are "readily available to a wide audience"), 26 (IETF participants "shall

publicly announce…every activity" relating to the standardization process). Indeed, POSITAs had implemented the revertive switching described in these RFCs. *See,* Vasseur '391, [0011], [0040]-[0041], [0067] (incorporating by reference the entire teachings of RFC 3209 and explicitly recognizing that "once P1 [the failed resource] is restored, the redirected traffic will return to its original path").

213. Lastly, the proposed combination relies on the teachings of the references and does not require physical incorporation of elements from Huang into Doshi.

3. Claim 8

a. [8.0] *The method of claim 4,*

214. See analysis at [4.0]-[4.1].

b. [8.1] *further comprising the step of configuring said working entity as revertive.*

215. **First**, as discussed at [2.1], Doshi describes "Switching Between Working and Protection LSPs" and explains that "after detecting a failure, end nodes of an LSP switch traffic from a primary (i.e., working) LSP to its corresponding protection LSP." Doshi, [0266]-[0267]. Doshi also incorporates by reference in its entirety RFC 3209 (Doshi, [0056]), which describes returning to an "original path when the failed resource becomes re-activated." RFC 3209 at, 12. Since Doshi describes switching from a working LSP to a protection LSP after a failure, and since Doshi incorporates RFC 3209's disclosure of returning to an original path that is re-

activated after a failure, it would have been obvious to POSITA for Doshi to revert back from the protection LSP to its original working LSP once the working LSP has been re-activated. That is, it would have been obvious to a POSITA to configure Doshi's working LSPs as revertive. Thus, Doshi renders obvious "*configuring said working entity as revertive*."

216. Second, like RFC 3209, Huang also teaches switching traffic from a failed working LSP to a backup LSP (e.g., a "protection LSP" in terms of Doshi):

> A backup LSP may be set up between the head end node 102A and the tail end node 102B to protect each of the working links in the working bundle such that, **in the event of a failure in the fiber, each of the working links in the working bundle may be switched to corresponding individual backup LSPs**, i.e., a backup bundle.

Huang, [0032].

217. I note that Huang's "working links" are LSPs. *See* Huang, [0030]-[0031] (explaining that each "working link" is associated with an LSP and that a link is a "working link" when traffic is flowing over the LSP). Accordingly, I will refer to Huang's "working links" as "working LSPs."

218. Huang goes on to teach revertive switching such that traffic is switched back to the working LSP once the failure affecting the working LSP has been cleared, which corresponds to "*configuring said working entity as revertive*":

> Additionally, the mesh protection may support **revertive switching** and there may be an option to set a limit on the number of hops in the backup route and the maximum cost (customer defined) that the backup route can have. Note that **under revertive switching, the connection will be switched back from the backup LSP to the working link once the working link has cleared the failure that caused the switchover**, and a provisioned wait to restore period has timed out.

Huang, [0053].

219. A POSITA would have found it obvious to build upon Doshi's teaching of switching between working and protection LSPs by adding revertive switching as taught by Huang, such that the switch configures the working LSP to be used again once the fault is cleared. As discussed at VIII.B.2., a POSITA would have recognized that in instances of quick restoration, e.g., in the case of a mistaken failure diagnosis that is quickly resolved, it would be simpler to revert back to the working LSP instead of calculating the costs for all LSPs to ultimately reach the same result of selecting the working LSP. Moreover, revertive switching was generally known in the art at the time as evidenced by, for example, RFC 3469 which describes a "Revertive Mode" in which "traffic is automatically switched back from the recovery path to the original working path upon restoration of the working path to a fault-free condition." RFC 3469, 16; *see also* Vasseur '391, [0067] (recognizing that "once P1 [the failed resource] is restored, the redirected traffic will return to its original path"). Further support for the combination is provided in section VIII.B.2.

220. Therefore, Doshi in combination with Huang renders obvious "*further comprising the step of configuring said working entity as revertive*," as claimed.

4. Claim 12

a. [12.0] *The method of claim 10,*

221. See analysis at [10.0]-[10.1].

b. [12.1] *further comprising the step of configuring said working entity as revertive.*

222. See analysis at [8.1].

C. Ground 3: The Combination of Doshi, Guichard, and Xu renders obvious claims 14-16.

1. Summary of Xu

223. U.S. Patent Pub. No. 2011/0141877 to Xu (Ex.1025, "Xu") was published June 16, 2011. Xu is titled "Method of simple and efficient failure resilient load balancing." Xu, Title. Xu was not before the Examiner during prosecution of the '821 patent.

224. Like the '821 patent and Doshi, Xu describes selecting paths to project against failures in an MPLS network. Xu, [0001], [0005]-[0006], [0022]-[0023], [0030], [0036], [0038], [0043]; *see* '821 patent, Abstract; *see also* Doshi, [0050]-[0054], [0142]-[0145]. Xu's MPLS network is described as including "multiple paths between each ingress-egress router pair in the network." Xu, [0035]. In Xu, Network-Management Software (NMS) "is responsible for selecting the end-to-end paths" and considers a "set of modeled failure states" when making selection

decisions. Xu, [0006], [0043], [0049]. The set of modeled failure states includes "link failures that occur simultaneously," where each of the modeled failure states has "a predetermined probability of occurrence." Xu, [0006]-[0007], [0049], [0080].

225. In selecting the end-to-end paths, Xu's NMS applies a weight to each failure state based on how common the state is (e.g., based on the probability of occurrence). Xu, [0053]. Paths are selected, for example, to "minimize congestion over the range of possible failure states." Xu, [0053].

2. Reasons to Combine Doshi and Xu

226. A POSITA would have been motivated to combine the teachings of Doshi and Xu, as discussed below.

227. **First**, Doshi and Xu are analogous art to the '821 patent. Just like the '821 patent and Doshi, Xu generally describes MPLS networks. '821 patent, 3:15-18, Abstract; Doshi, [0049]; Xu, [0036], [0038]. Similarly, Xu also addresses the problem of selecting paths within an MPLS network. '821 patent, 4:63-5:8; Doshi, [0014]; Xu, [0002], [0030], [0040], [0043], [0060]. Thus, given the similarities between the references and the fact that they address the problem of selecting paths, a POSITA considering Doshi would have naturally considered the teachings of Xu.

228. **Second**, a POSITA would have been motivated to combine the teachings of Doshi and Xu to produce numerous predictable and beneficial results.

229. Doshi teaches protecting services against link failures by considering disjointedness when selecting primary and protection paths. Doshi, [0155]-[0156]. Doshi explains that strictly disjoint paths share no common links or nodes other than their common ingress and egress nodes and as such a single network element failure is unlikely to impact two different strictly disjoint paths. Doshi, [0055], [0158]. Doshi recognizes, however, that there "will be some cases … where no strictly disjoint path exists" or where "other factors or constraints of the routing algorithm may lead to selection of a protection path, for a given primary path, that is not strictly disjoint from the primary path (e.g., the protection path may share some common links with the primary path)." Doshi, [0055]. Xu complements Doshi by teaching that a probability of occurrence is determined for each of a set of failure states and that path selection decisions consider the probability of failure states occurring. Xu, [0006], [0053]. Xu's NMS applies a weight to each failure state based on how common the state is (e.g., based on the probability of occurrence). Xu, [0053].

230. A POSITA would have been motivated to determine a probability of occurrence of failure states, as taught by Xu, when implementing Doshi's path selection in order to further inform its selection decisions and better protect services against link failures in circumstances where no strictly disjoint pair of paths exists or where other factors lead to selection of a partially disjoint pair as discussed above. Determining the probability of various failure states would allow for a more refined

selection between pairs of paths that have the same disjointedness (e.g., between pairs all having a primary path that shares one link in common with its protection path). Even though such pairs have the same disjointedness, there may be differences in risk of failure for different links. A pair in which the link in common between primary and protection paths of the pair has a higher risk of failure would, everything else being equal, be more likely to concurrently fail than a pair in which the link in common has a lower risk of failure. A POSITA would have been motivated to apply Xu's probability determination to differentiate between such pairs when implementing Doshi's selection techniques. This would reduce the probability of concurrent failure of Doshi's selected pair and thereby improve reliability.

231. The combination is merely the use of a known technique (Xu's probability determination) to improve a similar method (Doshi's method for selecting paths) in the same way with predictable results (selecting paths with lower probability of failure). I also note that the results would have been predictable and there would have been a reasonable expectation of success in the combination given that Doshi and Xu are analogous art and in the same field of endeavor.

232. Implementing the combination would have been within a POSITA's skillset since techniques for determining the probability of network failures were known in the art. *See* Ex.1Ronnen, 10:20-21 (describing a "method for computing various network element failure probabilities"). Xu leaves implementation details of

failure probability determination to a POSITA and thus recognizes that a POSITA would have had sufficient skills to configure the determination. Indeed, Xu explains that "the present disclosure [is] not limited to any specific combination of hardware and software and the computer program code required to implement the foregoing can be developed by a person of ordinary skill in the art." Xu, [0094]; *see also* Xu, [0096]. I also note that Xu states that its disclosure is capable "of being practiced or carried out in a variety of applications." Xu, [0029]. Since Doshi is analogous art in the same field of endeavor, it would have been obvious to carry out Xu's probability determination in the context of Doshi's path selection. In that regard, Doshi acknowledges that a POSITA would have had the necessary skill set to make "[v]arious modifications of the described embodiments." Doshi, [0300]. It would have been obvious for such modifications to include failure state probability determination as taught by Xu.

233. Lastly, the proposed combination relies on the teachings of the references and does not require physical incorporation of elements from Xu into Doshi.

3. Claim 14

a. [14.0] *A system for selecting entities within an MPLS network, comprising:*

234. As discussed at [1.0], Doshi teaches a path selection method where the selected paths are LSPs within an SMDN having an MPLS network architecture,

which corresponds to "*selecting entities within an MPLS network*." The network manager, nodes, components, modules, switches, and corresponding communication paths, involved in performance of the method correspond to the claimed "*system*." *See* Doshi, [0014], [0035], [0081], [0214], [0271].

235. As I further explain below, the prior art renders obvious all of the claimed features that the system comprises ("*comprising:"*).

b. [14.1] *a data structure comprising a plurality of transport entity descriptors;*

236. **First**, as I discussed at [1.1], Doshi in combination with Guichard discloses providing a plurality of LSPs between two end nodes where the LSPs correspond to "*a plurality of [] transport entit[ies]*."

237. **Second**, as discussed at [1.2]-[1.5], Doshi in combination with Guichard discloses calculating the cost of various LSP pairs and selecting the lowest cost pair. A POSITA would have found it obvious to store and reference "*descriptors*" of the LSPs at least temporarily in order to accomplish the cost calculation and selection described above. Such storage would be in a memory, entries in which a POSITA would understand corresponds to a "*data structure*." Moreover, Doshi expressly describes saving information about a plurality of LSP pairs in a data structure during cost calculation:

> In step 1108, the minimum sum is saved in the variable LowKSum and the **minimum-cost pair is saved in the data structure** LowKPair.

Doshi, [0143].

> Next in step 1114, the sum is calculated of the cost of the primary and the restoration path for each of the L pairs and in step 1116, the lowest sum is stored in the variable LowLSum and the **lowest-cost pair is stored in the data structure** LowLPair.

Doshi, [0144].

238. Doshi goes on to describe maintenance of link-state databases that include information used for path computation:

> The information made available to network nodes by these extensions can be used to build an extended **link-state database** just as node link-state advertisements are used to build a "regular" link-state database. The difference is that the extended link-state database (referred to below as the TE/Share database) has additional link attributes. **Uses of the TE/Share database include (1) monitoring the extended link attributes and (2) globally <u>optimized computation of protection paths</u>.**

Doshi, [0196].

239. These link-state databases include data structures containing path information:

> In this scheme, in order to support sharing/restoration, every node maintains a local sharing information database (e.g., the TE/Share database). **This database contains a <u>data structure</u> for each link L that the node owns. <u>The data structure for each link contains</u>**

> **<u>information (e.g., bandwidth, link-id)</u> about all the other links in the network for which link L provides restoration capacity**.

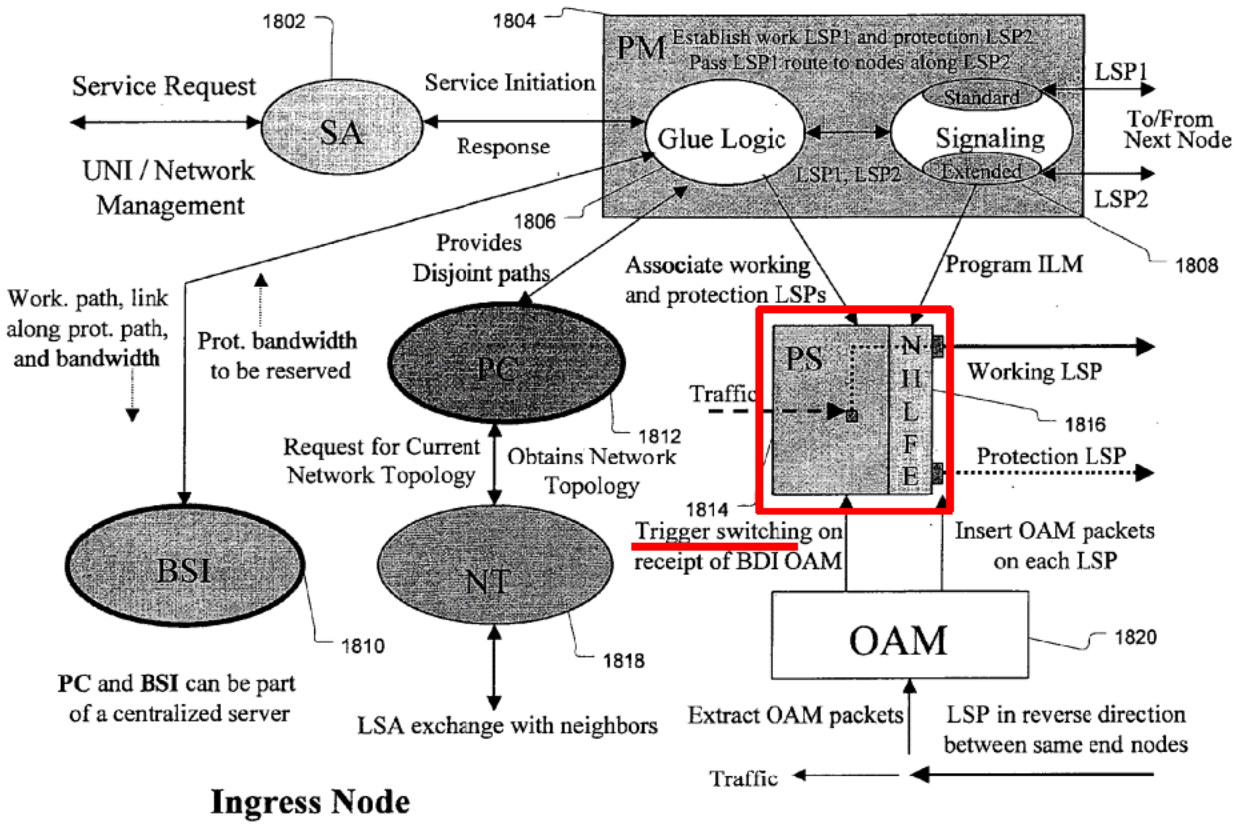
Doshi, [0197]; *see also* Doshi, [0082], [0091]-[0095], [0172]-[0176].

240. Doshi also describes a functional architecture that includes a "Path Management" module that is "responsible for path setup, refresh, tear-down, and monitoring functions." Doshi, [0273]. It would have been obvious to a POSITA to store "*descriptors*" of the monitored paths in order to differentiate the paths and to provide data entries for the paths such that changes can be tracked over time and saved in association with the relevant path.

241. Therefore, Doshi in combination with Guichard renders obvious "*a data structure comprising a plurality of transport entity descriptors*," as claimed.

c. [14.2] *an entity protection switch configured to switch between a working entity and a protection entity; and*

242. As discussed at [2.1], Doshi describes "Switching Between Working and Protection LSPs" and explains that "after detecting a failure, **end nodes of an LSP switch traffic** from a primary (i.e., working) LSP to its corresponding protection LSP." Doshi, [0266]-[0267]. Doshi further discloses a "**Protection Switching (PS)**" module that is "**responsible for switching the affected traffic** onto a protection LSP after detecting a failure or receiving failure notification." Doshi, [0278]. Doshi illustrates one example of the Protection Switching module in Figure 18 showing its inclusion in a node of an LSP:



Doshi, Fig. 18 (annotated).

243. A POSITA would have understood that the node (including the Protection Switching module) or the Protection Switching module itself corresponds to the claimed "*entity protection switch*."

244. Therefore, Doshi renders obvious "*an entity protection switch configured to switch between a working entity and a protection entity*," as claimed.

d. [14.3] *digital logic configured to select said working entity and said protection entity from said plurality of transport entity descriptors, comprising:*

245. First, as discussed at [1.0]-[1.5] and [2.1], Doshi in combination with Guichard discloses logical flowcharts (e.g., at Figures 10 and 11) for selecting an LSP pair from the plurality of MPLS LSPs. The selection includes selecting a primary LSP and selecting a protection LSP. Doshi's "primary" LSP is also referred to as a "working path" (e.g., working LSP). As discussed above at [2.1], these working and protection LSPs correspond to the claimed "*working entity*" and "*protection entity,*" respectively. As discussed at [14.1], a POSITA would have found it obvious to store and reference "*descriptors*" of the LSPs at least temporarily in order to accomplish cost calculation and selection of the LSPs. Accordingly, selection of a working LSP and protection LSP as discussed at [1.0]-[1.5] and [2.1] corresponds to "*select[ing] said working entity and said protection entity from said plurality of transport entity descriptors*."

246. Second, a POSITA would have found it obvious to implement Doshi's flowchart logic (e.g., at Figures 10 and 11) with software or a combination of software and hardware ("*digital logic*") to perform the calculation and selection techniques Doshi discloses a series of modules for performing its calculation and selection techniques. Doshi, [0270]-[0281]. A POSITA would have found it obvious to implement these modules in, for example, software or a combination of software

and hardware ("*digital logic*") since it was well known that such an implementation would allow for execution of instructions to control functionality (e.g., performing calculations and making decisions). *See* Xu, [0092], Fig. 10 (describing that the disclosed techniques "may be implemented using hardware, firmware, software or any combination thereof" and expressly recognizing computer programs and software as "logic"); *see also* Werner, [0072] (describing "computer-executable instructions for performing the methods of the disclosed innovation."); Jungck, 17:29-33 (describing "software and/or firmware provided in a read only memory … for execution" by a processor "to implement the various functions as detailed below."); Rothman, [0021] (describing using "software instructions to implement the present invention.").

247. Software or a combination of software and hardware that executes instructions, for example to carry out the flowchart of Figures 10 and 11, to calculate and select the lowest cost pair corresponds to "*digital logic configured to select said working entity and said protection entit*y." A POSITA would have found it obvious to include such software or a combination of software and hardware, for example, Doshi's "network manager," including the "Path Management" module that is "responsible for path setup, refresh, tear-down, and monitoring functions." *See* Doshi, [0271]-[0273].

248. Therefore, Doshi in combination with Guichard and Xu renders obvious "*digital logic configured to select said working entity and said protection entity from said plurality of transport entity descriptors*," as claimed.

e. [14.4] *logic configured to determine a probability of concurrent failure of said working entity and said protection entity;*

249. **First**, as discussed at [1.2] and [5.1], Doshi's path calculation considers disjointedness. As discussed at [2.1], these paths are working and protection LSPs ("*working entity*" and "*protection entity*," respectively). A POSITA would have understood that probability of concurrent failure is inversely related to disjointedness—as paths become more disjoint, the likelihood they will fail concurrently goes down.

250. For example, Doshi recognizes that when two paths are disjoint, e.g., by being node and link disjoint, "then a failure affecting one of them will not affect the other." Doshi, [0040]. Thus, it would take two distinct failures (one affecting each path) for both paths to fail concurrently. Since Doshi explains that "the **probability of occurrence of more than one failure** during the time interval required to repair a failure **is very insignificant**," a POSITA would have understood that determining that two paths are disjoint with no nodes or links in common is also a determination that the paths have a "very insignificant" probability of failing concurrently. Doshi, [0040].

251. Accordingly, determining path disjointedness (which is part of Doshi's flowchart) corresponds to "*determine[ing] a probability of concurrent failure*" since knowledge of disjointedness represents a determination that there is a "very insignificant" likelihood that the paths will fail at the same time.

252. Second, in circumstances where no disjoint paths having a "very insignificant" probability of failing concurrently exist, or where other factors prevent selection of such disjoint paths, it would have been obvious to build upon Doshi's calculation and selection techniques by applying Xu's determination probability of occurrence for each failure state. *See* Doshi, [0055] (recognizing that there "will be some cases … where no strictly disjoint path exists" or where "other factors or constraints" lead to a selection "that is not strictly disjoint.").

253. Xu discloses a "network architecture [that] uses multiple paths between each ingress-egress router pair" in an MPLS network and provides techniques for "selecting the end-to-end paths" between the pair by "Network-Management Software (NMS)." Xu, [0030]-[0038]. To inform path selection, Xu models failure states that would cause at least one path to fail, including failure states involving simultaneous failures. Xu, [0006], [0049], [0080], Claim 1, Claim 11. A POSITA would have understood that at least some of the modeled failure states would be states in which the failure affects two paths simultaneously and therefore corresponds to a state of concurrent failure.

254. Xu explains that each failure state has a "predetermined probability of occurrence." Xu, [0006]. A POSITA would have understood that a "pre**determined**" probability has been "*determine[d].*" Thus, when a failure state is one in which the failure affects two paths simultaneously (e.g., a state of concurrent failure), Xu's predetermining a probability of occurrence for that failure state corresponds to "*determin[ing] a probability of concurrent failure*." Each failure state is assigned a weight in the selection function based on how common the state is (e.g., based on the probability of occurrence). Xu, [0010]; *see also* Xu, [0033], [0045], [0053], [0080].

255. A POSITA would have found it obvious determine the probability of occurrence of failure states, as taught by Xu, when implementing Doshi's calculation and selection techniques. As discussed at VIII.C.2, determining the probability of various failure states would allow for a more refined selection between pairs of paths that have the same disjointedness (e.g., between pairs all having a primary path that shares one link in common with its protection path). A POSITA would have been motivated to apply Xu's probability determination to differentiate between such pairs when implementing Doshi's selection techniques. Further support for the combination is provided in section VIII.C.2.

256. **Third**, as discussed at [14.3], in view of Doshi, Guichard, and Xu, a POSITA would have found it obvious to use "*digital logic*" to perform Doshi's flow

chart logic, including calculation and selection techniques. As discussed above, it would have been obvious to expand Doshi's calculation and selection techniques to determine Xu's failure state probability of occurrence. A POSITA would have found it obvious to implement Doshi's various modules that would perform the combined techniques in, for example, software or a combination of software and hardware ("*digital logic*") since it was well known that such an implementation would allow for execution of instructions to control functionality (e.g., performing calculations and making decisions). Software or a combination of software and hardware that executes instructions, for example to carry out the flowchart of Figures 10 and 11, including as modified with the teachings of Xu, to calculate and select the lowest cost pair corresponds to "*logic*" for performing the task specified by the instruction(s). It would have been obvious for the instruction(s) to "*determine a probability of concurrent failure*" to be part of Doshi's "Path Computation" module, since this is the module that performs cost calculations. *See* Doshi, [0274].

257. Therefore, Doshi, Guichard, and Xu, a renders obvious "*logic configured to determine a probability of concurrent failure of said working entity and said protection entity*," as claimed.

f. [14.5] *logic configured to determine an entity cost of said plurality of transport entity descriptors: and*

258. **First**, as discussed at [1.2], Doshi's calculation calculates the path cost for each path in a candidate pair so that the path cost can be summed to generate the

cost of the pair together. This is done for all candidate pairs, which means that a path cost is calculated for each path. As discussed at [14.1], a POSITA would have found it obvious to store and reference "*descriptors*" of the LSPs at least temporarily in order to accomplish cost calculation and selection of the LSPs. Accordingly, calculating the path cost corresponds to "*determine[ing] an entity cost of said plurality of transport entity descriptors*"

259. Second, as discussed at [14.3], in view of Doshi, Guichard, and Xu, a POSITA would have found it obvious to use "*digital logic*" to perform Doshi's flowchart logic, including calculation and selection techniques. For example, a POSITA would have found it obvious to implement Doshi's various modules that perform the techniques in, for example, software or a combination of software and hardware ("*digital logic*") since it was well known that such an implementation would allow for execution of instructions to control functionality (e.g., performing calculations and making decisions). Software or a combination of software and hardware that executes instructions, for example to carry out the flowchart of Figures 10 and 11, to calculate and select the lowest cost pair corresponds to "*logic*" for performing the task specified by the instruction(s). It would have been obvious for the instruction(s) to "*determine an entity cost*" to be part of Doshi's "Path Computation" module, since this is the module that performs cost calculations. *See* Doshi, [0274].

260. Therefore, Doshi, Guichard, and Xu, render obvious "*logic configured to determine an entity cost of said plurality of transport entity descriptors*," as claimed.

g. [14.6] *logic configured to reselect said working entity and said protection entity from said plurality of transport entity descriptors upon a reselection event,*

261. First, as discussed at [1.0]-[1.5] and [2.1], Doshi in combination with Guichard discloses reselecting an LSP pair from the plurality of MPLS LSPs in response to an event. The reselection includes reselecting a primary LSP and a protection LSP provided that they remain the lowest cost pair. Doshi's "primary" LSP is also referred to as a "working path" (e.g., working LSP). Put differently, Doshi discloses reselecting a working LSP and a protection LSP. As discussed above at [2.1], these working and protection LSPs correspond to the claimed "*working entity*" and "*protection entity,*" respectively. As discussed at [14.1], a POSITA would have found it obvious to store and reference "*descriptors*" of the LSPs at least temporarily in order to accomplish cost calculation and selection of the LSPs. Accordingly, reselection of a working LSP and protection LSP as discussed at [1.0]-[1.5] and [2.1] corresponds to "*reselect[ing] said working entity and said protection entity from said plurality of transport entity descriptors upon a reselection event*."

262. Second, as discussed at [14.3], in view of Doshi, Guichard, and Xu, a POSITA would have found it obvious to use "*digital logic*" to perform Doshi's flow

chart logic, including calculation and selection techniques. Such selection techniques are simply repeated in the case of reselection, so it likewise would have been obvious to use "*digital logic*" to perform Doshi's reselection. A POSITA would have found it obvious to implement Doshi's various modules for performing the techniques in, for example, software or a combination of software and hardware ("*digital logic*") since it was well known that such an implementation would allow for execution of instructions to control functionality (e.g., performing calculations and making decisions). Software or a combination of software and hardware that executes instructions, for example to carry out the flowchart of Figures 10 and 11, to calculate and select the lowest cost pair corresponds to "*logic*" for performing the task specified by the instruction(s). For example, a POSITA would have found it obvious to include instruction(s) to monitor and reselect the current pair if it remains the lowest cost pair following a reoptimization evaluation in, for example, Doshi's "Path Management" module that is "responsible for path setup, refresh, tear-down, and monitoring functions." *See* Doshi, [0273].

263. Therefore, Doshi, Guichard, and Xu, renders obvious "*logic configured to reselect said working entity and said protection entity from said plurality of transport entity descriptors upon a reselection event*," as claimed.

h. [14.7] *wherein said reselection event is selected from a group consisting of adding an entity to said plurality of transport entities, removing an entity from said plurality of transport entities, an operational status change for one of said plurality of transport entities, and a change in overall cost for one of said plurality of transport entities.*

264. See analysis at [1.5].

4. Claim 15

a. [15.0] *The system of claim 14*

265. See analysis at [14.0]-[14.7].

b. [15.1] *wherein said entity protection switch comprises a 1:1 switch.*

266. Doshi discloses "*wherein said entity protection switch comprises a 1:1 switch*."

267. First, as discussed at [14.2], Doshi discloses the claimed "*entity protection switch*."

268. Second, Doshi describes a 1:1 protection scheme. As Doshi points out, "[i]f the bandwidth were allocated in advance of a failure, this would correspond, in the parlance of the field of protection and restoration for optical transport networks, to **a 1:1 protection scheme**." Doshi, [0060]. Accordingly, Doshi recognizes that some embodiments would use such a 1:1 protection scheme. Doshi's protection switching module in the context of a 1:1 protection scheme renders obvious "*wherein said entity protection switch comprises a 1:1 switch*," as claimed.

5. Claim 16

a. [16.0] *The system of claim 14*

269. See analysis at [14.0]-[14.7].

b. [16.1] *wherein said entity protection switch comprises a 1+1 switch.*

270. Doshi discloses "*wherein said entity protection switch comprises a 1+1 switch*."

271. **First**, as discussed at [14.2], Doshi discloses the claimed "*entity protection switch*."

272. **Second**, Doshi describes a 1+1 protection scheme. As Doshi points out, "If the bandwidth were not only allocated, but additionally if a copy of the service path's data were to be duplicated to the protection path, this would correspond to a **1+1 protection scheme**." Doshi, [0060]. Doshi expressly recognizes there are circumstances "when **a 1+1 restoration scheme is the best option available for restoration**" and notes that 1+1 protection schemes are "popular." Doshi, [0147], [0178]. Doshi's protection switching module in the context of a 1+1 protection scheme renders obvious "*wherein said entity protection switch comprises a 1+1 switch*," as claimed.

IX. CONCLUSION

273. I declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true, and

that these statements were made with knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under section 1001 of Title 18 of the United States Code.

Dated: January 9, 2023

Henry H. Houh

Henry H. Houh, Ph.D.