Computer Networks 77 (2015) 18–41

Contents lists available at ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet

Survey Paper

A survey on methods to provide multipath transmission in wired packet networks

Jerzy Domżał [a,*], Zbigniew Duliński [b], Mirosław Kantor [a], Jacek Rząsa [a], Rafał Stankiewicz [a], Krzysztof Wajda [a], Robert Wójcik [a]

[a] AGH University of Science and Technology, Department of Telecommunications, Al. Mickiewicza 30, 30-059 Kraków, Poland
[b] Jagiellonian University, Faculty of Physics, Astronomy and Applied Computer Science, Reymonta 4, 30-059 Kraków, Poland

ARTICLE INFO

Article history:
Received 11 April 2014
Received in revised form 18 November 2014
Accepted 1 December 2014
Available online 6 December 2014

Keywords:
Routing
Multipath
Load balancing

ABSTRACT

IP networks were designed to provide general connectivity. At their advent, routing methods focused only on finding one optimal path between given endpoints. Although many solutions to sending traffic via multiple paths have appeared over time, the majority of current IP networks are still managed to support only single-path transmissions. This survey examines various approaches which can provide multipath transmissions in existing IP networks. Firstly, the most recognizable solutions are presented, and later, less well-known proposals are introduced. We show how it is possible to realize multipath transmission in source and hop-by-hop routing, multi-topology routing, bio-inspired routing solutions, Valiant's routing, Multi-Protocol Label Switching, Software-Defined Networks, Flow-Aware Multi-Topology Adaptive Routing, Shortest-Path Bridging, Transparent Interconnection of Lots of Links, network virtualization, and Multipath TCP. Moreover, the mentioned approaches are compared, contrasted and subjectively assessed. The goal of the survey is to show that multipath transmissions can be achieved in the current IP networks and in many different ways.

1. Introduction

Wide area telecommunications networks are composed of a multitude of nodes and links. In the great majority of cases there is more than one possible path between any two end-nodes. There are two main benefits of having the possibility of sending traffic via more than one path. Firstly, when failures occur, the traffic can be quickly redirected to alternative path(s) and the network maintains full connectivity. Secondly, although not always employed, the operator has the opportunity to simultaneously use two or more paths between given endpoints in the network, thereby increasing throughput between those points. Load balancing is the natural consequence of multipath transmissions. Therefore, in this paper, load balancing, as a form of multipath transmission is also presented. However, multipath transmissions may also be used for other purposes, e.g. to differentiate the way traffic is served in the network. Distinct paths may be used to meet the requirements of different applications, e.g., use low delay paths for voice traffic or high throughput paths for large data transfers. Additionally, customized paths may be used to meet the requirements of different customers, e.g., by creating a separate network logically disjoint from a public network and with its own topology.

Multipath routing is widely exploited in wireless networks where the needs and benefits of using them are well investigated. Several papers including solution surveys are

* Corresponding author at: al. Mickiewicza 30, 30-059 Krakow, Poland. Tel.: +48 126172846.
E-mail address: jdomzal@kt.agh.edu.pl (J. Domżał).

http://dx.doi.org/10.1016/j.comnet.2014.12.001
1389-1286/

available, including [1,2]. Some aspects of the multipath solutions for wireless networks are also presented in [3]. In wired networks, multipath routing is less popular, although various technologies enable such solutions. The capabilities of multipath routing in wired networks have been underestimated in past years. However, the major reason for this low popularity is a belief that multipath solutions impose significant scalability and complexity problems. Multipath routing requires a relatively more complex network design, optimization and maintenance. Additionally, automated network management and configurations are more challenging.

Solutions for multipath transmissions have been proposed for years. The author of [4] proposed a routing algorithm which enables multipath transmission and minimizes the overall packet delay in the network. The routing tables are updated independently based on information about delay to destination nodes. One of the main features is that during all phases of the algorithm the network is guaranteed to be loop free, even in transient periods. The algorithm is similar to the one used in the Advanced Research Projects Agency Network [5]; however, instead of minimizing only a packet delay, it minimizes the overall delay of all messages transmitted in the network. A class of algorithms based on the method presented in [4] was proposed in [6]. The algorithms allow to find an optimal quasi static routing. They utilize second derivatives of the objective function proposed by Gallager and ensure high speed of convergence and almost insensitivity of performance to variations of external traffic.

The congestion-oriented shortest multipath routing protocol was proposed in [7]. The authors present the protocol which enables multipath routing in packet-switched networks that minimizes the probability of congestion and also decreases packet delays. The protocol is based on packets which are routed on a hop-by-hop basis. Packets are accepted in the network only if there exists at least one path to the destination node which is able to transfer this packet within a finite time. Each router in the network reserves a buffer space for each destination and forwards the packet along one of multiple loop-free paths. Two metrics are used – a short-term metric based on hop-by-hop credits to reduce a link delay and long-term metric based on path delay to minimize end-to-end delay. Moreover, the volume of incoming and outgoing traffic on all router's interfaces is observed. Based on these values, it is possible to choose the best possible paths to the destination node.

An evolutionary architectural framework named BANANAS, which enables multipath capabilities which can be leveraged at different levels in the networking stack was proposed in [8]. In this model, end hosts initiate flows and map them to outgoing interfaces. Several end-to-end paths may be provided by the network through the independent upgrades of selected nodes, possibly placed in different administrative domains. An upgraded node may be aware of only a subset of available paths to destination nodes. BANANAS provides a concept and specifies building blocks to realize this model.

The authors of [9] proposed an iterative algorithm to organize multipath transmission in a network. The presented solution guarantees that transmission demands are achieved. However, it differs from other algorithms presented above because it does not minimize total delay in the network but aims to minimize maximum delays of flows. The theoretical analysis and the results of performed experiments presented in the paper confirm the usefulness of the proposed solution.

Multipath routing introduces an extra overhead to the network as well as a *control plane* and *data plane* of routers [10]. An overhead in the *control plane* encompasses increased storage and computation requirements. The former stems from the need to store more information on network topology and paths. This information needs to be updated. Solving optimization tasks and computing new paths increase the computational requirements. In turn, the path discovery process may require sending additional control traffic and thus imposes additional bandwidth requirements. Path computation and network maintenance may require additional network monitoring and measurements. The overhead in the *data plane* encompasses processing overheads related to packet marking or labeling as well as mapping packets to respective paths. More router memory is needed to store larger forwarding tables. In some solutions there is also some network overhead due to the need to stack extra labels or headers in packets.

A survey of possible approaches to multipath routing accompanied by a discussion of various types of overheads related to them can be found in [10]. The authors discuss various generic concepts and some possible technologies enabling multipath routing with a relatively low overhead. They focus on layer 3. Both intra- and inter-domain approaches are discussed. In contrast, our paper focuses on concrete solutions available at different layers and provides a wide-reaching survey. Some aspects of multipath transmission to solve the routing and wavelength assignment (RWA) problem in transparent optical networks in layer 1 are considered in [11].

In this survey we present, compare and contrast the solutions (protocols, technologies or algorithms) which allow network operators to send traffic between two nodes via multiple paths. Throughout the paper, we consider multipath transmission as a transmission where two or more disjointed paths are used to transmit data between given endpoints. Paths disjointness is considered from the point of view of the layer in which a given solution operates. For example, a network/Internet layer solution must provide multiple paths which are distinctively different in this layer. Paths do not need to be fully disjointed. However, at least one node (link) must be different in each path. The presented solutions concern only wired networks and unicast transmission. Additionally, we do not venture into physical layer solutions to divide traffic into physically disjointed paths as can be done in the optical layer. We have also limited our consideration to multipath mechanisms in a single domain (single autonomous system). The multipath solutions related to inter autonomous system communication is out of the scope of this paper. Therefore we do not deal with inter as multipath solutions such as: various BGP-related solutions, LISP [12] and ILNP [13]. The goal of the survey is to show that multipath transmission can be achieved in many different ways. We

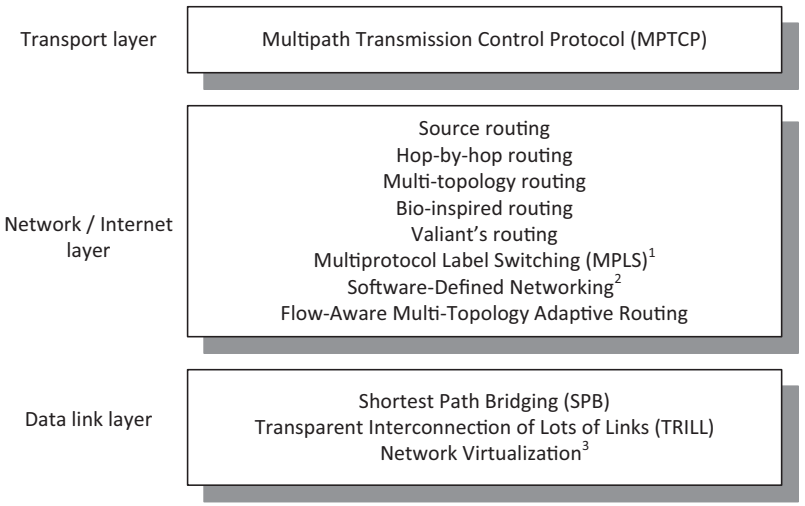


Fig. 1. Presented solutions divided into layers in which they operate.

also investigate the benefits, drawbacks, possibilities and difficulties associated with each presented approach.

We focus our description on nominal situations; however, potential multipath transmission enhances network resilience, i.e. successful recovery from failures. Primarily, without failures, traffic flow is sent using the working path, but one or more backup paths can be precomputed and used when a failure occurs. Such immediate switching from primary to precomputed backup paths is called "protection". A more sophisticated mechanism when backup paths are computed online after failure occurs, using information about network state and specified optimization framework, is also used, and is called "restoration". Resilience can be a part of a protocol, e.g., for Multiprotocol Label Switching (MPLS), or can be supported by a management framework.

The solutions presented in the paper are divided into Sections based on the layer in which they operate. Fig. 1 shows the described solutions. We start with Section 2 where we analyze solutions that work in the most natural layer in terms of path selection, i.e., the Network layer from the OSI/ISO model or the Internet layer from the TCP/IP model. The following solutions are described: load-balancing in source routing and in hop-by-hop routing protocols, multi-topology routing, bio-inspired solutions, Valiant's routing, MPLS,[1] Software-Defined Networking[2] and recently proposed Flow-Aware Multi-Topology Adaptive Routing. Next, three methods from the data link layer are presented in Section 3, namely: Shortest Path Bridging (SPB), Transparent Interconnection of Lots of Links (TRILL), and network virtualization.[3] The last section introducing new approaches, Section 4, is devoted to transport layer solutions in which there is one representative, i.e., Multipath TCP (MPTCP). After having shown all the approaches, we compare and contrast them in Section 5. Section 6 conveys the authors' prediction of how the presented multipath approaches will develop in the future, after which the paper is concluded.

2. Network/Internet layer solutions

In this section we describe solutions that operate at the Network/Internet layer. The network layer means the third layer of the OSI/ISO model, while the Internet layer means the second layer of the TCP/IP model. We begin with legacy IP solutions – hop-by-hop and the source routing. We also describe how the load balancing works. Next we explain how to organize the multi-topology routing and concepts of bio-inspired solutions and the Valiant's routing. At the end of this section, we present the multipath concept in MPLS and in Software-Defined Networks. Since MPLS is usually described as a layer 2.5 protocol, we decided arbitrarily to analyze it in this section. Even though the management of Software-Defined Networks is performed at the application layer, we decided to describe the multipath possibilities of such networks in this section. The reason is that packets are switched at the network layer. Finally, we present the Flow-Aware Multi-Topology Routing mechanism which is a novel solution for realizing multipath transmission at the network layer.

2.1. Legacy IP solutions

Packets in a network may be routed in several ways. In this section we describe two concepts of routing of packets, i.e., source routing and hop-by-hop routing. In source routing, the available paths to destination nodes are estimated in source nodes, and the best are chosen. The identifiers of the nodes across such paths are written to packet headers, and based on them the packets are forwarded in a network. Source routing can be classified as *strict* of *loose*. In the former, a packet must contain information about all the hops

[1] Although MPLS is not strictly a network layer architecture, it was included here as its functions related to multipath routing are best suited to this Section.

[2] We decided to describe Software-Defined Networking at network layer; however, we are aware that while packets are switched at this layer, the management by a central controller is performed at the application layer.

[3] Similarly to MPLS, network virtualization may not be considered as a strict data link layer solution; however, its functions related to multipath transmission are best suited to this Section.

on the path. In loose source routing only some of the intermediate nodes are specified: a packet is routed from one of them to another and must visit all the intermediate nodes. Between the specified nodes, respective routers individually decide where to forward the packet.

Each source node estimates the best paths individually and may use many criteria and algorithms to do so. However, a source node must have access to full routing information for each link belonging to a path. As the described process takes time, packet processing time in the source node is increased. Moreover, the size of a packet header increases rapidly with the size of the network. While each node in a network is aware of all paths to a destination node, it may choose several paths for packet transmission between two nodes. In this way, the concept of multipath routing may be implemented in networks with source routing.

The idea of hop-by-hop routing is very simple. A decision about packet forwarding is taken by each router independently. This means that a source router, based on the available data in a network, chooses the next node with the minimum metric on the path to the destination node. Next, routers must make a decision based on the same assumption. Each node in a network maintains a routing table with next hops for all destinations. The tables are updated, either periodically or if a change occurs in a network topology. The packet header in this concept is much smaller in comparison to the source routing because packets do not have to carry the full forwarding path.

A routing protocol is a formula that specifies how routers communicate with each other. Particularly, routers disseminate information which enables them to select paths (routes) between any pair of nodes of a telecommunication network. There are several routing protocols used in current networks, such as Routing Information Protocol (RIP), Open Shortest Path First (OSPF), and Intermediate System to Intermediate System (IS–IS). Although they operate differently, the final outcome of their operation is the routing table, which is used by the routers to forward packets to proper interfaces.

When a router learns multiple routes to a specific network, it installs in the routing table only the one with the lowest cost (or metric). Standard router operation forces it to use only this route to forward all the packets to this network. More advanced devices can balance the traffic over multiple paths. Such a function, referred to as "load balancing", is equipment-specific and is not standardized by the routing protocols. However, as shown below, certain types of load balancing cannot be used with every protocol.

2.1.1. Equal-cost and unequal-cost load balancing

Whenever a router finds multiple paths with the same cost to a destination, all those paths can be used to balance the traffic. In this approach, called "equal-cost load balancing", only those paths are taken into account. The number of paths used is usually limited by the number of entries a routing protocol puts into the routing table. Depending on the device, its manufacturer, capabilities and software version, this number can vary from 1 (no load balancing), through typical 4, to 16 and even more in some extreme cases. Traffic is distributed equally among these paths. Such behavior is a standard load balancing technique and is supported by all major routing protocols and many device manufacturers. Note that packet forwarding mechanisms in routers do not have a vision of a path. Rather, they see a list of interfaces that supposedly provide access to a certain destination with a given cost. Nevertheless, the term 'path' is used here not to overcomplicate the discussion.

Contrary to the presented approach, unequal-cost load balancing can use multiple paths of different costs. It means that apart from the optimal path (with the lowest cost), some sub-optimal ones can also be used. The *variance*[4] parameter instructs the router how close to the optimal path (in terms of cost) other paths need to be in order to be considered for load balancing. For example, this parameter can force the router to put all the paths whose cost is not greater than 200% of the lowest path's cost into the routing table. Although this is not obligatory, traffic can be distributed proportionally to the costs of the paths. For example, when two paths are taken into account with the costs of 1 and 2, respectively, the better path (with the cost of 1) will receive twice as much load as the second one. In [14] the authors propose an algorithm which makes it possible to assign different weights to the links. As a result, the routing protocol may distribute traffic among all available paths to the destination node proportionally to the weights assigned to the links in the network, and thereby to the costs of paths between the source and destination nodes. The weights may change dynamically, e.g. according to the available link capacity or to the physical distances. The analysis presented in the paper shows that it is possible to support as much as twice the amount of traffic in a network. However, a limitation of this solution is that it is necessary to know the traffic matrix before computing the link weights in a network.

Although, in principle, the above-mentioned approaches work almost identically, in practice there is a difficulty in implementing unequal-cost load balancing. When routers use only optimal paths (regardless of how many are found) to route traffic, all paths are guaranteed to be loop-free, because all of them are on-par with respect to the total path cost. If there are multiple paths of unequal cost, loops can occur easily. Fig. 2 explains this issue. Let us suppose we want to route traffic from R1 to R3. Let us assume that there are two feasible paths: one directly to R3, and the other via R2. The cost of the primary path is 10, whereas the cost of the secondary path is 60. Therefore, it is reasonable to start balancing the traffic in the proportion 6:1 favoring the primary path. However, a problem appears when R2 implements the same mechanism, as it will balance the traffic destined to R3 in the proportion 30:40. This means that a significant quantity of packets will be forwarded back and forth between those routers. The situation is worse if R2 does not support load balancing, as it will bounce all the traffic back, since a route through R1 is its primary path to R3.

[4] The *variance* parameter comes from the IGRP (Interior Gateway Routing Protocol) and its enhanced version EIGRP – the only widely available routing protocols that implement unequal-cost load balancing.

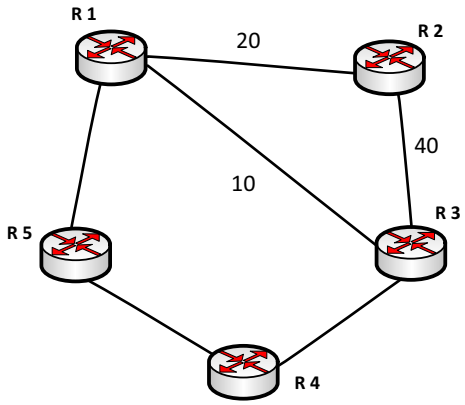


Fig. 2. A network with three routers in a triangular topology; numbers represent link costs.

Given the explanation it is easy to see that equal-cost load balancing can be established in all routing protocols, whereas unequal-cost load balancing is more challenging to implement. Out of all widely available Interior Gateway Protocols (used inside autonomous systems), only IGRP (obsolete) and EIGRP support unequal-cost load balancing. They use other metrics, such as: Advertised Distance and Feasible Distance to find alternative paths. These indicators ensure that a chosen path will never lead to a loop [15]. However, the authors of [16] clarify that the operation of EIGRP might lead to unpredictable and sometimes undesirable effects such as route flapping and instability.

2.1.2. Per-destination and per-packet load balancing

Load balancing makes it possible to establish several paths to the same destination and use them in the packet forwarding procedure. The traffic is distributed among the available paths. Given multiple equal-cost paths exist a router can balance traffic across them in several ways.

Load balancing can be realized in a per-packet or per-destination manner. Per-packet load balancing means that a router sends one packet to a certain destination network over the first path, the second packet to the same destination network on the second path, and so on in a round-robin manner. In per-destination load balancing, a router distributes the packets based on the exact destination address, not the destination network address. Given two or more paths to the same network on which load balancing is active, all packets to the exact destination A go over the first path, while all packets to the exact destination B go over the second path, and so on. There are advantages and disadvantages to both of these approaches.

Table 1 compares per-packet and per-destination load balancing. In per-destination load balancing the routing process is performed based on more information which can be found in the routing table. In addition to the list of possible target interfaces, a router needs to know to which particular one the packet is to be sent, as the process needs to be consistent. Therefore, another table, which typically uses hash functions, is usually required. Such a table is not needed in the case of per-packet load balancing, where a router chooses an interface in a round-robin manner, or simply the one that is least loaded. This also guarantees that traffic is always distributed equally.

Table 1
Per-destination and per-packet load balancing comparison.

	Load balancing	
Feature	Per-destination	Per-packet
Routing decision	Based on destination address	Based on link utilization or round-robin
Packet order	Preserved	Not preserved
Link utilization	Usually unequal	Equal
Route caching	Possible	Not possible
Requirements	Memory and processing power	Processing power

In per-destination load balancing, packet order is preserved because all the packets belonging to a certain destination address follow the same path. However, the fact that per-packet load balancing does not preserve packet order (which is obvious) is not its biggest disadvantage. More important is the fact that route caching cannot be used with per-packet load balancing. The route cache, also known as fast switching is a technique which is still used in some routers or in other environments, such as Linux-based operating systems, to accelerate packet forwarding. The route cache stores recently used routing entries in a fast and convenient hash lookup table, and is consulted before the standard routing or forwarding tables, depending on the vendor. If a matching entry is found, the packet is forwarded and a respective table is not inspected. Unfortunately, because the route cache information includes only one outgoing interface, balancing the traffic is not possible.

Nowadays, modern devices do not usually use route caching. Instead, a forwarding table is constructed based on the information taken from the routing table. Devices that use forwarding tables to process packets do not observe the mentioned limitation and can be used to realize per-packet load balancing.

Additionally, when per-packet load balancing is configured to use the least congested link, monitoring the link loads is required. Although this ensures equal utilization of the links, it is a processor-intensive task and it directly impacts the overall device forwarding performance. This form of per-packet load balancing is not well-suited for higher speed interfaces.

As we can see, the load balancing techniques may be used for realizing the multipath concept in a network. In this solution, it is very easy to find several paths to a destination node and to transmit packets through them.

2.1.3. Other possibilities

Hop-by-hop routing is applicable for many solutions. One of them is multipath routing in networks with connectionless services dynamically adapted to congestions [17]. The proposed routing uses two metrics. The first, called "short-term", is based on hop-by-hop credits to reduce congestion over a given link, while the second, called "long-term", relies on an end-to-end path delay to estimate and

reduce delays from a source to the destination node. The main idea of such a solution is that a packet intended for a given destination may enter the network if, and only if, there exists at least one uncongested path (with sufficient resources) to ensure its delivery within a finite time. Moreover, the routing protocol ensures that all paths are loop free.

The multipath concept realized by implementing hop-by-hop routing is also presented in [18]. In the paper, a new routing primitive (path splicing) that makes it possible to construct network paths by combining multiple routing trees ("slices") to each destination over a single network topology was proposed and analyzed. In this solution, packets may be switched between trees in any network element along their paths to destination nodes. To do this, only a small number of bits in packet headers needs to be changed. The simulation results presented in the paper show that for intradomain routing using slices generated from perturbed link weights the proposed solution is reliable and approaches the best possible paths using a small number of slices. Moreover, the increase of latency is small and no adverse effects on traffic in the network are observed.

Research into multipath hop-by-hop routing has been a hot topic for researchers for many years. In 2008, the new protocol Penalizing Exponential Flow-spliTting (PEFT) was proposed as an alternative to OSPF, which is one of the most representative hop-by-hop routing protocols [19]. PEFT splits traffic over multiple paths with an exponential penalty on longer paths and achieves optimal traffic engineering in a network. It is shown in the paper that the new protocol increases capacity utilization by 15% in comparison to OSPF. Moreover, the time needed to compute the best link weights is significantly reduced.

2.2. Multi-topology routing

Multi-Topology Routing (MTR) allows each router in a network to maintain several valid routes to the same destination over a single IP infrastructure. This increases the possibilities of spreading traffic towards a destination over multiple paths. Multi-topology routing may be used by service providers to engineer traffic in their networks. The topology structure is configured statically.

Different types of traffic flows traverse networks, with different requirements for each. For instance, it is expected that voice traffic will go through links with low latency, jitter and packet loss. File transfer traffic should be transferred though links which offer high bandwidth. When multi-topology routing is used, the flows of two traffic types can follow paths which traverse selected links that can differ completely even if source and destination addresses are the same.

For the purpose of setting up a few different topologies, operators can use multiple OSPF or IS–IS instances (if routers support them). Each instance maintains a separate link state database and builds a separate routing table for each link topology. The router interfaces may belong to different topologies. This approach is inefficient; each protocol instance maintains its own link state database and adjacencies, and sends its own Hello messages. The MTR extension for routing protocols overcomes these problems.

In the MTR approach, a few different link topologies can be defined. Starting from a basic topology, containing all active links, one can specify which links (a subset of the basic topology) belong to a specific topology dedicated for some types of traffic. The same link may belong to a few topologies (Fig. 3). Links belonging to the basic topology are marked with continuous lines. Selected links are also members of other topologies, which are depicted by different dotted lines. When a new topology is created, the total number of routes is increased by the number of routes established in each new topology.

The IP packet entering a router is examined to determine which topology should be used. For instance, the DSCP field can be used for traffic assignment to one of the defined topologies. Separate forwarding tables are used for these topologies; a particular packet can be forwarded in accordance with the specific topology if the destination address exists in the appropriate forwarding table. The content of each routing table used by separate topologies is calculated independently and transferred to the appropriate forwarding table. Information about a link topology membership should be propagated by the routing protocol in the whole network. Fig. 4 shows the MTR basic forwarding model. When a packet arrives at the incoming interface, the specific fields are examined. If the packet marking matches a topology criterion, the associated forwarding table is consulted, the next hop for that topology is determined, and the packet is forwarded. If there is no forwarding entry within a topology, the packet is dropped. If the packet does not match any classifier, it is forwarded to the base topology.

Routing protocols such as OSPF or IS–IS support MTR. The OSPF extension for multi-topology routing (MT-OSPF) is standardized [20]. For IS–IS protocol multi-topology support (M-ISIS) is also standardized in [21].

In MT-OSPF, an identifier (MT-ID) is defined for each topology. It is configured on routers and must be consistent among all routers belonging to the same topology. One instance of the MT-OSPF can be used for information

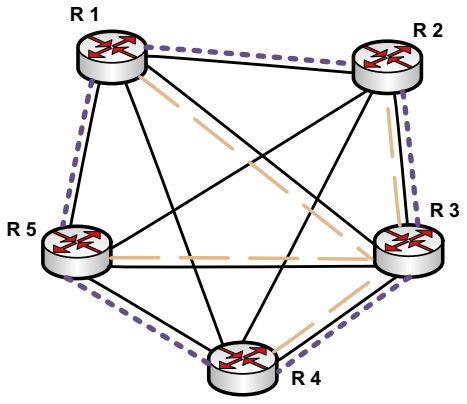


Fig. 3. MTR – two link topologies; different link colors represent distinct topologies. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

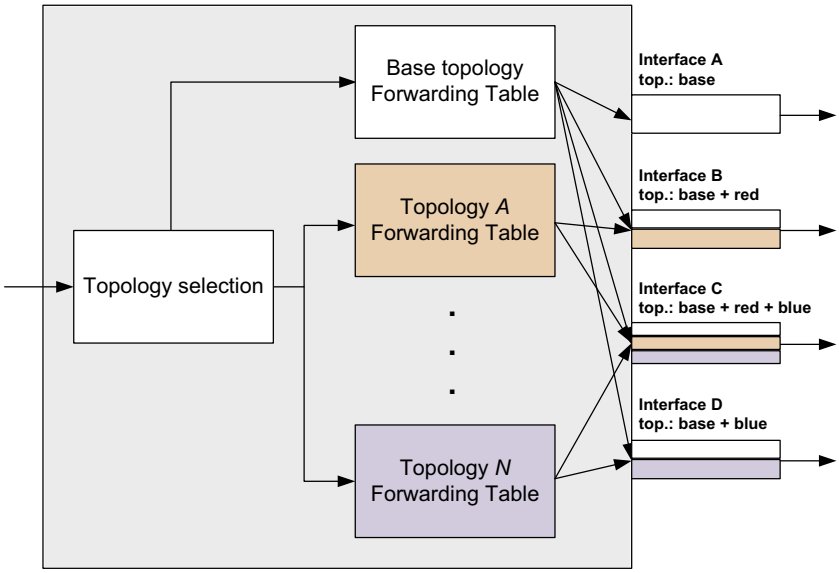


Fig. 4. MTR router forwarding model.

propagation for all topologies. The OSPF link state advertisement (LSA) carries the MT-ID which uniquely identifies the topology. The MT-ID value is copied into the unused service field of the LSA.

The routers exchange topology-specific link state advertisements describing the properties of each link. The LSA describing particular links contains the MT-ID and the metrics which are specific for a link in the particular topology.

IS–IS supports multiple topologies by defining new Type-Length-Values (TLV). They have additional fields for the multi-topology identifier MT-ID. The multi-topology TLVs are used to advertise a router and interface topology membership.

2.3. Bio-inspired routing

A set of concepts and several routing protocols that are conceptually based on observation of animal behavior have been proposed. Bio-inspired algorithms have many applications to solve optimization problems, including finding routing paths, in wired and wireless networking. They also have applications outside networking. This paper focuses only on their applications in finding paths in wired networks and multipath routing support. The family of nature-inspired protocols is usually classified into two main groups: swarm intelligence (SI), and evolutionary algorithms (EA) [22].

2.3.1. Swarm intelligence

Swarm intelligence based protocols are conceptually categorized into two groups: based on ant colony or bee colony behavior while foraging for food.

The majority of applications of swarm intelligence-based algorithms can be found in mobile ad hoc networks (MANET), wireless sensor networks (WSN) and wireless mesh networks (WMN) [23]. They are outside the scope of this paper. SI algorithms for wired networks are less popular and therefore not as thoroughly investigated. However, some solutions which enable multipath routing have been proposed.

Ants are able to find the shortest path to a source of food by using a form of indirect communication called "stigmergy". Ants foraging for food leave pheromone trails on their paths. When going back to the nest ants also leave pheromones, but their concentrations may depend on food quality. Additionally, pheromones evaporate over time. Ants going back to the nest along a shorter path reinforce the pheromone faster than ants choosing a longer path. Other ants are attracted by the pheromone and are likely to follow a path with a more intense pheromone concentration. As a result, the swarm of ants tends to choose the shortest path leading to the best food source in a steady state. It has, however, been shown that success in finding the best path is only statistical and sub-optimal paths are also possible.

Such behavior inspired the authors of the first ant-based routing algorithms [24,25]. The Ant Colony Optimization (ACO) metaheuristic is fundamental for routing protocols proposed since then [24,26]. The basic idea is to implement artificial ants that pass along network paths, collect information, and deposit virtual pheromones at network nodes (routers). In practice, the ants are implemented as agents, usually special types of signaling packets (implementation details depend on the given protocol). Each router maintains a pheromone routing table containing numbers that represent pheromone concentration related to each destination node – outgoing link pair. The decision on forwarding the packet through a given link depends on its final destination and current pheromone concentration. The root element of ACO is stigmergy, although appropriate pheromone control is a key issue in the development of routing protocols enabling efficient finding of an optimal path, offering dynamics and adaptivity to changing network

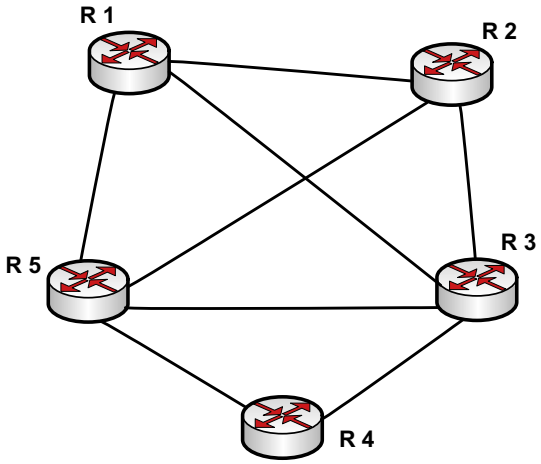
conditions (e.g., failures or congestion), and avoiding stagnation of topology. The main pheromone control mechanisms include: *evaporation*, *aging*, *limiting and smoothing pheromone*, and *privileged pheromone laying* (for more details see [22,27]).

Fig. 5 shows an example of a pheromone routing table for node R4. It has two neighbor nodes: R3 and R5. All other nodes can be reached via these neighbors. The suitability of a path to each possible destination leading through each neighbor is represented by the pheromone concentration stored in the table. Nodes may also store a matrix known as the statistical parametric model, which can be used by the ant-routing algorithm for pheromone control and routing decisions. Fig. 5 shows a matrix for node R4 starting a triplet of parameters for each destination. The parameters could be as follows: an estimated average trip time to a destination, standard deviation, and the best measured trip time between R4 and the given destination [24,28].

ACO-based routing protocols offer single-path or multipath routing. The single-path routing protocol examines a pheromone table and, for a given destination, chooses an outgoing interface with the highest concentration of pheromone – that is, the shortest path in terms of the metric used by pheromone control. Since the pheromone concentration is constantly changing, e.g., due to varying network conditions, once a new interface achieves the highest concentration of pheromone, the new best path is used.

Multipath routing can be realized in various ways. In the simplest approach the outgoing interface is chosen with a probability proportional to the pheromone concentration. Such an approach was proposed in the *AntNet* algorithm to provide load balancing on a per-packet basis [24]. If several routes are available for a given destination the number of selectable ones may be limited to perhaps two or three with the highest pheromone concentration. Packets or flows are routed over selected interfaces with the probability proportional to the pheromone concentrations or distributed statistically equally over them. Such an approach is robust against failures and allows for load balancing. If one of the links becomes unavailable or congested, the related pheromone concentration decreases. Then the probability of routing traffic over this link decreases (to as low as zero in the case of link failure), although other routes are still available and active. One ACO metaheuristic based solution that enable multipath routing is *AntNet-FA* [22].

The second group of approaches assumes the existence of multiple types of ants. Several independent ant colonies exist, each of which recognizes only its own type of pheromone. Routers maintain multiple pheromone routing tables. The resulting solution is similar to MTR, since each



Destination	Neighbor: R3	Neighbor: R5
R1	0.1	0.3
R2	0.9	0.4
R3	0.1	0.5
R5	0.2	0.8

(a)

Destination	Parameter: $p1$	$p2$	$p3$
R1	2.2	0.5	0.2
R2	1.3	0.3	0.1
R3	1.4	0.5	0.3
R5	1.2	0.2	0.2

(b)

Fig. 5. Example of a network with ACO based routing; node R4: pheromone routing table (a) and statistical parametric model (b).

type of ant may in fact create its own topology. Different ant colonies may use distinct metrics and be dedicated to different types of traffic, specific constraints, etc. This approach was widely exploited in several proposed solutions for enabling load balancing (e.g. *Multiple Ant Colony Optimization* (MACO) [27]) and QoS routing (e.g. *AntQoS* [29] or *AntNet-QoS* [30]). The latter integrates the *AntNet* approach with Differentiated Services (DiffServ). Each class of service may be dedicated to a separate ant type and create a distinct topology. Ant type may be bound with DSCP.

The second group of SI solutions is inspired by the natural behavior of honey bees, including such activities as foraging and mating. A Bee Colony Optimization (BCO) metaheuristic has been proposed in [31]. The best known multipath routing protocol proposed within this group for wired packet networks is *BeeHive* [32]. One of the design principles of BeeHive was the ability to find and maintain multiple paths between node pairs. Similarly to ant-based protocols BeeHive routers maintain routing tables storing information on neighbor nodes that may be used to reach a given destination with a metric expressing a quality of each path. Paths are discovered by bee agents. The network is divided into *foraging zones* and non-overlapping *foraging regions*. Two types of agents are used to create and maintain them as well as exploring the network and collecting information on the quality of links: *short distance bee agents* and *long distance bee agents*. Agents estimate the quality of each discovered path (a delay is usually used as a metric). Information about link quality is stored in an *intra foraging zone* routing table. Each node also stores a *inter foraging zone* routing table that maintains values of the quality metric for reaching a *representative node* of a *foraging region*. Finally, *foraging region membership* table provides mapping of known destinations to *foraging regions*. The algorithm does not need global network information. It works with a local view of network topology discovered by *short distance bee agents* that collect local information within a *foraging zone*. This approach makes the protocol scalable in terms of network overhead, processing power and algorithm complexity. The network overhead introduced by bee agents is below 1% while the complexity is claimed to be lower than in OSPF [33].

The BeeHive algorithm is similar to ant colony-based solutions. One of the main differences is that the route discovery policy is deterministic rather than probabilistic. The selection of the next hop for a data packet (choosing from multiple possible paths) can be either probabilistic or deterministic.

2.3.2. Evolutionary algorithms

Evolutionary Algorithms (EA) have been inspired by natural evolutionary processes of living beings. Some routing protocols based on EA have been proposed. Similar to ant-inspired solutions they are capable of providing multipath routing. The network is explored by agents launched at network nodes. Agents discover candidate paths and evaluate their quality. The evolutionary algorithm is responsible for the selection of best paths and supporting dynamic changes to avoid stagnation of established routing paths. The algorithm operates on individuals, chromosomes and genes [22]. An individual is a solution generated by an evolutionary algorithm; it represents the path found between a given source – destination pair. It is a string that consists of a sequence of nodes from source to destination. A chromosome is a new solution discovered by an agent. It is a sequence of nodes traversed by the agent. Finally, a gene is part of a node sequence. The quality of a given chromosome (expressed as e.g., trip time or hop count) is evaluated by an agent going back from destination to source node along the sequence of nodes represented by the chromosome. Then the algorithm tries to find new solutions (individuals) using the following three operators: selection, crossover, and mutation. The selection operator finds n best individuals from the previous generation for replication in the next generation. Those best individuals evolve in the next step, while poor solutions are forgotten. The crossover operator exchanges partial solutions (genes) between selected individuals and newly found chromosomes. Finally, part of the solution of an individual is mutated randomly. In this way, best paths found in previous generations are kept, but crossover and mutation ensure that new paths are found and stagnation is avoided. Poor routes are deleted. This algorithm enables finding multiple paths. Some realizations support multipath routing. Examples of the implementation of the EA algorithm for routing are *Genetic Adaptive Routing Algorithm* (GARA) [34] and *Synthetic Ecology of Chemical Agents* (SynthECA) [35]. The latter is capable of realizing multipath routing.

2.4. Valiant's routing

Valiant's load balancing for processor interconnection networks was introduced in [36] and the recent results for scalable routers with performance guarantees were presented in [37,38]. In [39], the concept of Valiant's load balancing for backbone network design was analyzed. It assumes that the outgoing interface may be randomly selected among all non-congested ports. The solution works only in a fully-connected logical mesh network. However, the links may not be physical connections but may also be implemented by tunnels or an overlay mechanism.

Valiant's load balancing routing assumes that traffic is sent over two-hop paths and, therefore, it is very easy to estimate the aggregate traffic which enters and leaves nodes. Each packet entering a backbone network traverses two links between ingress and intermediate nodes and between intermediate and egress nodes. All traffic is spread equally among all available links in a network and flows are load-balanced in ingress nodes across all available two-hop links to egress nodes. This solution is easy to implement in nodes where only one additional table is needed to send packets to the proper ports and to track the available paths. While such a solution ensures maximum usage of available capacity in a network, it has one main disadvantage. Packets of flows are delivered to destination nodes through different paths and as a result may reach egress nodes in undesirable sequences. This problem may be solved by hashing flows and blocking transmission of packets which belong to the same flow through different paths. However, this simple way of providing multipath

transmission is invaluable when failures occur in a network. In such a case packets are easily redirected to other paths. Failure recovery and restoration is as fast as failure detection at a single router.

An example of a network with Valiant's routing scheme is presented in Fig. 6. The routers are connected one to another by physical or logical links. The two-hop possible paths from R1 to R4 are presented with dotted lines.

Such a concept is radically different to routing methods currently used in backbones. However, it ensures predictable and guaranteed performance, even in the case of a network element failure or when a traffic matrix changes. Moreover, in networks with Valiant's routing, convergence time after a failure is very short, which supports transmission of real time applications. Another advantage is the minimum capacity needed to ensure maximum link usage in comparison to other architectures [39].

The described solution appears to be similar to MPLS; however, the techniques are different. Valiant's load-balanced networks are fully automated, while in MPLS the paths need to be set and torn down, usually before transmission starts. Moreover, MPLS needs complex protocols to switch paths when failures occur.

Valiant's load balancing is considered for use in OpenFlow [40] switches as an element which allows it to increase speed of traffic [41].

2.5. Multipath routing in MPLS networks

MPLS [42] is a popular networking technology and has been used for several years. With MPLS, the idea is to label ingress IP packets based on their destination address or other preconfigured criteria. These labels enable the routers to forward the traffic by looking at the label and not the destination IP address. The MPLS labels are advertised between routers so that they can build label-to-label mapping.

Together with path establishment protocols such as RSVP-TE, MPLS is essentially a technology for establishing Label Switched Paths (LSPs) between any pair of routers in a network domain [43]. In an MPLS network, ingress routers may establish one or more paths to a given egress in the MPLS domain. While multiple LSPs are available, the goal of the ingress node is to distribute the traffic across the LSPs so that the network utilization and the network performance perceived by users are enhanced.

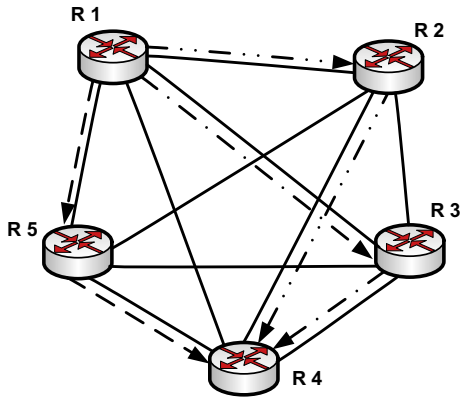


Fig. 6. Example of a network with Valiant's routing (paths for traffic from R1 to R4 are indicated).

2.5.1. Load-balancing algorithms in MPLS

To balance the load in MPLS networks, both static and dynamic algorithms are available. Generally, methods utilizing traffic statistics, linear programming, and analytical approaches for determining traffic assignment are used for traffic assignment in a multipath environment.

In the first group of algorithms, historical traffic statistics collected over time are used to determine LSP topology and distribute traffic over multiple paths [44]. Such calculated LSPs typically change on a relatively long time-scale, and do not attempt to adapt to unpredictable traffic variations or changing network conditions.

The algorithms based on linear programming are used to optimize offline flow allocation in MPLS-traffic engineering (MPLS-TE) networks [45]. In such optimization methods some goal functions, such as minimizing congestion, bandwidth consumption and operational costs, can be considered.

The third group of algorithms use analytical approaches for determining the traffic allocation to individual paths. Such a method was adopted e.g. in [46], where a stochastic framework based on an M/M/1 queuing model is presented. Within this framework, a set of parallel edge disjointed LSPs is modeled by parallel queues.

The dynamic load balancing algorithms in an MPLS multipath network utilize dynamically changing network status information in order to determine the set of LSPs to be used in traffic delivery and/or traffic proportioning among those paths.

An approach where both above the mentioned processes are performed is used in the MPLS optimized multipath (MPLS-OMP) solution [47]. At the MPLS ingress router an algorithm is applied to select additional paths if congestion persists in the current path set. Then, the traffic splitting ratio is adjusted and the traffic load is distributed over the newly updated path set. A similar mechanism was proposed in [48]. In this approach, when the shortest path between the considered ingress and egress is congested, the developed algorithm is used to find a low load subshortest path for the congestion path, based on the bandwidth utilization and the topology.

However, most dynamic load balancing algorithms proposed for MPLS networks assume that the set of candidate paths for an ingress–egress pair is fixed. The main goal of such algorithms is to avoid network congestion by adaptively balancing the load among multiple paths. Traffic assignment to the established LSPs may be based, for example, on measurement and analysis of path congestion [49,50].

Some dynamic load balancing algorithms distribute incoming traffic based only on the average LSP delay measurements [51]. However, such delay-based measurements cannot reflect the congestion state accurately in networks with low-speed and high-speed links. In this kind of network environment it is difficult to determine whether the source of the measured delay is the speed of a link or heavy network traffic.

To overcome these problems, some algorithms which utilize more network parameters were proposed. Most of these distribute incoming traffic based on directly measured metrics such as the number of hops, LSP delay, and packet loss probability [52,53]. Such algorithms distribute traffic among LSPs according to the measured parameters of each LSP adaptively, which can avoid congestion results from shortest path forwarding in traditional IP routers.

The authors of [54] propose the end-to-end Self Protecting multipath (e2e SPM) which protects flows in MPLS after a network element failure. They assume that after a failure, traffic may be redirected to more than one backup path, and may be distributed among several backup paths. As a result, multipath routing makes it possible to save backup capacity needed in the event of failure. The authors explain that the proposed solution is easy to configure and no signaling is needed. Simulation analysis presented in the paper shows that in comparison to OSPF, e2e SPM needs just 20% of the OSFP extra bandwidth in case of failure in the analyzed topology. Moreover, the proposed solution is faster to respond to failure than OSPF.

2.5.2. GMPLS support for multipath

An important (and somehow natural) extension of the MPLS framework is the Generalized Multi-Protocol Label Switching (GMPLS) [55] open set of networking and switching protocols. In contrast to MPLS, which is dedicated exclusively to packet switching, GMPLS extends switching capabilities to time division multiplexing (TDM), layer-2 switching, wavelength switching, and fiber-switching.

The wide selection of path establishment proposals in MPLS is complemented by the Path Computation Element (PCE) proposal [56]. The PCE concept is a flexible solution designed for path computation in MPLS and GMPLS networks. The main goal of PCE is to compute a path in large, multidomain and also multilayer networks using any specific model, metric and interworking architecture. Typical usage of PCE assumes the implementation of a single entity per domain, i.e. supporting a decentralized approach. However, for complex multidomain networks the centralized model is also considered but is difficult to implement in practice, due to computational and signaling limitations. The PCE concept allows multipath computation in order to make load sharing possible and improve resilience.

The PCE concept is loosely defined as a whole; however, any specific proposed solution needs to be fast, optimal and scalable, making it possible to spread the path computation efforts among cooperating PCEs with an option to reoptimize. They also need to be robust against network instability and situations when computation of paths satisfying the required set of constraints is not possible [56]. The PCE concept and architecture was recently completed by the specially-designed Path Computation Element Communication Protocol (PCEP) [57], allowing functional interworking among PCE instances located in different domains.

2.5.3. Multiprotocol Label Switching Transport Profile

Multiprotocol Label Switching Transport Profile (MPLS-TP) aims to enhance the Operation, Administration and Maintenance (OAM) functions that are insufficiently addressed in the MPLS. These functions aim to make MPLS comparable to SONET/SDH and OTN in terms of reliability and monitoring capabilities. An MPLS-TP network should be operated in an SDH-like manner, and a network management system (NMS) should be used to configure connections. Connection management and restoration functions, however, can alternatively be provided utilizing the GMPLS control plane protocols which are also applicable to the MPLS-IP data plane. According to [58], the ECMP load-balancing must not be performed on an MPLS-TP LSP. Such LSPs may operate over a server layer, where load-balancing is supported; however, it must be transparent to MPLS-TP. In [59] the recommendations for anyone who plans to define an application to run over an MPLS network and who wish to avoid packet reordering as a result of the ECMP load-balancing are presented. The recommendations, which are in fact an Internet Best Current Practices for the Internet Community, rely on inspection of the IP version in packet headers. The goal is to avoid multipath transmission of packets of one flow, which may cause packet reordering or not acceptable jitter.

2.6. Software-Defined Networking

Software-Defined Networks (SDN) become more and more popular in currently performed research and in real networks. The fundamental assumption of SDN is the ability to program the external software controller which is responsible for traffic engineering in the domain. Moreover, the interfaces should be open and able to serve signaling traffic generated by different protocols. As the authors of [60] noticed, SDN have the ability to dynamically modify network parameters and to enable multipath transmission. It is possible to change connectivity in e.g. data center networks every few minutes or even seconds. As a result, the priority traffic may be sent to the destination node through paths with minimum delay. The central controller is able to select paths based on information about flows received from the network or from the application. In some network services the load balancing possibility is more than welcome. Usually, online services are replicated on at least two hosts. For the reliability, availability and efficiency reasons the load balancing is very useful. In SDN it is possible not only to implement basic functionality of load balancing but also to engineer the traffic at flow level. This gives a wide spectrum of possibilities to manage traffic sent through different paths.

Multipath transmission in SDN can be realized in many ways. For example, the authors of [61] propose a new solution based on SDN which integrates Dynamic Load Balancing Multipath (DLBMP) with the congestion control algorithm. In the proposed mechanism the information about the link load from edge routers is directly sent to the central controller which controls admission procedures of incoming traffic. As a result, source nodes can react faster to traffic load in a network and the probability of packet losses is minimized. Moreover, as traffic can be distributed through many paths, the bandwidth is utilized better.

Software-Defined Networking architecture is also applied in a private WAN – B4, which is a network connecting Google's data centers around the globe. This network

has a unique characteristics: huge demand on bitrates, diverse traffic and the necessity to provide full control. Such specific requirements led the network administrators to the conclusion that Software-Defined Networking is a good choice to manage their network [62]. As a result, the multipath approach implemented in the network allows for around 70% utilization of links in the network. This results in 2–3x efficiency improvements in relation to standard WAN. Moreover, it is possible to prioritize the selected traffic according to its needs.

SDN is currently perceived as a hot research topic. Many scientists and engineers work on new mechanisms and improvements of existing ones which work based on the Software-Defined approach. One of the most popular issues are related to multipath transmission.

2.7. Flow-Aware Multi-Topology Adaptive Routing

Flow-Aware Multi-Topology Adaptive Routing (FAMTAR) is a new proposal for realizing multipath transmission in IP networks [63,64]. This solution, as the name implies, performs traffic management based on flows. A flow may be identified in any way, e.g. based on source and destination addresses, source and destination ports and transmission protocol. Each router in the network is equipped with flow forwarding table (FFT), where at least identifiers of flows, identifiers of outgoing interfaces and flow time stamps are written when the first packet of a flow arrives at the router. Time stamps are updated with each consecutive packet of the flow. The identifiers of flows are added to FFT based on current routing table maintained by the routing protocol. The content of the routing table changes dynamically according to the status of outgoing links connected to the router. When a link in the network becomes congested, its cost is changed to a high value and the proper information is propagated by the routing protocol. As a result the paths containing such a link in most cases will be replaced with new ones (with lower total cost). On the other hand, when a link becomes uncongested, its cost is changed to the original value and new flows may be accepted on paths which contain such a link.

The operations on FFT are illustrated in Fig. 7. Three first flows sending traffic from S to D are accepted in the direct link between R1 and R3. After acceptance of the third flow this link becomes congested and its cost is changed to 1000. Then, the fourth flow is accepted on the path through R2 with total cost equal to 2 (lowest in the network). All flows accepted before the link cost change, remain on their path. New flows, however, such as flow 4, use a new path.

One of the challenges of FAMTAR is how to make a decision that a link is congested. One possibility is to observe link load, packet drops or delays. However, such estimations should be performed in real or near real time. As a result the complexity of the algorithm is greater than in routers used currently. Moreover, FAMTAR routers must have enough computational resources to ensure proper operations in the network. The main goal of FAMTAR is to improve transmission parameters in the network. The authors of [65] noticed that in highly loaded network, all paths to the destination network may be congested. In such a case all new flows are accepted on the path initially selected by the routing protocol – when all links were uncongested. To solve this problem, the admission control mechanism was proposed. It allows for acceptance of new flows only in links which are not congested. In more advanced version of this algorithm, also presented in [65], the border router may decide whether to accept a new flow or not based on the total cost of the available path. The simulation results confirm that FAMTAR is an efficient mechanism and allows to significantly improve transmission parameters in the network.

3. Data link layer solutions

Currently, the predominant technology for exchange of data at the link layer is the Ethernet [66]. The Ethernet standard is extremely popular in LAN networks. Ethernet is also starting to be used in metropolitan and wide area networks. Such proposed standards as IEEE 802.1ad, IEEE 802.1ah or IEEE 802.1ag, alongside the possibility of sending data at 40 Gbit/s and 100 Gbit/s speeds, facilitate the introduction of the Ethernet in carrier networks. An

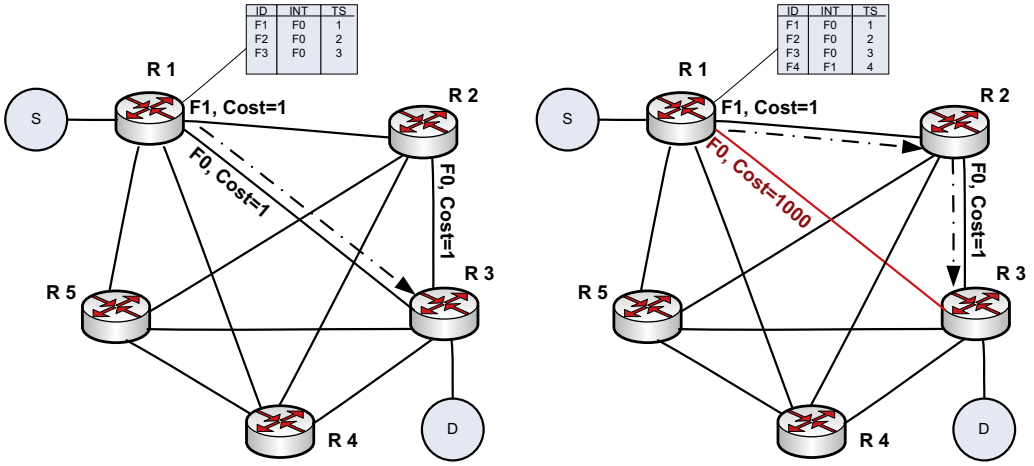


Fig. 7. Example of a network with FAMTAR.

important feature tackled by Ethernet standardization bodies is the ability to carry synchronization signals for mobile backhaul networks. Synchronization provided by the Ethernet may vastly speed up widespread introduction of the standard in some transport networks.

The technology is well known, understood and widely implemented. However, new approaches to control Ethernet forwarding have been proposed recently. The designed solutions vastly improve the functionality of the Ethernet including the ability to provide multipath routing.

3.1. Shortest Path Bridging

Taking multipath routing into account, one of the most important standards is IEEE 802.1aq [67]. This standard specifies Shortest Path Bridging (SPB) of unicast and multicast frames as well as the protocols to calculate multiple active topologies. SPB reuses the IS–IS routing protocol [68,69] which is proved to be scalable, well understood and well behaved even in large networks. The extension of the IS–IS protocol to operate in Provider Backbone Bridging, i.e., enhanced Ethernet standard, is minimal. The aim of SPB is to provide the ability to operate in large networks e.g., in a 1000-node Ethernet network. In SPB, simultaneous use of multiple equal shortest paths is ensured. The SPB sets up at least one shortest path tree (SPT) in a node. So far 16 diverse tunable shortest paths between a pair of nodes are allowed. However, this number may be extended further. The identification of a tree is based on a VLAN ID (SPBV) or a MAC (SPBM) address. If the VLAN ID is used then well-proven MAC learning is preserved. If MAC addresses are used to distinguish trees, then MAC learning is turned off and a forwarding database is computed from the IS–IS database. However, it should be noted that in this case the IEEE 802.1ah standard, known as a MAC-in-MAC, must be used. Ethernet is extremely vulnerable to well-known problems with looped frames. Therefore, the port of arrival of a frame is audited to check whether frames arrive at the port leading to the station with a given source address.

Data is assigned to a given path at an ingress node. Traffic assigned to a path may be based on the service or other criteria, according to the operator's preference.

So far, the IEEE 802.1aq standard does not ensure per-flow or per-packet load balancing between a pair of end stations unless these end stations use unique VLAN identifiers for each flow. A new working group was created in order to allow the use of multiple next-hops for frames within a single service in the SPBM network. The group is working on the extension of the SPB standard and is titled IEEE 802.1Qbp – Equal Cost Multiple Paths. The work is currently ongoing [70].

3.2. Transparent Interconnection of Lots of Links

Transparent Interconnection of Lots of Links (TRILL), standardized by the IETF, is a solution for transparent unicast shortest-path frames routing at the link layer [71]. Implementation of the TRILL should make it possible to diminish one of the main drawbacks of Ethernet networks, i.e., excessive and not optimal usage of some links. With a spanning tree established in an Ethernet network, frames are sent or aggregated to small subset of available links. All other links are unused in order to avoid creating loops. The latest spanning tree protocols, both proprietary and standardized, make it possible to build more than one tree; however, the number of trees remains low, and a significant amount of configuration is needed. Moreover, using the spanning tree protocols may lead to vast changes of the tree, and the path may be switched as a result, even if the change in the network is small. Such changes usually take time to propagate and converge. In the approach proposed by the IETF TRILL Working Group (TRILL WG), an attempt to combine strengths of link and network layer protocols is made. The TRILL should make it possible to improve path efficiency and stability of data paths. Moreover, it is able to use more than one path between a pair of nodes [72]. The TRILL is not limited to the Ethernet standard, and it may be implemented in other link layer technologies such as the Point-to-Point Protocol (PPP). The specification of TRILL for the PPP protocol is published in [73]. The TRILL WG specifies the Routing Bridges (RBridges) which make it possible to use link state routing in a VLAN-aware network when used with the TRILL protocol [71]. The RBridges run a link state routing protocol between them to gather and distribute information about the network. Subsequently, the link state routing protocol computes a pair-wise optimal path for unicast traffic.

An example of conveying a frame in a network with an implemented TRILL protocol is shown in Fig. 8. When an end user frame is received by an edge RBridge (named as RB1 in Fig. 8), it is encapsulated by two headers: the TRILL and Outer Ethernet header. The TRILL header is composed of the Ingress Nickname of an ingress RBridge, the Egress Nickname for an egress RBridge and hop count, among others. The TRILL Nicknames are based on the IS–IS IDs and the hop count is used in a similar way as a Time-To-Live (TTL) value in IP packet.

The Outer Ethernet header is a typical Ethernet header and includes source and destination MAC addresses, an optional Tag, and the Ethertype (set to a new TRILL value). The MAC addresses in the Outer Ethernet header are changed on a path from a node to another node to reflect the source address of the transmitting node and the destination address of the receiving node. On the link from RB2 to RB3, the source and destination addresses in the Outer

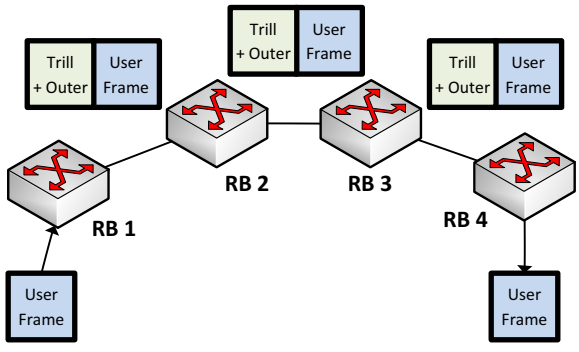


Fig. 8. Transport of data in a TRILL network.

Ethernet headers are RB2 and RB3 MAC addresses, respectively. Therefore, from a frame forwarding point of view, an RBridge behaves as a network layer node by replacing the link layer header. Additionally, the value of the hop count in the TRILL header is decreased. It should be noted that Frame Check Sequence (FCS) must be updated on encapsulation, decapsulation and every TRILL hop due to changes in the Outer Ethernet header source and destination addresses and the value of hop count.

At the edge of the TRILL network (RB4 in Fig. 8), the egress RBridge decapsulates the frame by removing the Outer Ethernet and TRILL headers.

In the ingress and egress RBridges, the MAC addresses of end stations are learned. In an ingress RBridge the forwarding is performed on the basis of data gathered from the IS–IS protocol: the egress RBridge is selected, the TRILL header is built next, then the Outer Ether header is added, and the frame is sent to the next node on the path from the ingress RBridge to the egress RBridge. In a transit RBridge, the forwarding table built on the basis of the IS–IS protocol is used to find the next hop, compose the Outer Ether header, and decrease the hop count; finally, when FCS is updated, the frame is sent towards the destination. In the egress RBridge, the Outer Ether and TRILL headers are removed from the frame, an outgoing interface is sought in the MAC address table, and the frame is transmitted to the destination. Both RBridges and traditional bridges without implemented TRILL protocols may be used in a network. In such cases, traditional bridges are transparent to RBridges. The TRILL uses an extended IS–IS protocol to select a path for a frame. Using the IS–IS simplifies the implementation of the TRILL protocol since it may run directly over the link layer without a need for IP addresses. Moreover, the IS–IS protocol uses the Type-Length-Value concept, hence it may be easily adapted to new needs. The option to use multiple paths in a TRILL network is provided by the IS–IS protocol. As a result, the outcome is relatively similar to the network with the SPB standard.

3.3. Link aggregation group

The concept of Ethernet link aggregation or bundling has been used by vendors for several years. Various solutions have different methods of aggregation. This leads to certain problems with aggregation between nodes produced by different vendors. Therefore, in 2000, an extension to the traditional Ethernet protocol was standardized. The newest version of the standard was proposed in 2008 [74]. The standard defines the option to combine several physical links into a bundle known as a Link Aggregation Group (LAG) [74]. A LAG is seen as a single link by a MAC Client layer. The LAG makes it possible to increase bandwidth utilized to carry data between a pair of nodes. Using LAG, bandwidth may be increased more linearly than by order of magnitude, as it is performed with each generation of Ethernet physical layers standards. The traffic may be shared among all links in a group; however, it is not spread on the path but on the link level. Moreover, link aggregation enhances resilience, since data may be moved from one corrupted link to another in a group. This is a significant improvement: without this functionality it is not possible to use more than one active link, since an extremely dangerous Ethernet loop is created. The IEEE standard [74] does not specify a mandatory distribution algorithm for selecting a link from a link aggregation group; however, any used link selection method should ensure that frames disordering and duplication are avoided. A wide spectrum of parameters is generally utilized by the distribution algorithm, including source and destination MAC addresses, reception port, Ethertype, network layer source and destination addresses, and transport layer port numbers. It should be noted that the LAG does not increase the bandwidth for a single flow, but it means that a high bandwidth with multiple flows transported between a pair of nodes can be achieved.

The link aggregation group can be constructed automatically using the Link Aggregation Control Protocol (LACP) standard, a proprietary solution, or manually. The standard specifies that a LAG may be created using links with the same speed and with a full-duplex mode only. A device with an active LACP waits passively for an invitation to create a LAG, or actively requests a peer to create LAG [74]. Some vendors include an option of building link aggregation which starts in a node and ends in two other nodes. Such aggregation is sometimes known as the Multi-Chassis Link Aggregation Group (MC-LAG). It is possible to create such link aggregation by using a proprietary protocol to ensure appropriate data transport, coordination, and delivery. From a single device point of view, the other end of the MC-LAG is usually seen as a single node, even though there are two distinct nodes. Several types of MC-LAG are used. Sometimes all links are in active mode and convey data; in other cases some links are in active mode, while others are used in case of failure of an active connection. The exact specification of MC-LAG operation is vendor specific and access to the specification is usually restricted. There is no standard for MC-LAG.

It should be noted that implementation of the LAG makes it possible to increase bandwidth on a link between a pair of directly connected nodes. It is a limited solution in comparison to the SPB or TRILL, since the mechanism is more multilink than multipath.

3.4. Network virtualization

Virtualization is considered to be an approach aimed at using physical resources for many purposes; resources are allocated using software solutions. Virtualization was originally introduced in computer science in order to allow many applications to use a single machine. Virtualization is a fundamental concept for grids and cloud computing. It has also recently been considered to be an important innovative technology for Next Generation Networks. There are many types of virtualization, depending on what kind of resources are "virtualized", i.e., split and used separately by the created substructures. As a result, it is possible to have virtualized hardware, memory, data, software, desktops and networks. In network virtualization, physical resources are split using subaddressing and dedicated connections to create isolated paths and, as a consequence, virtual networks.

Virtualization itself does not introduce multitopology routing, but enables the creation of many virtual topologies above the same physical resources and, as a further consequence, sending traffic over multiple paths.

A second generation of virtualization systems was oriented towards network resources, such as bandwidth (as well as output buffer or packet schedulers). For the splitting of node resources such as processing power, storage and RAM, there is dedicated software known as "virtual machine manager" or "hypervisor". Virtualization of network resources was accomplished in an intrinsically consistent way in ATM using VP-level bandwidth partitioning. The successor of ATM, the MPLS technology, also utilizes the concept of virtualization by running advanced routing and constraint-based routing (CBR).

Currently we are also observing the introduction of the virtual router concept, where physical routers are split into several virtual instances. Another emerging solution is transport virtualization which means running a virtual environment over protocol-agnostic transport.

3.4.1. Virtual network provisioning

The general drawback of virtual network provisioning is resource splitting into virtual networks (VN). This problem has received much attention in recent years. Virtual network provisioning creates a multi-topology environment and does not support multipath routing directly. Rather, it enables the running of different and separated paths over physical links. Splitting of physical resources (also known as "substrate resources") into virtual instances can be done in a flexible way, using formal optimization methods with chosen goal functions. As an example of a complete framework for virtual network provisioning we refer to [75,76], describing both theoretical and computational aspects of this process.

Network virtualization provides a way to run multiple architectures simultaneously on a single infrastructure enabling the sharing of a physical infrastructure between many virtual networks with varying characteristics. This approach also provides a clear separation of services and infrastructures [77]. Multiple challenges are associated with the deployment of network virtualization in an operator infrastructure.

As has already been mentioned, the allocation of physical resources can be optimized with regard to different chosen goal functions. Such functions mainly consider performance issues (e.g., minimizing link bandwidth occupancy, CPU computation power or transmission delay). Virtualization also supports energy-efficiency (which is in-line with the concept of green networking) since physical resources are used for many (virtual) instances, and can also be oriented towards additional targets, such as security (e.g., node reliability, link encryption).

3.4.2. Transport virtualization concept

The previously described concept of network virtualization (NV), based on sharing network resources, has recently been enhanced towards the concept of transport virtualization (TV) [78]. The difference between network and transport virtualization involves creating virtual resources in TV rather than just sharing resources, which was a fundamental concept for NV. The creation of resources in TV can be done in a flexible way, by combining multiple transport resources which can be of the physical or virtual type, and such resources can even belong to different providers. At first, physical or overlay resources are combined in order to constitute an abstract data transport resource, which is then split into virtual transport paths.

The concept of transport virtualization is further enhanced by a mechanism known as "concurrent multipath" (CMP) transmissions, which means using few paths simultaneously. CMP enhances resilience and increases throughput of transport; it also potentially introduces out-of-order packets and thus involves buffering at the destination. The buffer size required to deal with out-of-order packets can be dimensioned using an analytical model proposed, for example, in [78].

Virtualization of network and computational resources is a fundamental paradigm in the current evolution towards IT environments implementing joint concepts of intelligent, software-defined and flexible networking. Specific solutions such as grids, clouds and Software Defined Networks are the main targets. They provide improved resilience, efficiency (also energy efficiency), robustness and configuration flexibility in networks.

4. Transport layer solution

Path selection is naturally associated with layers 3 and 2 from the OSI/ISO model, therefore most multipath approaches operate in those layers. However, multipath transmission is also possible in layer 4. This may be difficult to imagine at first, as layers 4 and above have nothing to do with the path through which data is transported. Nevertheless, creating a multipath transmission is possible in layer 4, and Multipath TCP (MPTCP) [79] is the best approach for this.

MPTCP is an IETF version of the TCP protocol which supports concurrent transmissions. The idea is that one TCP session supervises two or more standard TCP sessions, thereby making it possible to use several paths for data transmission under the same TCP connection with all its benefits. MPTCP relies on the prerequisite that at least one of two devices has multiple logical interfaces with the network and therefore has more than one IP address.

Fig. 9 illustrates a typical scenario in which MPTCP can be useful. Two hosts, A and B, communicate with each other. Both hosts provide two disjoint connections to the internet, and thereby two IP addresses. As a result, they can be reached by connecting to each of those addresses. There are, therefore, up to four different paths between the hosts: $A1 \rightarrow B1$, $A1 \rightarrow B2$, $A2 \rightarrow B1$, and $A2 \rightarrow B2$. Although this scenario involves four possible paths, the MPTCP protocol can be used when at least two are available. The created paths are disjointed at least at the edges. The protocol cannot guarantee full disjointness; moreover the level of disjointness is unknown.

This scenario may sound artificial at first; however, it has applications in the mobile market and in wired networks. For example, smartphones can usually use two separate interfaces: 2G/3G/LTE and a WiFi connection. When

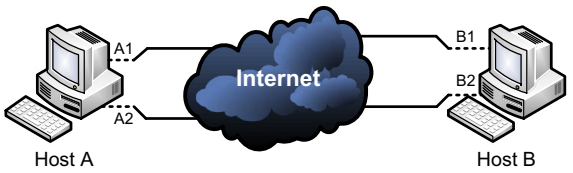


Fig. 9. Simple Multipath TCP usage scenario.

using MPTCP, smartphones benefit from increased throughput and improved resilience. Throughput is increased even for a single transmission, as one TCP stream can be separated into two subflows and sent via different paths, thereby utilizing the sum of the available bitrates. Resilience is improved by the fact that if one of the interfaces loses connection for whatever reason, the other can seamlessly take over. The connection is maintained, although additional throughput is lost. When the broken connection is reestablished it can be used by the protocol again, seamlessly to any transfer already in progress.

Fig. 10 shows the layered architecture of MPTCP. MPTCP supervises standard TCP sessions, known as "subflows", to provide the transport per path. The underlying network is unaware of this operation and treats each subflow as a regular TCP connection. The MPTCP-specific information is also carried by the standard TCP connection. This way, backwards compatibility is maintained. In order to manage the subflows, MPTCP implements the following functions:

- Path management: this is used to detect multiple possible paths between devices. It scans for the presence of multiple IP addresses and signals this information to the other side.
- Packet scheduling: this function divides the data received from the application into segments which are then transmitted on one of the available subflows. This function depends on the path management to find available paths. It is also responsible for re-ordering of packets received from different subflows.
- Subflow interface: a component that takes segments from the packet scheduler and transmits them over the specified path. To ensure delivery and to maintain compatibility, a standard TCP is used to maintain data in a subflow.
- Congestion control: this is used to coordinate congestion control functions of the standard TCP-managed subflows. When one subflow is suffering from congestion, instead of decreasing its bitrate, excessive data is transferred to another subflow.

The idea of MPTCP is simple and straightforward, although in reality there are difficulties. The authors of [80] expose certain problems and provide means of optimization to improve performance. As mentioned, MPTCP is particularly applicable in the mobile market [81], where multiple interfaces are commonplace. However, MPTCP is also used in wired applications, e.g., in large data-centers, where is it important to provide constant availability with the highest possible bitrate [82,83]: the two main benefits of using MPTCP.

5. Comparison and contrast

The paper presents ten approaches in which it is possible to provide multipath transmissions. Some were designed driven by the notion of multipath transmission, whereas in others this was in fact a byproduct. All the approaches are different in many ways; they were designed at different times and for different purposes. Moreover, they use various technologies and mechanisms.

The purpose of this survey is not to show advantages or disadvantages of using one technology or another to provide multipath transmissions, although such conclusions are easy to draw. The main goal is to show that although it is rarely used, multipath transmission can be achieved in many existing architectures. Furthermore, no additional actions are usually required to use multipath transmission.

In this section, we compare most of the presented mechanisms and point out their major differences. We omit the LAG from the comparison due to fact that this method only data to be sent through several links; as such, the solution is more multilink than multipath.

We start by showing the maturity of each solution, and examine whether it is well known and widely implemented, whether it is still at research stage or somewhere in between. Afterwards, we describe how the routing is realized and how the multiple paths are established, and where and how those paths can be chosen. Finally, we

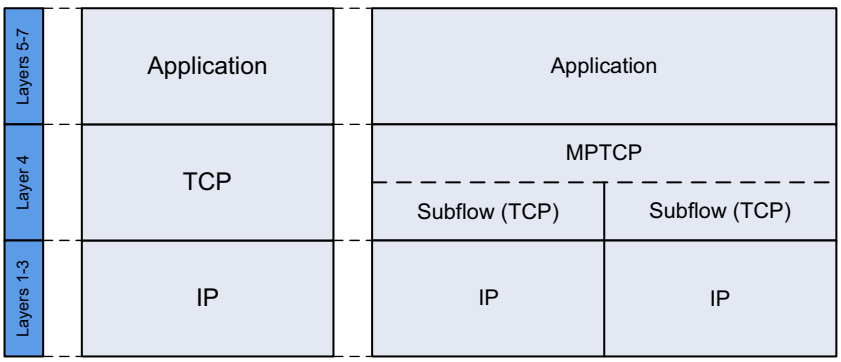


Fig. 10. Comparison of standard TCP (left) and MPTCP (right) protocol stacks.

define the requirements associated with each mechanism. We show how the signaling information is passed, and assess the overall mechanism complexity and the time scale in which paths are established.

5.1. Maturity of multipath mechanisms

All routing solutions described in previous sections are compared in Table 2. We analyze whether a particular method for packet routing is under research or whether it is implemented in devices that are available on the market. The source routing approach had been analyzed in depth in literature. Currently, this method for packet routing is available in some devices, although it is rarely used (mostly in small topologies). However, it is still of interest to some researchers. The hop-by-hop concept is widely used by several routing protocols. While this is a stable solution, research is currently ongoing. Although multi-topology routing was originally proposed several years ago and is now available, it is the subject of ongoing research. Bio-inspired and Valiant's routing are not currently available, and research interest in these solutions is moderate. The multipath in the MPLS solution has been the subject of analysis and development for several years, and a significant number of researchers are still working on this topic. The mechanism of packet routing in multipath MPLS is widely available and utilized. Software-Defined Networking is a hot research topic now. This technology is currently available in devices present on the market. FAMTAR is a new routing technology proposed in 2014. It is still under research and not yet available commercially. The SPB solution was standardized on 29 March 2012. So far, several vendors claim that the SPB standard is supported in their products. Similarly, several vendors support the TRILL solution. Network virtualization is also a hot topic in telecommunications. The method makes it possible to create virtual networks within a single architecture and to send selected traffic through fixed paths or based on a routing algorithm within a virtual network. MPTCP is still under research but first implementations started to appear. For example, it is included in Apple's IOS7, it is also available for Android and Linux-based systems.

5.2. Set-up features of multipath mechanisms

Table 3 provides a brief comparison of the proposed algorithms, taking into account routing type (strict or loose), path choice (ingress or core), and path setup (centralized or distributed). In strict routing, the signaling procedure is used which specifies the path, node by node, that must be visited by packets on the way to the destination node. This path does not need to follow the lowest

Table 2
Maturity of multipath mechanisms.

	Multipath in ...	Research	Market availability
Transport layer	MPTCP	Some interest	Available
Network/Internet layer	Source routing	Some interest	Widely available
	Hop-by-hop routing	Stable solution	Widely available
	Multi-topology routing	Some interest	Available
	Bio-inspired routing	Some interest	Not available
	Valiant's routing	Some interest	Not available
	MPLS	Constant interest	Widely available
	Software-Defined Networking	Hot topic	Available
	FAMTAR	New proposal	Not available
Data link layer	SPB	Some interest	Available
	TRILL	Some interest	Available
	Network virtualization	Hot topic	Available

Table 3
Comparison of multipath mechanisms.

		Routing type		Path choice		Path setup	
	Algorithm/Protocol	Strict	Loose	Ingress	Core	Central.	Distrib.
Transport Layer	MPTCP	[a]		X			X
	Source routing	X	X	X			X
Network/	Hop-by-hop routing	X			X		X
	Multi-topology routing		X	X		X	
layer	Bio-inspired routing		X		X		X
	Valiant's routing		X	X			X
	MPLS	X	X	X		X	X
	Software-Defined Networking	X	X			X	
	FAMTAR	X			X		X
Data link	SPB	X			X		X
	TRILL	X		X			X
	Network virtualization		X	X		X	

[a] Lower layers are responsible for routing.

cost path. In contrast, loose routing represents a procedure in which selected nodes must be visited by packets on their path to the destination. The shortest path is used between the specified loose nodes. In general, a path may be composed of segments which represent loose routing and the strict routing rule. The proposed categories cover long-term development of network solutions combining fundamental engineering concepts and their implementation due to the level of computational power, programming languages and networking.

The first column compares whether a decision about a routing path is strict or can be influenced by additional factors. Multi-topology routing, Bio-inspired routing and Valiant's routing are proposed for loose categories, supporting flexibility of decision-making within the network. On the other hand, in hop-by-hop routing and in FAMTAR the paths are strictly chosen based on information available when transmission of traffic begins.MPLS is the protocol that has both features, i.e., strict and loose routing (described in MPLS as loose and strict hops). Also in source routing and in Software-Defined Networking it is possible to make a strict or loose decision on routing. In source routing it is possible to define a whole path for packets or only its part. In Software-Defined Networking a routing decision depends on a central controller functionality.The column presenting path choice shows where the routing decision is elaborated in terms of the administrative organization of the network, domain or AS (at the border or within the network). The case of routing decisions taken at an ingress node is source routing, since in this case the decision is made at the origin of the route. Most protocols only use path choice in the ingress node in order to stabilize and avoid problems such as loops or excessive delays. Four examples of path choice made within the network are hop-by-hop (fundamental concept), FAMTAR, SPB, and bio-inspired routing. The third column presents how path setup is accomplished. In this category most solutions are distributed in nature, i.e. forwarding action is done locally, with available routing information. Again, MPLS is the only concept in which path setup is both centralized and distributed.

5.3. Complexity and operating features of multipath mechanisms

In the next step, we analyze all the presented mechanisms, taking into account requirements related to signaling, complexity of the overall mechanism, and the time scale in which paths are established. The described solutions are compared in Table 4. We start by stating whether an exchange of signaling traffic is required to provide multipath transmissions in each analyzed approach. In source routing, the required information is included and transported in headers of data packets. A similar situation is seen in MPTCP where option fields in standard TCP headers are used for signaling purposes. In hop-by-hop routing, FAMTAR and network virtualization solutions, no additional exchange of signaling information for multipath operation is necessary. In the first two cases, the forwarding table is already provided by routing protocols; in the last case, virtual networks are created – frequently manually – and traffic can be sent through statically-determined paths or through paths calculated by routing algorithms used within a virtual network. Additionally, dedicated signaling protocols are required for providing multipath in both MPLS and multi-topology routing. In the former case the RSVP-TE protocol is used, making it possible to establish MPLS LSPs, taking into account network constraint parameters such as available bandwidth and explicit hops. In the latter case the extensions to OSPF and IS–IS should be introduced to support multiple topologies. In turn, only minimal extension of the IS–IS protocol is necessary to provide a multipath mechanism in the SPB approach. Similarly, the TRILL needs some extensions. In Software-Defined Networks a signaling protocol is needed to exchange information between a controller and a device. One of the most popular protocols is OpenFlow [40]. A different approach is used in bio-inspired routing, where signaling information is exchanged through agents. In Valiant's routing, signaling is not required. A source node only needs to know all two-hop paths to a destination, and then it chooses one at random. Network virtualization can be done by any mechanism enabling separation of resources and, as a consequence, separation of transported

Table 4
Features of multipath mechanisms.

	Multipath in ...	Signaling	Mech. complexity	Time scale
Transport layer	MPTCP	in TCP headers (Options)	low	Short
Network/Internet layer	Source routing	In headers	Low	Short
	Hop-by-hop routing	Routing protocols	Low	Short
	Multi-topology routing	MT-OSPF, MT-IS–IS	Medium	Long
	Bio-inspired routing	Through agents	Medium to high	Short
	Valiant's routing	None	Low	Long
	MPLS	RSVP-TE	Medium	Long
	Software-Defined Networking	e.g. OpenFlow	Medium	Short
	FAMTAR	Routing protocols	Medium	Short
Data link layer	SPB	IS–IS	Low to medium	Short
	TRILL	IS–IS	Low to medium	Short
	Network virtualization	None dedicated	High	Long

traffic streams. As such, it is done by management rather than signaling. Additionally, it is justified observation that even if transport virtualization is considered, it is done by mechanisms implemented in nodes.

The overall complexity of multipath mechanisms is assessed in the next step. The proposed assessment is rather subjective. It takes into account both the complexity of algorithms for multipath computation and the complexity of processes/algorithms required for using the candidate paths. The reference is the hop-by-hop routing for which multipath processing is the native asset of routing protocols. As such, path computation and load balancing can be performed without any additional configuration, apart from the required routing configuration. While providing multipath solutions for SPB, TRILL, source routing and Valiant's routing is relatively simple, other approaches require additional operations. In the case of MPLS, an additional algorithm is used in the computation of constrained paths and the RSVP-TE protocol is used for managing the existing LSP paths and setting up new LSPs. Furthermore, it is necessary to configure separate routing instances for each topology on each node for the MTR solution to work. In FAMTAR, it is necessary to maintain a list for flows where the outgoing interfaces are written. The similar approach is usually used in Software-Defined Networks. However, in this case additional communication with the central controller is needed. The most complex mechanism is used for network virtualization, as it is decided at the management plane. Some optimization processes need to be launched to divide the available infrastructure resources into the requested virtual networks. The MPTCP technique presents yet another scenario. As MPTCP works in the transport layer, no additional functionality is required in the lower layers. Therefore, the complexity of the analyzed mechanism is rather low.

The next comparison criterion is time scale. By time scale we mean the estimated duration of established paths (expected duration for paths in the "on" state) in non-failure network conditions. It does not include the time required for path computation or setup. In the source and hop-by-hop routing approaches, as well as in MPTCP, the duration of the paths can be short, as new paths can be found and started very quickly if required. A similar explanation can be used for arguing that SPB or TRILL paths are rather short, as new paths are provided using the IS–IS routing protocol. In the case of MPLS networks, the set of proposed paths is usually provided after performing a complex optimization process. Taking this into account, the paths are expected (under normal conditions) to work for a relatively long time. The same arguments can be used for justifying the statement about the long duration of established paths in MTR, Valiant's routing and network virtualization solutions. In Software-Defined Networks and in FAMTAR the paths are setup for flows. As a result, the time scale is short.

6. Future of multipath routing

One of the most important applications of future networks seems to be in cloud computing. Another important feature is the ability to efficiently reduce consumed energy produced from fossil fuels. Both cloud computing and a reduction of consumed energy mean that networking infrastructures should be able to rapidly increase their capacity.

For example, in order to reduce the amount of consumed energy and to increase revenues, a network operator should be able to relocate the switching of data in transit nodes to somewhere with green energy, perhaps produced by a wind farm with strong wind, or to a location with low exterior temperatures. This way the amount of energy produced from fossil fuels or the total energy used to decrease the temperature inside rooms with networking equipment should be substantially reduced.

Multipath routing is a promising solution addressing challenges related to data centers and cloud services. They require data transfers between nodes, server clusters or data centers located in the same autonomous system or in different, distant domains. Data centers require fast and reliable transfers of large amounts of data. Cloud services also rely on data transfer over the network, between nodes operated by a single cloud provider as well as inter-cloud communication. Resources such as virtual machines, memory content and data sets. may need to be transferred, usually within a strict time regime. Currently, data center-based services as well as cloud services frequently face dynamic changes of user demand, such as behavior influenced by the users' interactions on social networks (e.g., flash-crowd demands for content or resources). As a result, establishing a cloud have a strong impact on the demanded capacity for links around servers (or data centers) serving a given cloud. It is worth noting that a cloud may be built for a short period, hence overprovisioning is not an economical solution. Additionally, emerging scenarios related to global service mobility requirements and the rapid growth of mobile services results in new challenges for cloud providers. In all the above sample cases, implementing the service may face a bottleneck such as bandwidth scarcity, congestion, violation of delay constraints, or service or network unavailability. Such bottlenecks can be avoided by using a multipath solution, which would help by:

- increasing bandwidth between two end points (bandwidths of paths are summarized) – enable rapid transfer of resources,
- choosing a path fulfilling specific QoS constraints (e.g., low delay path) – paths between a given endpoint pair used by various services may be selected according to QoS requirements,
- avoiding congestion on a single path (e.g., caused by rapid growth of traffic due to content popularity) – higher availability to users and improved Quality of Experience (QoE),
- increasing service availability in case of path failure (by having spare paths established in advance) – high service availability, facilitated cloud management, internal cloud operability maintained, better performance for end-users,

- quick distribution of popular content to more data centers, resulting in balancing server loads and spreading the traffic to different parts of a network, thus avoiding congestion – better QoE.

High QoS demands between a given pair of endpoints may occasionally be difficult to guarantee if just a single path is used. Adding a second path increases effective bandwidth between those nodes. If a single node is not capable of fulfilling all QoS requirements, they may be met if more nodes are involved, therefore a multipath algorithm is used. Because a portion of streams or packets is sent over a new path, the traffic on the first path decreases. Therefore, queues in nodes become shorter, which potentially results in a decreased delay between the endpoints. The traffic management may be realized in various ways. Decisions on sending traffic over a given path may be taken on a stream basis or packet by packet. In the first case, streams of various services may be distinguished and routed over paths best meeting their QoS requirements, or all streams may be treated equally and distributed over paths, either randomly or according to the ISP's traffic management policy. If a decision on path selection is taken packet by packet without recognizing streams, benefits still include increased bandwidth and potentially reduced average delay, although jitter may be increased. Additionally, if we consider explicit QoS reservations for some streams, it is also easier to meet their QoS demands and serve more streams with QoS guarantees. The QoS supporting capabilities of multipath routing methods may be used for more efficient implementation of Internet services, as well as cloud management and traffic optimization.

In summary, specific requirements coming from the application layer may be met by using multipath routing. This offers path customization to application performance requirements (also meant as QoS/QoE requirements), increased end-to-end reliability, and congestion avoidance.

One can imagine solutions where some paths will be activated on demand depending on the traffic load or quality requirements. Traffic between a pair of endpoints, generated by groups of services, can flow through different paths, increasing the capacity of the core network. The path can be established in advance and activated when needed, or created when a request for more network resources appears. Mobile operators may differentiate service quality requirements by establishing separate paths for different services. Each cell can be connected to the same data center by a few paths. A selected path can be used by a specific service. A few non-overlapping predefined paths between a cell and data center enable flexible usage of network resources.

The relative location of the content and mobile user may be important. By choosing the content in user proximity, network resources can be saved. Presently, many data centers are based on virtualization technologies (cloud computing). A handover performed by a mobile user may change the proximity relation between data center and user. The new cell may be closer to another data center, therefore handover may be an incentive for virtual machine migration. Services requiring fast virtual machine migration and fast data synchronization, such as gaming, may use dedicated, separate paths between the cell and a new data center. Normal traffic is transferred through other paths connecting the cell and data center.

The architecture of the Future Internet is related to the Internet of Things and the Internet of Services. Currently, the Internet is dominated by human–human interactions and human–machine interactions. In the future it is expected that machine–machine interactions will generate a notable part of all Internet traffic. The smart-city idea is a good example of interacting devices. It is very probable that multipath transmission will be used by communicating sensors and other devices. One can imagine a situation where transmission from different sensors located in the same network region may require different paths to the same server. It may happen that data generated by some of the mentioned sensors requires strict time scheduling, and that transfer must not be disturbed by other transfers on the way to the server. This can be achieved by establishing a separate path with appropriate parameters. It is expected that automatic procedures for path setup, controlled by sensors or other nodes, will be invented.

6.1. Challenges and requirements for optimal multipath architectures

Multi-topology functionality may be implemented in several ways. In this section, we present the main problems of solutions described in this paper, and present requirements for optimal multipath architectures. Our observations are summarized in Table 5.

- Scalability
  Scalability is an important issue in networks that grow rapidly. To solve this, network devices are usually upgraded or replaced by new ones. However, adding new hardware to solve the problem of inefficient scalability in an optimal network architecture is recommended.
- Manageability
  Manageability is an important feature in current networks because of the growing complexity of networks. The more sophisticated the network, the richer the set of services and functions. It also means a higher degree of virtualization and increased difficulties in effectively managing a network. It is not trivial to manage a network with a rich set of interactions between each component.
- Reliability
  In modern network architectures, the multipath concept enables transmission by using several paths at the same time, as well as offering efficient protection and restoration functionality. To ensure proper transmission in a network, even when failures occur, additional resources should be available. Moreover, the available resources should be used effectively.
- Cost
  The cost of network operation should be minimized in current and future networks. To deal with this problem, network operators should use network resources as effectively as possible. Moreover, they should implement devices with low operational costs.

Table 5
Challenges and requirements for optimal multipath architectures.

	Challenges to existing multipath architectures	Requirements for optimal multipath architectures
Scalability	To upgrade or replace existing hardware	To add more devices or components to increase computing power and capacity
Manage-ability	To react effectively to network or client needs	To manage a higher number of connections and interactions between network components
Reliability	To ensure co-existence of protection/ restoration mechanisms with multipath algorithms	To ensure sufficient network resources and efficient reliability mechanisms
Cost	To minimize cost of new devices	To ensure effective network architecture which delivers scalable bandwidth at a rational cost
Power consumption	To reduce power consumption by usually overprovisioned network resources	To ensure that power consumption is as low as possible and power is not consumed unnecessarily by excessive paths
Fault tolerance	To eliminate or reduce failures of network elements	Protection/restoration mechanisms ensure short breaks in transmission and high performance when failure occurs
Mobile network demands	To react to different user demands in order to avoid transmission problems related to bandwidth shortage and QoS	To selectively deliver specific services to chosen regions of a mobile network
Time synchronization	To synchronize in time network nodes and services	To make a distinction between time scheduling for different services and nodes in order to separately deliver a time reference signal for different services and nodes
Robustness	To cope with path heterogeneity, throughput fluctuations, and jitter	To deal with persistent reordering of data packets

- Power consumption
  Reduction of power consumption in networks is currently a hot topic. This issue is related to the previous one, and indicates that network operators should implement devices whose operation ensures a high efficiency in relation to consumed power.
- Fault tolerance
  This item is related to reliability. Each network element is a potential point of failure. It is recommended that network operators implement reliable hardware whose protection/restoration mechanisms operate efficiently.
- Mobile network demands
  Services offered to mobile users by mobile operators impose high bandwidth and QoS management requirements. The mobile core network must react to user demands, and connections between radio access nodes, core nodes and data sources (data centers, clouds) need to be managed in a very flexible way. Different traffic levels and different services may be delivered to the same localization following different paths. Many services are constructed in a distributed way (distributed application) and the resources comprising the service can be acquired from distinct locations; as a result, at times a few specific paths should be chosen for a particular service delivered to one cell.
- Time synchronization
  Many services and network operating nodes require time synchronization. This is very important in mobile networks. It is expected that the Internet of Things will impose strict conditions on time-scheduled applications and devices working in a synchronized mode. A few communication channels between devices may be required, some with specific synchronization and others without. This may be achieved by establishing multiple paths between devices with different time synchronization parameters.
- Robustness
  The requirement related to robustness states that multipath solution should be able to deal with persistent reordering of data packets caused by traffic fluctuations or changes in network capacity or topology. Moreover, the multipath solution should provide an aggregation benefit, e.g., a throughput improvement in comparison with a single path solution.

Other challenges of multipath transmission in wired networks are related to congestion control and traffic engineering. The authors of RFC 6077 [84] claim that efficient congestion control mechanisms for multipath transmissions should lead to significant benefits related to resilience and resource usage. However, it is necessary to understand interactions with network controlled routing schemas and traffic engineering when planning and developing congestion control schemas for multipath transmission. We should be aware that, when multipath transmission is available, an end node may divide its flows into subflows which may result in more efficient transmission of its traffic. On the other hand, this may also lead to problems with fairness in the network. The authors of [85] explain that fairness in congestion is one of the most important requirements which multipath congestion control mechanisms should meet.

Another goal which should be achieved when multipath is possible is resource pooling. A node which transmits its traffic through several paths should send as much traffic as possible by non-congested paths. It is also very important

to make a proper decision when dividing a flow into subflows should it be available and how to assign subflows to the paths and schedule their traffic in a fair regime. Interaction between the end systems and routing protocols and policies may also play an important role in multipath transmission. Usually, the end systems are not informed about all possible paths among nodes or even whether the multipath capability is available in a network.

As we can see, multipath transmission may result in more effective traffic assignment in a network. However, still many challenges are open and research and development work is necessary to propose new, efficient solutions.

7. Conclusion

Establishing multiple paths between network endpoints has obvious advantages. However, it is also associated with certain costs. We have shown that there are many ways of supporting multipath transmissions in IP networks and that it is an easy way of increasing throughput and resilience. As such, why are network operators so reluctant to use multipath transmissions, even though there are numerous established and mature options? It is not because they are not aware of them. Rather, it is due to the fact that operators want to be in full control of the traffic.

It is currently common practice that after optimal paths have been established by a routing protocol, operators supersede some of those paths by creating MPLS tunnels. This is manual traffic engineering. One of the reasons behind it is the need to reroute a portion of the traffic, thereby reducing congestion somewhere in the network. This would not be necessary if multipath transmissions were commonly used. Unfortunately, this is one of the network management operations that are predominantly still done manually.

In this paper we have shown that there are many options when it comes to providing multipath transmissions. We believe that in a situation where network resources are scarce, the operators will be more interested in employing multipath transmission, as this is the cheapest method of increasing network efficiency.

Acknowledgment

The research was carried out with the support of the project "High quality, reliable transmission in multilayer optical networks based on the Flow-Aware Networking concept" founded by the Polish National Science Centre under Project No. DEC-2011/01/D/ST7/03131.

References

[1] S. Adibi, S. Erfani, A multipath routing survey for mobile ad-hoc networks, in: 3rd IEEE Consumer Communications and Networking Conference, 2006, CCNC 2006, vol. 2, 2006, pp. 984–988.
[2] J. Al-Karaki, A. Kamal, Routing techniques in wireless sensor networks: a survey, IEEE Wireless Commun. 11 (6) (2004) 6–28.
[3] E. Alotaibi, B. Mukherjee, A survey on routing algorithms for wireless Ad-Hoc and mesh networks, Comput. Networks 56 (2) (2012) 940–965.
[4] R.G. Gallager, A minimum delay routing algorithm using distributed computation, IEEE Trans. Commun. 25 (1) (1977) 73–85.
[5] F.E. Heart, R.E. Kahn, S. Ornstein, W. Crowther, D.C. Walden, The interface message processor for the ARPA computer network, in: Proceedings of the May 5–7, 1970, Spring Joint Computer Conference, ACM, 1970, pp. 551–567.
[6] D.P. Bertsekas, E.M. Gafni, R.G. Gallager, Second derivative algorithms for minimum delay distributed routing in networks, IEEE Trans. Commun. 32 (8) (1984) 911–919.
[7] S. Murthy, J. Garcia-Luna-Aceves, Congestion-oriented shortest multipath routing, INFOCOM'96. Fifteenth Annual Joint Conference of the IEEE Computer Societies, Networking the Next Generation, Proceedings IEEE, vol. 3, IEEE, 1996, pp. 1028–1036.
[8] H.T. Kaur, S. Kalyanaraman, A. Weiss, S. Kanwar, A. Gandhi, BANANAS: an evolutionary framework for explicit and multipath routing in the Internet, ACM SIGCOMM Computer Communication Review, vol. 33, ACM, 2003, pp. 277–288.
[9] F. Devetak, J. Shin, T. Anjali, S. Kapoor, Minimizing path delay in multipath networks, in: IEEE International Conference on Communications (ICC), 2011, IEEE, 2011, pp. 1–5.
[10] J. He, J. Rexford, Toward internet-wide multipath routing, IEEE Network 22 (2) (2008) 16–21.
[11] S. Azodolmolky, M. Klinkowski, E. Marin, D. Careglio, J.S. Pareta, I. Tomkos, A survey on physical layer impairments aware routing and wavelength assignment algorithms in optical networks, Comput. Networks 53 (7) (2009) 926–944.
[12] D. Farinacci, V. Fuller, D. Meyer, D. Lewis, The Locator/ID Separation Protocol (LISP), IETF RFC 6830.
[13] R.J. Atkinson, S.N. Bhatti, Identifier-Locator Network Protocol (ILNP) Engineering Considerations, IETF RFC 6741.
[14] B. Fortz, M. Thorup, Internet traffic engineering by optimizing OSPF weights, in: Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies, IEEE INFOCOM 2000, vol. 2, 2000, pp. 519–528.
[15] J.J. Garcia-Lunes-Aceves, Loop-free routing using diffusing computations, IEEE/ACM Trans. Network. 1 (1) (1993) 130–141.
[16] A. Riedl, D.A. Schupke, Routing optimization in IP networks utilizing additive and concave link metrics, IEEE/ACM Trans. Network. 15 (5) (2007) 1136–1148.
[17] S. Murthy, J.J. Garcia-Luna-Aceves, Congestion-oriented shortest multipath routing, in: IEEE INFOCOM '96, vol. 3, 1996, pp. 1028–1036.
[18] M. Murtaza, E. Megan, F. Nick, V. Santosh, Path splicing, ACM SIGCOMM Comput. Commun. Rev. 38 (4) (2008) 27–38.
[19] D. Xu, M. Chiang, J. Rexford, Link-state routing with hop-by-hop forwarding can achieve optimal traffic engineering, in: IEEE INFOCOM 2008, vol. 18, 2008, pp. 466 –474.
[20] P. Psenak, S. Mirtorabi, A. Roy, L. Nguyen, P. Pillay-Esnault, Multi-Topology Routing in OSPF, IETF RFC 4915.
[21] T. Przygienda, N. Shen, N. Sheth, Multi Topology Routing in Intermediate System to Intermediate Systems, IETF RFC 5120.
[22] H.F. Wedde, M. Farooq, A comprehensive review of nature inspired routing algorithms for fixed telecommunication networks, J. Syst. Archit. 52 (8–9) (2006) 461–484.
[23] M. Abolhasan, T. Wysocki, E. Dutkiewicz, A review of routing protocols for mobile ad hoc networks, Ad Hoc Networks 2 (1) (2004) 1–22.
[24] G. Di Caro, M. Dorigo, AntNet: distributed stigmergetic control for communications networks, J. Artif. Intell. Res. 9 (1) (1998) 317–365.
[25] R. Schoonderwoerd, O. Holland, J. Bruten, L. Rothkrantz, Ant-Based Load Balancing in Telecommunications Networks, 1996.
[26] M. Dorigo, G. Di Caro, The Ant Colony Optimization meta-heuristic, New Ideas Optim. (1999) 11–32.
[27] K.M. Sim, W.H. Sun, Ant colony optimization for routing and load-balancing: survey and new directions, IEEE Trans. Syst. Man Cybernet. Part A: Syst. Humans 33 (5) (2003) 560–572.
[28] G. Pavani, H. Waldman, Routing and wavelength assignment with crankback re-routing extensions by means of ant colony optimization, IEEE J. Sel. Areas Commun. 28 (4) (2010) 532–541.
[29] Y.-M. Kim, E.-J. Lee, H.-S. Park, Ant colony optimization based self-organizing QoS framework in IP networks, IEEE Commun. Lett. 14 (11) (2010) 1074–1076.
[30] L. Carrillo, C. Guadall, J.L. Marzo, G. Di Caro, F. Ducatelle, L.M. Gambardella, Differentiated quality of service scheme based on the use of multi-classes of ant-like mobile agents, in: ACM Conference on Emerging Network Experiment and Technology, CoNEXT '05, New York, USA, 2005, pp. 234–235.
[31] P. Lučić, D. Teodorović, Computing with bees: attacking complex transportation engineering problems, Int. J. Artif. Intell. Tools 12 (3) (2003) 375–394.

[32] H. Wedde, M. Farooq, Y. Zhang, BeeHive: an efficient fault-tolerant routing algorithm inspired by honey bee behavior, in: Ant Colony Optimization and Swarm Intelligence, Lecture Notes in Computer Science, vol. 3172, 2004, pp. 83–94.
[33] M. Farooq, Bee-Inspired Protocol Engineering: From Nature to Networks, Springer, 2009.
[34] M. Munetomo, Y. Takai, Y. Sato, An adaptive network routing algorithm employing path genetic operators, in: Seventh International Conference on Genetic Algorithms, 1997, pp. 643–649.
[35] A.R.P. White, SynthECA: A Synthetic Ecology of Chemical Agents, Ph.D. thesis, Ottawa, Canada, 2000.
[36] L. Valiant, G. Brebner, Universal schemes for parallel communication, in: 13th Annual Symposium on Theory of Computing, Milwaukee, Wisconsin, USA, 1981.
[37] C.-S. Chang, D.-S. Lee, Y.-S. Jou, Load balanced Birkhoff-von Neumann switches, Part I: One-stage buffering, in: IEEE HPSR '01, Dallas, USA, 2001.
[38] I. Keslassy, S.-T. Chuang, K. Yu, D. Miller, M. Horowitz, O. Solgaard, N. McKeown, Scaling internet routers using optics, in: ACM SIGCOMM '03, Karlsruhe, Germany, 2003.
[39] R. Zhang-Shen, N. McKeown, Designing a predictable internet backbone network, in: HotNets, San Diego, USA, 2004.
[40] OpenFlow <www.openflow.org>.
[41] B. Heller, S. Seetharaman, P. Mahadevan, Y. Yiakoumis, P. Sharma, S. Banerjee, N. Mckeown, ElasticTree: saving energy in data center networks, in: 7th USENIX Conference on Networked Systems Design and Implementation, NSDI'10, 2010.
[42] E. Rosen, A. Viswanathan, R. Callon, Multiprotocol label switching architecture, IETF RFC 3031.
[43] D. Awduche, MPLS and traffic engineering in IP networks, IEEE Commun. Mag. 37 (12) (1999) 42–47.
[44] Y. Lee, Y. Seok, Y. Choi, C. Kim, A constrained multipath traffic engineering scheme for MPLS networks, in: IEEE International Conference on Communications, ICC 2002, vol. 4, 2002, pp. 2431–2436.
[45] S. Lahoud, G. Texier, L. Toutain, FATE: a polynomial time framework for flow allocation in MPLS-TE networks, in: The 14th IEEE Workshop on Local and Metropolitan Area Networks, 2005, LANMAN 2005, 2005, pp. pp. 1–6.
[46] E. Dinan, D. Awduche, B. Jabbari, Analytical framework for dynamic traffic partitioning in MPLS networks, in: IEEE International Conference on Communications, ICC 2000, vol. 3, 2000, pp. 1604–1608.
[47] C. Villamizar, MPLS Optimized Multipath (MPLS-OMP), Internet Draft.
[48] F. Ya-qin, W. Lin-zhu, An algorithm of static load balance based on topology for MPLS traffic engineering, in: WASE International Conference on Information Engineering, ICIE '09, vol. 2, 2009, pp. 26–28.
[49] S. Nelakuditi, Z.-L. Zhang, R. Tsang, Adaptive proportional routing: a localized QoS routing approach, in: Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies, IEEE INFOCOM 2000, vol. 3, 2000, pp. 1566–1575.
[50] A. Elwalid, C. Jin, S. Low, I. Widjaja, MATE: MPLS adaptive traffic engineering, in: Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies, IEEE INFOCOM 2001, vol. 3, 2001, pp. 1300–1309.
[51] X. He, H. Tang, M. Zhu, Q. Chu, Flow-level based adaptive load balancing in MPLS networks, in: Fourth International Conference on Communications and Networking in China, ChinaCOM 2009, 2009, pp. 1–6.
[52] B. Cui, Z. Yang, W. Ding, A paralled label switch paths traffic allocation algorithm based on minimum utilization of resource, J. Beijing Univ. Posts Telecommun. 28 (2) (2005).
[53] G. Yuan, Y. Chen, Y. Wei, S. Nie, A distributable traffic-based MPLS dynamic load balancing scheme, in: Asia-Pacific Conference on Communications, 2005.
[54] M. Menth, A. Reifert, J. Milbrandt, Self-protecting multipaths – a simple and resource-efficient protection switching mechanism for MPLS networks, Lect. Notes Comput. Sci. 3042 (2004) 526–537.
[55] E. Mannie, Generalized Multi-Protocol Label Switching (GMPLS) Architecture, IETF RFC 3945.
[56] A. Farrel, J.-P. Vasseur, J. Ash, A Path Computation Element (PCE)-Based Architecture, IETF RFC 4655.
[57] J. Vasseur, J.L. Roux, Path Computation Element (PCE) Communication Protocol (PCEP), IETF RFC 5440.
[58] D. Frost, S. Bryant, M. Bocci, MPLS Transport Profile Data Plane Architecture, IETF RFC 5960.
[59] S. Swallow, S. Bryant, L. Andersson, Avoiding Equal Cost Multipath Treatment in MPLS Networks, IETF RFC 4928.
[60] M. Jarschel, T. Zinner, T. Hoßfeld, P. Tran-Gia, W. Kellerer, Interfaces, attributes, and use cases: a compass for SDN, IEEE Commun. Mag. 52 (6) (2014) 210–217.
[61] S. Fang, Y. Yu, C.H. Foh, K.M.M. Aung, A loss-free multipathing solution for data center network using software-defined networking approach, in: APMRC, 2012 Digest, IEEE, 2012, pp. 1–8.
[62] S. Jain, A. Kumar, S. Mandal, J. Ong, L. Poutievski, A. Singh, S. Venkata, J. Wanderer, J. Zhou, M. Zhu, et al., B4: experience with a globally-deployed software defined W AN, in: Proceedings of the ACM SIGCOMM 2013 Conference, ACM, 2013, pp. 3–14.
[63] R. Wojcik, J. Domzal, Z. Dulinski, P. Gawlowicz, D. Kowalczyk, Performance evaluation of flow-aware multi-topology adaptive routing, in: IEEE CQR International Workshop, Tucson, USA, 2014.
[64] R. Wojcik, J. Domzal, Z. Dulinski, Flow-aware multi-topology adaptive routing, IEEE Commun. Lett. 18 (9) (2014) 1539–1542.
[65] J. Domzal, R. Wojcik, D. Kowalczyk, P. Gawlowicz, P. Jurkiewicz, A. Kamisinski, Admission Control in Flow-Aware Multi-Topology Adaptive Routing, in: ICNC 2015, Anaheim, USA, 2015 (in press) <http://www.kt.agh.edu.pl/~jdomzal/ICNC2015.pdf>.
[66] IEEE Standard 802.3-2012 – IEEE Standard for Ethernet, IEEE standard.
[67] IEEE 802.1aq-2012 – IEEE Standard for Local and Metropolitan Area Networks–Media Access Control (MAC) Bridges and Virtual Bridged Local Area Networks–Amendment 20: Shortest Path Bridging, IEEE standard.
[68] Information Technology – Telecommunications and Information Exchange Between Systems – Intermediate System to Intermediate System Intra-Domain Routeing Information Exchange Protocol for Use in Conjunction with the Protocol for Providing the Connectionless-Mode Network Service (ISO 8473), ISO/IEC 10589.
[69] D. Oran, OSI IS–IS Intra-Domain Routing Protocol, IETF RFC 1142.
[70] IEEE Standard for Local and metropolitan area networks – Media Access Control (MAC) Bridges and Virtual Bridged Local Area Networks – Amendment 22: Equal Cost Multiple Path (ECMP), 2014.
[71] R. Perlman, D. Eastlake-3rd, D. Dutt, S. Gai, A. Ghanwani, Routing Bridges (RBridges): Base Protocol Specification, IETF RFC 6325.
[72] J. Touch, R. Perlman, Transparent Interconnection of Lots of Links (TRILL): Problem and Applicability Statement, IETF RFC 5556.
[73] J. Carlson, D. Eastlake-3rd, PPP Transparent Interconnection of Lots of Links (TRILL) Protocol Control Protocol, IETF RFC 6361.
[74] IEEE 802.1AX-2008 – IEEE Standard for Local and Metropolitan Area Networks – Link Aggregation, IEEE standard.
[75] A. Fischer, J. Botero, M. Duelli, D. Schlosser, X. Hesselbach, H. de Meer, ALEVIN – a framework to develop, compare and analyze virtual network embedding algorithms, Electron. Commun. EEAST 37 (2) (2011) 1005–1015.
[76] X. Liu, Y. Chan, W. Xu, Stochastic programming methods used for network optimization, Perform. Eval. 63 (2) (2006) 1005–1015.
[77] L. Petersson, S. Shenker, J. Turner, Overcoming the internet impasse through virtualization, in: Hotnets2004, 2004.
[78] T. Zinner, K. Tutschku, A. Nakao, P. Tran-Gia, Using Concurrent Multipath Transmission for Transport Virtualization: Analyzing Path Selection, 2010, pp. 1–7.
[79] A. Ford, C. Raiciu, M. Handley, S. Barre, Architectural Guidelines for Multipath TCP Development, IETF RFC 6182.
[80] S. Barré, C. Paasch, O. Bonaventure, Multipath tcp: from theory to practice, in: 10th International IFIP TC 6 Conference on Networking, NETWORKING'11, vol. Part I, 2011, pp. 444–457.
[81] S.-C. Nguyen, T.-M.-T. Nguyen, G. Pujolle, S. Secci, Strategic evaluation of performance-cost trade-offs in a multipath tcp multihoming context, in: IEEE International Conference on Communications, ICC 2012, 2012, pp. 1443–1447.
[82] C. Raiciu, C. Pluntke, S. Barre, A. Greenhalgh, D. Wischik, M. Handley, Data center networking with multipath tcp, in: 9th ACM SIGCOMM Workshop on Hot Topics in Networks, Hotnets-IX, 2010.
[83] C. Raiciu, S. Barre, C. Pluntke, A. Greenhalgh, D. Wischik, M. Handley, Improving datacenter performance and robustness with multipath tcp, in: ACM SIGCOMM 2011 Conference, 2011, pp. 266–277.
[84] D. Papadimitriou, M. Welzl, M. Scharf, B. Briscoe, Open Research Issues in Internet Congestion Control, IRTF RFC 6077.
[85] C. Raiciu, M. Handley, D. Wischik, Coupled Congestion Control for Multipath Transport Protocols, IETF RFC 6356.

Jerzy Domżał received the M.S. and Ph.D. degrees in Telecommunications from AGH University of Science and Technology, Krakow, Poland in 2003 and 2009, respectively. Now, he is an Assistant Professor at Department of Telecommunications at AGH University of Science and Technology. He is especially interested in optical networks and services for future Internet. He is an author or co-author of many technical papers, four patent applications and one book. International trainings: Spain, Barcelona, Universitat Politecnica de Catalunya, April 2005; Spain, Madrid, Universidad Autónoma de Madrid, March 2009, Stanford University, USA, May–June 2012.

Zbigniew Duliński received the Ph.D. degree in theoretical physics from the Jagiellonian University. He works at Faculty of Physics, Astronomy, and Applied Computer Science at the Jagiellonian University. He previously worked in the area of theoretical and experimental elementary particle physics. For 10 years he has been working on problems in telecommunication. He is currently working on management mechanisms in overlay networks and inter cloud communication. His researched interests include distributed computing, network management mechanisms and traffic engineering.

Mirosław Kantor received his M.Sc. and Ph.D. degrees in Telecommunications from the AGH University of Science and Technology, Krakow, Poland in 2001 and 2010, respectively. Since 2001 he works at AGH-UST as an assistant professor at the Department of Telecommunications. His research interests focus on SDN, virtualization, cloud computing, Internet routing, inter-domain traffic optimization. He has actively participated in several European projects (LION, NOBEL, BONE, SmoothIT, Euro-NF) as well as grants supported by the Ministry of Science and Higher Education. He is the co-author of two books and over 30 research papers.

Jacek Rząsa received an M.Sc. degree in telecommunications in 2001. Since then he has been working in the Department of Telecommunications at AGH University of Science and Technology. He has participated in research ordered by telecommunication operators and worked in many international projects. He is author and co-author of several research papers. His research interests focus on energy aware optical transport networks, traffic engineering in optical networks and Carrier Ethernet.

Rafał Stankiewicz received the M.Sc. and Ph.D. degrees in Telecommunications from AGH University of Science and Technology, Krakow, Poland in 1999 and 2007, respectively. He is employed as at the Department of Telecommunications of AGH. His current research interests focuses on networking techniques, QoS provisioning methods, performance modeling and evaluation, traffic management and optimization at network and overlay/application layers (including cloud traffic management) and information security. He is an author of several conference and journal research papers and co-author of two books. He actively participated in European research FP4, FP5, FP6 and FP7 projects. He is TOGAF 9 Certified.

Krzysztof Wajda received his M.Sc. in Telecommunications in 1982 and Ph.D. in 1990, both from AGH University of Science and Technology, Krakow, Poland. In 1982 he joined AGH and is currently an assistant professor. He spent a year at Kyoto University and half year in CNET (France). He serves as a reviewer of a few journals: IEEE Communications Magazine, Telecommunications Systems, Computer Communications and international conferences. The main research interests include: traffic management, performance evaluation, network reliability, control plane, management systems, network services. He is the author (or coauthor) of 6 books and over 100 technical papers.

Robert Wójcik received his M.Sc. and Ph.D. (with honors) degrees in telecommunications from AGH University of Science and Technology, Kraków, Poland in 2006 and 2011, respectively. Currently, he works as a researcher at the Department of Telecommunications of AGH. He is the co-author of 5 international journal papers, 2 books, 1 book chapter, 3 patent applications and a number of conference papers. He has been involved in several international scientific projects, including: SmoothIT, NoE BONE and Euro-NF. His current research interests focus on Multipath routing, Flow-Aware Networking, Quality of Service and Network Neutrality.