Received July 5, 2019, accepted July 17, 2019, date of publication July 22, 2019, date of current version August 15, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2930229

Segment Routing Based Traffic Scheduling for the Software-Defined Airborne Backbone Network

KEFAN CHEN, SHANGHONG ZHAO, NA LV, WEITING GAO, XIANG WANG, AND XINQING ZOU

School of Information and Navigation, Air Force Engineering University, Xi'an 710077, China

Corresponding author: Kefan Chen (1148180199@qq.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61472443 and Grant 61701521, and in part by the Natural Science Basic Research Program of Shanxi under Grant 2018JQ6074.

ABSTRACT With the ubiquitous use of networking technologies in the military domain, network-centric warfare, which aims at improving the system of systems (SoS) capability, is a no-way-back trend of military development. Recently, it has become increasingly important to construct an airborne backbone network (ABN) to maintain coverage and provide reach-back to military units because the terrestrial networks cannot be flexibly deployed and have poor survivability. Given the advantages of software-defined networking (SDN), a software-defined airborne backbone network architecture (SD-ABN) is proposed for the ABN to ensure network flexibility, openness, interoperability, and evolvability. To meet the challenges of traffic management in SD-ABN, segment routing (SR) is applied with some practical modifications for the SD-ABN. Moreover, a network traffic scheduling algorithm termed MRP-TS is designed based on SD-ABN to improve the transmission reliability and bandwidth utilization by balancing network traffic to multiple reliable transmission paths. Relevant theories have been proposed and proved to illustrate the correctness of the algorithm. The network simulation experiments have been conducted using the network simulator EXata 5.1, and the simulation results validate the effectiveness of our proposals.

INDEX TERMS Airborne backbone network, software-defined networking, network architecture, segment routing, traffic scheduling.

I. INTRODUCTION

With the rapid development of information technology, network-centric warfare has become a no-way-back trend. The presence of a stable, high-capacity network structure has become a necessity in recent years owing to the rapid growth of net-centric applications and operations. However, the fixed communications infrastructure cannot guarantee its survivability on the battlefield, and its rigid network structure and configuration are difficult to adapt to the complex and changeable communications environment and demands. As a result, it has become increasingly important to extend this infrastructure dynamically in the absence of stable and wired communications facilities.

One of the off-the-shelf ways to build such a communications infrastructure is to rely on satellite communications (SATCOM). However, SATCOM is extremely limited in capacity and has deficiencies in security and survivability as well. Therefore, in recent years, there has been a significant push to rapidly deploy on-demand air resources such as blimps, unmanned aerial vehicles, and larger aircraft to construct an airborne backbone network (ABN) that can maintain coverage and provide reach-back to military aircraft as well as surface and space military units [1]–[3]. The military airborne networks (MAN) currently in use are built on discrete data link systems. One type of data link system is designed in a stacked and one-stop manner to satisfy specific tactical communications demands, leading to a vertically integrated network architecture with complex configuration processes, reduced flexibility and interoperability, and high upgrading difficulty [4]. If the future ABN is designed based on the traditional MAN architecture, the above issues will be inevitable. Therefore, there is a need to design a new ABN architecture to better facilitate future military communications.

On the other hand, when conducting NCW-based missions, different military units need to share information efficiently to form a strong systematic operational capability, requiring ABN to provide efficient communications for situational awareness, command and guidance, and cooperation of

The associate editor coordinating the review of this manuscript and approving it for publication was Chien-Fu Cheng.

various weapon and sensor systems. In this context, a large number of network flows with different quality of service (QoS) requirements should be transmitted with high quality, making scheduling the network traffic appropriately one of the core issues of ABN. As a critical component of traffic engineering, traffic scheduling (TS), whose main task is to find reasonable routing paths to guide network traffic to improve utilization of network resources, or better meet its QoS requirements [5], becomes an essential network application of ABN. Designing an efficient TS scheme for ABN is necessary to provide truly differentiated communications services to adapt to increasingly growing, uneven, and highly variable military communications patterns.

The issues of current MANs have many similarities with those of public networks. To solve the problems of flexibility, openness, interoperability, and evolvability of public networks, software-defined networking (SDN), a new network paradigm that might be able to address the deficiencies of current network infrastructures, has been proposed. SDN decouples control and data planes, logically centralizes the network intelligence and state in the controller, and abstracts the network infrastructure layer from the applications through well-defined northbound and southbound programming interfaces. SDN has been proven to simplify networking operations, optimize network management, and introduce innovation and flexibility [6], [7]. Attracted by the advantages of SDN, extending the SDN paradigm from the wired domain toward the wireless domain, such as 5G networks and vehicular ad hoc networks (VANETs) has recently received significant attention. Meanwhile, SDN is very friendly to traffic engineering; that is, SDN makes it practical to conduct fine-grained TS based on a global network view. For example, Google built a B4 network to connect its global data centers based on the SDN paradigm, and achieves nearly 100% link utilization with a low network management overhead [8], [9]. Given the advantages of SDN, it is attractive and promising to design future ABN architecture and the ABN TS scheme according to the SDN concept; however, this work has not yet been studied yet.

While the SDN paradigm generally makes it easier for the decision plane to have a global view of the network state and permits the network traffic to be transmitted as desired easily. Nevertheless, it requires the controller to maintain per-flow state in the network and program network elements along the flow path, which causes some new problems [10]: (1) poor scaling as the number of flows and network size grows; (2) increased network convergence time and relatively low success rate of network updates because of too many interactions. Compared with terrestrial networks that have stable and high-quality wired links, ABN has poorer and more changeable link qualities because of the factors such as changing topology, airframe occlusion, doppler effect, and malicious jamming, all of which bringing management challenges for network traffic under the SDN paradigm.

Segment routing (SR) is a source routing-based technique that allows a source node to specify a path as an ordered list of segments to guide a packet across the network. In SR, the header has sufficient information to steer the packets from the ingress node to the egress node of the path. The combination of SDN and SR can significantly alleviate the above problems as the states and forwarding rules of flows are only maintained and programmed on the ingress node [10], [11]. Therefore, the use of SR in SDN gives ABN more flexibility and makes schedule the network traffic more concise. Nevertheless, SR was initially designed for wired networks. Thus, some new designs and modifications are needed to make SR available for ABN. Apart from the traffic management mechanism, the TS strategy (or routing strategy) is also an essential part of TS. Because the changing link quality is one of the prominent features of ABN, how to ensure reliable transmission is a vital issue that should be considered by a TS strategy for ABN. Moreover, improving the real-time performance of forwarding network traffic is another important target of the TS strategy. ABN undertakes the task of forwarding all kinds of traffic on the battlefield, resulting in the traffic transmission demand is much higher than that of ordinary MANs. Owing to the limited number of transmission hops of ABN (usually smaller than 5), network hop count is not a critical factor that affects the transmission real-time performance. Instead, if the bandwidth resources of ABN are not used well, network congestion will occur, which will severely reduce the real-time performance, and even cause queue cache overflow and affect the transmission reliability. In general, the TS strategy of ABN is required to consider both the reliability and bandwidth utilization metrics.

In this study, we investigate the SR-based TS for SDN enabled ABN. The main contributions of this paper are as follows:

- A network architecture, the software-defined airborne backbone network (SD-ABN), is proposed to integrate the SDN paradigm into ABN, aiming to promote the migration of the SDN advantages to ABN.
- The SR technology is implemented in SD-ABN to improve traffic management efficiency.
- A TS algorithm, which comprehensively considers reliability and bandwidth utilization metrics, is designed for SD-ABN to improve the reliability and real-time performance of forwarding network flows.
- Relevant theories are proposed and proved to illustrate the correctness of the proposed algorithms; extensive experiments were conducted to validate our proposal.

The rest of this paper is organized as follows. We review the related work in Section II and specify the SD-ABN architecture in Section III. We then describe the implementation ideas of the SR technology for SD-ABN in Section IV and propose an SR-based traffic scheduling strategy in Section V. Simulation validation is presented in Section VI. And Section VII concludes this research.

II. RELATED WORKS

MANs currently in use are built on discrete data link systems, such as Link-11, Link-16, TTNT, and MADL. One type of data link system is designed in a stacked and one-stop manner to satisfy specific tactical communications demands. Existing MANs have been able to provide necessary communications for command and guidance, situational awareness, and primary tactical coordination [4]. However, the long-term development of chimney-style has formed the vertically integrated network architecture, leading to a lack of flexibility, interoperability, and evolvability, and also hindering the improvement of network performance. Restricted by the applications demand, the development speed of MANs is relatively slow. Nevertheless, deploying network nodes in the air has become an important scheme of improving the communications capabilities of public networks owing to the impetus of enormous communications demands and ubiquitous use of unmanned aerial vehicles (UAVs). Some research works can be used in both the military and civilian domains. Naturally, as a promising network paradigm, SDN has been a recent research focus area within the aviation field.

References [15] and [16] point out that the SDN paradigm provides an opportunity to control UAV networks programmatically and makes it easier to configure and manage UAV networks. In [17], an SDN-based prototype system, which uses the knowledge about aerial node orbits, is designed to predict future network events, especially link outages, which increases the availability and performance of aerial networks. Zacarias et al. investigated the combination of SDN with DTN approaches in UAV relay networks and proposed an SDNs-DTN network architecture [18]. Seniti et al. [19] proposed an SD-UAV network architecture and a routing algorithm for SD-UAV, and the simulation shows that the proposed framework performs well in terms of an end-to-end outage with a moderate reduction in end-to-end delay. Based on SDN, Zhang et al. [20] have proposed an SSAGV network architecture to exploit the advantages of space, air, and ground segments, to support diverse vehicular services in various scenarios efficiently and cost-effectively. Zhou et al. [21] have proposed an air-ground integrated mobile edge computing (MEC) framework that is based on SDN by exploiting the benefits of high mobility and flexible computing resource allocation of UAVs and vehicles. A simple comparison of existing airborne network architectures is shown in Table 1.

The use of the SDN paradigm indeed improves the flexibility, interoperability, openness, and evolvability of airborne networks. Nevertheless, the network robustness is not fully considered in current research works, which is critical for military applications. By merely adopting the centralized network control mode, the advantages of distributed network control such as flexibility and robustness are also sacrificed. Applying the SDN concept in airborne networks is developing and has a good prospect. However, most of current research is for civilian use and focuses on the design of network architecture and on UAV platforms. Comparatively speaking, the service object, network structure, communications mode, and communications environment of ABN are quite different, making the proposed network architectures cannot be well applied to ABN. However, these architectures provide useful references for the design of SD-ABN. Moreover, there is little research on the TS for the airborne network designed based on the SDN paradigm. Although [19] have proposed a TS scheme for SD-UAV, they considered the omnidirectional communications mode with the aim of reducing the interferences between different UAVs; however, this is not applicable for ABN that mainly employ the directional communications.

III. THE SD-ABN ARCHITECTURE

A. DESCRIPTION OF SD-ABN

Before describing SD-ABN, some assumptions are given for better illustrating our design:

1) SD-ABN is a newly designed network architecture that does not consider the co-existence with its legacy network as ABN has not been deployed yet.

2) The wireless control channel between the control plane and data plane of SD-ABN is constructed based on omnidirectional communications, while the transmission within data plane adopts directional communications mode.

3) The airborne backbone network nodes have sufficient load.

As shown in Fig. 1, SD-ABN is logically divided into an application plane, a control plane, and a data plane, because SD-ABN needs to follow the basic network architecture of SDN, which is also divided into the above three planes. In the application plane, a variety of network management applications can be easily designed and deployed. The requests from a network application can be translated into rules by the northbound interface (NBI) to the control plane, which are further interpreted into instructions to dictate the data plane behaviors via the southbound interface (SBI).

The SD-ABN control plane provides an abstract global network view and a programmable interface for the application plane to help network applications to make and translate network control decisions. Meanwhile, the SD-ABN control plane translates the instructions of the application plane into specific network configuration rules and deploys the rules by configuring the SD-ABN data plane via SBI. Unlike the traditional SDN architecture, the SD-ABN control plane is divided into an ABN control subplane and a platform control subplane. The information interaction rules between the two control subplanes are stipulated by the control plane interface (CPI). The ABN controller works for the ABN control subplane and manages network behaviors in a centralized manner. As a vital communications infrastructure on the battlefield, ABN has a high robustness requirement, which is contradictory to the centralized network control feature of SDN. Therefore, in the ABN control subplane, SD-ABN uses a multi-controller structure to make multiple controllers jointly control the network and provide back up for each other. Moreover, the ABN controller is onboard aircraft keeping in the safe place of the battlefield such as airborne warning aircraft, large electronic reconnaissance aircraft to improve its

TABLE 1. Comparison of different airborne network architectures.

Architecture		Merit	Limit
Architectures of deployed military airborne networks			
The airborne network builds on Link-16 (for example)		• Simple structure • Strong robustness	• Complex and fixed network planning • Difficult to upgrade • Closed system
The airborne network builds on TTNT (for example)		• Simple structure • Strong robustness • Flexible information sharing	• Hard to conduct online optimization • Difficult to upgrade • Closed system
Airborne network architectures proposed in researches			
~~SDN~~	A. Sen, *et al.* [12]	• More optimal deployment of airborne network nodes	• Only network topology is optimized • The vertically integrated network architecture is neglected
	J. P. Rohrer, *et al.* [13]	• Improve network performance by a cross-layer design of the TCP/IP protocol stack • Strong robustness	• Insufficient scalability • Hard to conduct online optimization • The vertically integrated network architecture is not totally changed
	H. Ahmadi, *et al.* [14]	• Enhance network capacity and coverage • Strong robustness	• Hard to conduct online optimization • The vertically integrated network architecture is neglected • Unable to provide communications for wide-area battlefield
SDN	H. Iqbal, *et al.* [17]	• Enhance network route selection • Flexible network configuration • Good interoperability and evolvability	• Poor robustness in battlefield • Advantages of the distributed network control are disabled • Heterogeneous networks are neglected
	I. Zacarias, *et al.* [18]	• Support the diverse range of strict requirements for applications in the last-mile tactical edge network • Flexible network configuration • Good interoperability and evolvability.	• Insufficient robustness in battlefield • Advantages of the distributed network control are disabled • Unable to provide communications for wide-area battlefield
	G. Secinti, *et al.* [19]	• Better integration of heterogeneous communications technologies • Flexible network configuration • Good interoperability and evolvability.	• Insufficient robustness in battlefield • Advantages of the distributed network control are disabled • Unable to provide communications for wide-area battlefield
	N. Zhang, *et al.* [20]	• Seamless integration of space-air-ground networks • Flexible network configuration • Good interoperability and evolvability • Efficient allocation of communications resources	• Insufficient robustness in battlefield • Advantages of the distributed network control are disabled • Need the support of fixed infrastructures
	Z Zhou, *et al.* [21]	• Support numerous IoT use cases • Flexible network configuration • Good interoperability and evolvability	• Insufficient robustness in battlefield • Advantages of the distributed network control are disabled • Need the support of fixed infrastructures • Unable to provide communications for wide-area battlefield

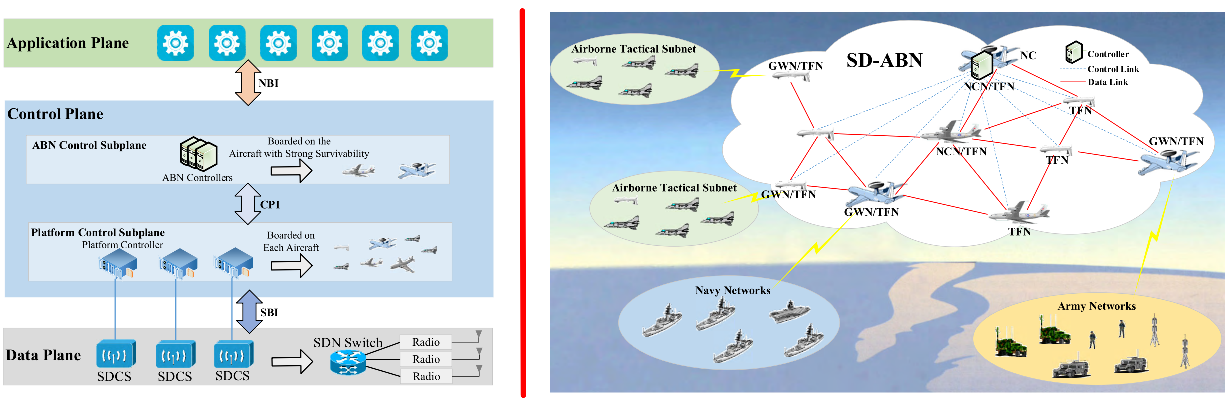


FIGURE 1. The SD-ABN architecture.

survivability. The platform control subplane is composed of the platform controllers, and each aircraft has one platform controller. On the one hand, the platform controller acts as a proxy for the ABN controller to control the data plane, which can simplify controller design and reduce network control overhead. On the other hand, a platform controller has independent network control logic, giving SD-ABN distributed network control capabilities for when the centralized network control mode is invalid or inefficient. The definition of platform control subplane is still in accord with the SDN principle as the control plane and data plane remain separated.

The SD-ABN data plane is in charge of processing and forwarding network traffic and is composed of the software-defined communication system (SDCS) boarded on each aircraft. With the help of heterogeneous radios, SDCS can simultaneously provide communications for multiple heterogeneous networks. SDCS is programmable, and its behaviors are managed by controllers instead of integrated network control logic. SDCS uses an SDN switch to perform multihop routing and has a common radio-to-router interface (R2RI) to connect different radios [22]. This system architecture can improve network interoperability and routing performance by separating the radio capability (one RF hop) from the router functionality (multihop). Moreover, this separation distinguishes the programmability of the network layer and that of the lower layers (link layer and physical layer) to facilitate more fine-grained programmability. As shown in Fig. 2, a radio and a switch interface of SDCS are allocated to construct SD-ABN. The radio uses a phased array antenna to generate multiple directional beams. Antenna elements corresponding to one generated directional beam are mapped to one transceiver through the radio distribution network (RDN), which forms an independent transmission channel. After configuring frequencies that do not interfere with each other for different transmission channels and aircraft, a pseudo-wired network can be formed, allowing an ABN node to communicate with its multiple neighbors simultaneously.

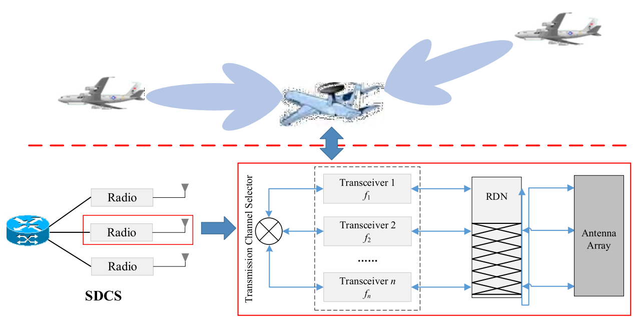


FIGURE 2. The SDCS structure.

From the perspective of network composition, SD-ABN is composed of network control node (NCN), gateway node (GWN), and traffic forwarding node (TFN). NCN is in charge of carrying the ABN controller. There may be more than one NCN in SD-ABN because multiple ABN controllers may be onboard different aircraft. GWN is responsible for connecting heterogeneous networks and different military units to ABN through its SDCS, which has the functions such as message format conversion, link selection, and power control. TFN is in charge of forwarding local and passed network flows, and all nodes in SD-ABN are TFNs.

B. ADVANTAGES OF SD-ABN

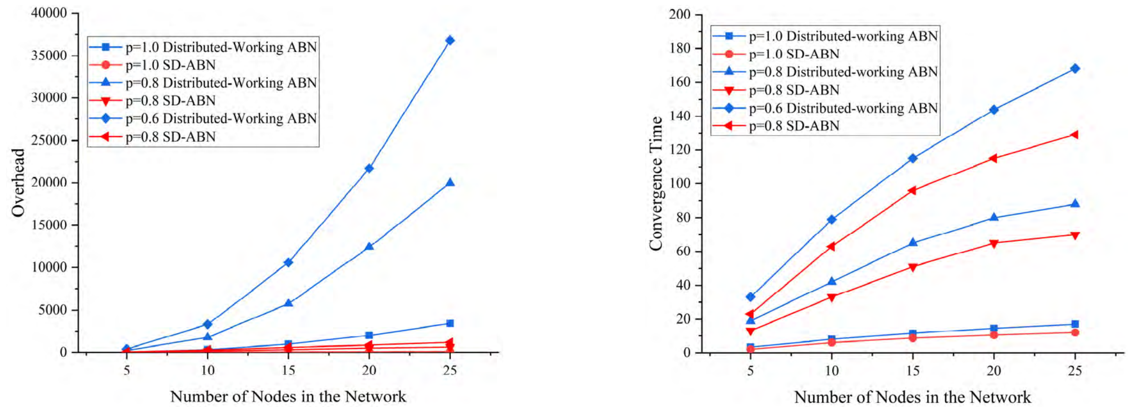
The behaviors and performances of current MANs rely on manual configurations performed on the ground, and configuring network devices require highly specialized and proprietary tools that merit specialized training for each type of device. As a result, the behavior of current MANs cannot generally be dictated while the network is running, which leads to reduced flexibility and adaptability. Benefitting from the distinctly separated control and data planes of SDN, the ABN control plane can dictate network behaviors in a timely and flexible manner via SBI (e.g., OpenFlow, ForCES, OpFlex). Moreover, many user-friendly network control applications can be easily developed and installed for SD-ABN to satisfy diverse ABN management demands based on NBI, which enables the customized network control capability.

Military units need to dynamically build diverse coalition relationships that require dedicated QoS guarantees of communications. A distributed network paradigm has always been an attractive development direction for military networks considering the invulnerability requirements. Nevertheless, for a fully distributed ABN, complicated negotiations are inevitable concerning complex coalition relationships with varied communications demands when allocating network resources, leading to enormous network control overhead and making it difficult to converge the expected point of the network state. The SDN paradigm allows for the centralized network control, which enables scheduling of the ABN resources with a global network view and lower network control overhead, and thus avoids local optima and accelerates the convergence rate of the network state (if there are enough computing resources allocated to the controller). For instance, if each ABN node has a status message that must be obtained by the other nodes to form a consistent network view for network optimization. Based on the SDCS structure and the shortest-path-forwarding principle, the overhead (presented by the number of interactions within the network) and the convergence time (time needed for all nodes to form a consistent network view) for the distributed-working ABN are shown in (1) and (2) respectively (assuming that the message cannot be fused, different nodes can simultaneously transmit messages, and the network can be modeled as a connected graph):

$$overhead_D = \sum_{v_i, v_j \in V, v_i \neq v_j} \left[\frac{1}{p^{\min d(v_i, v_j)}}\right]_r \times \min d(v_i, v_j) \quad (1)$$

$$convergence\_time_D = \max_{v_i, v_j \in V, v_i \neq v_j} \left(\min d\left(v_i, v_j\right)\right) \quad (2)$$

where v denotes the set of nodes in ABN, $\min d\left(v_i, v_j\right)$ represents the minimum hop count between v_i and v_j, p represents the success probability of one transmission, and $[]_r$ denotes the right rounding. For SD-ABN, a node can send its status message to a control node, and the overhead and convergence time can be represented by (3) and (4) respectively,



(a) (b)

FIGURE 3. **Overhead and convergence time comparison between SD-ABN and the distributed-working ABN. (a) Network control overhead. (b) Network convergence time.**

where v_c denotes the control node.

$$overhead_C = \sum_{v_i, v_c \in V, v_i \neq v_c} [\frac{1}{p^{\min d(v_i, v_c)}}]_r \times \min d(v_i, v_c) \quad (3)$$

$$convergence\_time_C = \max_{v_i, v_j \in V, v_i \neq v_c} (\min d\,(v_i, v_c)) \quad (4)$$

According to the above example, Fig. 3 shows the numerical simulation results. The dimension of the simulation scenario was set to 600km 600km 10km, and the node transmission range was set to 200km. According to the simulation results, SD-ABN has much lower overhead and convergence time than the distributed-working ABN.

To satisfy emerging communications demands, massive software and hardware upgrades are inevitable for current MANs because the network control and data planes are integrated into distributed network devices, which leads to a long upgrade period with a high economic cost, and hinders the generation of new communications capabilities. If the SD-ABN architecture is employed with new hardware technologies, such as hardware virtualization and integrated RF, new network services can be deployed to ABN by easily upgrading controller components and network applications, which leads to much lower costs regarding both time and expenditure. Moreover, with the decoupling of software and hardware, SDN facilitates the use of common hardware and simplifies the work of orchestrating software functions. Therefore, with the SD-ABN architecture, the interoperability of ABN can be significantly improved, and the advantages of various network technologies can be well utilized.

Presently, new technologies such as cloud computing, big data, and artificial intelligence have been applied gradually in the military field, and these emerging technologies play increasingly important roles on the NCW-based battlefield. SDN is a promising network paradigm for the integration of networks, cloud computing, big data, and artificial intelligence, and it can significantly benefit both network users and the network itself. SD-ABN separates network control logic from widely distributed devices, making it convenient to utilize the computing resources provided by cloud computing infrastructures. The SDN controller can centralize the network state information of SD-ABN in a timely and low-cost manner, which facilitates the comprehensive utilization of network state information and the other types of information (e.g., battlefield situation information, mission planning information) with the help of big data technologies. Benefitting from abundant computing resources and rich information, artificial intelligence can be used to generate appropriate network control policies and mission plans.

IV. ENABLING SR-BASED TRAFFIC SCHEDULING

The SR architecture is mainly divided into a control plane and a data plane. The SR control plane is responsible for distributing labels and maintaining routing information of the whole network. The SR data plane is concerned with defining the procedure for encoding a sequence of segments (instructions) on a packet and processing the segments list for forwarding the packet. SR has been widely used at present, with sufficient technical references. Therefore, the implementation of SR in SD-ABN can refer to existing SR instances, and further modify the SD-ABN data plane, control plane, and related interfaces accordingly. We implemented the SR data plane of SD-ABN based on MPLS in terms of compatibility [23]. The platform control subplane of SD-ABN enables distributed network management capability for SD-ABN, laying the foundation for the SR control plane implementation. As the open shortest path first protocol (OSPF) is one of the alternative routing protocols for future ABN and has been modified to support SR [24], we implemented OSPF to play the role of the SR control plane in SD-ABN. As SD-ABN has a MPLS-based data plane, correspondingly, the PCE-PCC (PCE is the path computing element and PCC is the path computing client) [25] network traffic management paradigm has been employed to schedule the network traffic.

A data plane device of SD-ABN is composed of an SDN switch (responsible for routing network traffic) and a transmission radio matched to a specific interface of the SDN switch (responsible for accessing wireless channel), while the SR control plane needs to be implemented by the platform controller. Therefore, the difficulty of deploying SR in SD-ABN is that a coordination and communication mechanism among the above three independent devices must be designed to support the SR functions, which needs not to be considered in general SDN networks. Therefore, as shown in Fig. 4, we have designed a segment routing extension kit (SREK-ABN) to enable SR-based traffic scheduling by establishing the SR-related coordination and communication mechanism among the three devices. SREK-ABN consists of three modules working on the platform controller (SREK-ABN-C), the SDN switch (SREK-ABN-S), and the transmission radio (SREK-ABN-R), respectively.

FIGURE 4. Overview of SREK-ABN.

One of the critical functions of SREK-ABN is that it integrates OSPF to distribute SR information and maintain routing information. SREK-ABN relies on SREK-ABN-C to implement OSPF, and SREK-ABN-C has the following functions: (1) implementing the OSPF logic, (2) maintaining the routing information base (RIB), (3) updating the forwarding information base (FIB) in SDN switch when RIB is updated, and (4) generating and processing OSPF messages. As shown in Fig. 5, the OSPF messages within SDCS are transmitted through the transmission channel established by SREK-ABN-S and SREK-ABN-R. SREK-ABN-R sends the OSPF messages received from the wireless channel to the SDN switch or sends the OSPF messages from the SDN switch to the wireless channel. SREK-ABN-S sends the OSPF messages from the radio to the platform controller via SBI-P or sends the OSPF messages from the platform controller to the radio for accessing wireless channel. The information exchange between SREK-ABN-S and SREK-ABN-R is regulated by R2RI. The dynamic link exchange protocol (DLEP)

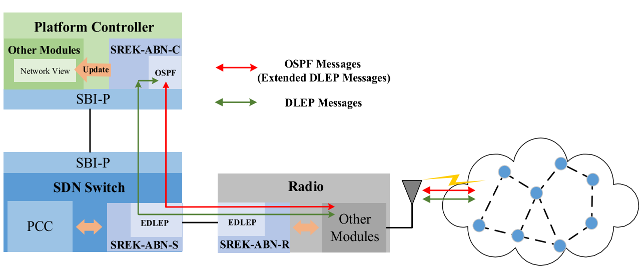


FIGURE 5. SREK-ABN processes OSPF messages and DLEP messages.

is an R2RI defined in RFC8875, which has many advantages compared with other R2RIs, such as lower overhead, simpler implementation, and better standardization support [22]. Therefore, we extend DLEP (EDLEP) to enable the extended DLEP messages, such as OSPF messages and PCEP messages (described below) to be transmitted between SREK-ABN-S and SREK-ABN-R. As shown in Fig. 5, because DLEP can report the link state information to the SDN switch, SREK-ABN-S also sends the link state information obtained via DLEP messages to SREK-ABN-C via SBI-P to enhance the link state awareness of OSPF. Based on the network information obtained by OSPF, SREK-ABN-C updates the network view of the platform controller in a timely manner.

The ABN controller perceives the global network view using a similar approach with BGP-LS [26] as OSPF provides the necessary means to perceive the network status information needed for traffic scheduling. For the control node, its SREK-ABN-C periodically synchronizes the network view of its platform controller into its ABN controller through the avionics network. The PCE in the ABN controller manages network flows using the path computation element communication protocol (PCEP) that has been extended to support SR [27]. As shown in Fig. 6, because the platform controller acts as a proxy of the ABN controller, all messages transmitted between SDCS and the ABN controller need to be translated by the platform controller. A flow forwarding requirement of SDCS is first transmitted to the platform controller through SBI-P, and then corresponding PCEP messages are generated by SREK-ABN-C and are sent to the SDN switch. SREK-ABN-S will send the received PCEP messages to the radio using EDLEP to access wireless channel. Finally, the PCEP messages are forwarded to the control node. The instructions issued by the ABN controller are configured in the SDN switch following the reversed direction.

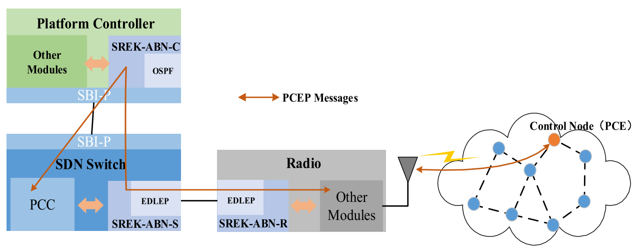


FIGURE 6. SREK-ABN processes PCEP messages.

Here, we have introduced the basic idea of our SR implementation in SD-ABN, especially the solution to the special problem of SR extension in SD-ABN. With SREK-ABN, SR can be deployed in SD-ABN by simply extending the functions of the platform controller, the ABN controller, SDCS, and related interfaces according to the SR technical requirements. The latter work is not specifically described in this paper as the standardization work for SD-ABN has not been conducted yet, and our work is more focused on functional implementation instead of a benchmark. Therefore, it is recommended to deploy SR in SD-ABN referring to but not limited to our work.

V. TRAFFIC SCHEDULING STRATEGY FOR SD-ABN

In this section, we propose a multi-reliable-paths based traffic scheduling algorithm (MRP-TS) to improve the transmission reliability, reduce the probability of congestion, and provide differentiated traffic forwarding services. MRP-TS decomposes a network flow into multiple transmission paths in the premise of satisfying the reliability requirements of the flow to improve network bandwidth utilization while ensuring transmission reliability. Meanwhile, to preferentially guarantee the transmission quality of high-priority flows, MRP-TS allows the bandwidth of low-priority flows to be preempted by high-priority flows, and MRP-TS also alleviates the adverse effect of bandwidth preemption by traffic decomposition. Particularly, the following assumptions are necessary for our design:

1) Each node in the network can generate multiple directional beams as needed to establish multiple point-to-point transmission links with other nodes.

2) The controller can efficiently generate a global network view.

3) The controller can obtain the basic information of all network flows, such as bandwidth need, reliability requirement, and priority.

4) Reasonable frequency planning has made all point-to-point transmission links non-interference with each other.

5) The link reliability is bidirectional equivalent.

A. THE MRP-TS ALGORITHM

For the ease of algorithm design, SD-ABN is modeled as an undirected graph $G = (V, E)$, where V denotes the set of nodes in the network, and E denotes the set of edges representing the wireless links. To find the most reliable transmission path, we define the following metric:

$$\max\left(\prod_{e\in P_{select}, P_{select}\in \mathrm{P_a}} r_e\right) \tag{5}$$

r_e denotes the link stability of link e, which is measured by the probability of correctly receiving a packet through link e. Each node in SD-ABN interacts with its neighbors periodically with hello messages to learn the network topology and r_e, and the learn results will be sent to the control node. $\mathrm{P_a}$ represents all available paths from the source node to the destination node. The above metric means that the greater the product of the stability values of all the links on a path, the higher transmission reliability can be achieved.

If there is only one available path p between a pair of nodes u and v, the transmission reliability between u and v can be expressed as follow:

$$R_{u,v} = \prod_{e\in P_{select}} r_e \tag{6}$$

If there are multiple edge-disjoin paths between u and v, the transmission reliability between the two nodes can be expressed as follow:

$$R_{u,v} = \sum_{p_{slelet}\in \mathrm{P}_{u,v}} \frac{b_{p_{select}} \prod_{e\in p_{select}} r_e}{B_{u,v}} \tag{7}$$

where $\mathrm{P}_{u,v}$ represents the set of all edge-disjoin paths between nodes u and v, $B_{u,v}$ denotes the total bandwidth needed by the flows transmitted from u to v, and $b_{p_{select}}$ represents the bandwidth allocated on path p_{select}.

Before describing the details of MRP-TS, we first give several lemmas that are necessary to illustrate the correctness of MRP-TS:

Lemma 1: $E_C^{u,v}$ is the set of cut edges that affect the connectivity of (u, v). The graph G_S is composed of disjoint subgraphs that are obtain by deleting $E_C^{u,v}$ from graph G. For each $G_S^i \in G_S$, $G_S^i \cap E_C^{u,v} = \phi$.

Proof: G_S is composed of disjoint subgraphs obtained by deleting $E_C^{u,v}$ from graph G, thus $G_s \cap E_c^{u,v} = \phi$. Because $G_S^i \in G_S, \quad G_S^i \cap E_C^{u,v} = \phi$.

Lemma 2: The number of vertices of any $e_c \in E_c^{u,v}$ in each $G_s^i \in G_s$ is no more than 2.

Proof: Supposing that G_S^i contains more than 2 vertices belong to e_c. It is obvious that these vertices can only be the ingress nodes or egress nodes of the flows transmitted between u and v, and there are at least two ingress nodes or two egress nodes in G_S^i. Assuming that there are two inflow nodes A and B in G_S^i. According to lemma 1, the vertices A' and B', which belong to the cut-edges that A or B belongs to, do not belong to G_s^i. Meanwhile, A' and B' do not belong to the same $G_S^j (j \neq i)$. Otherwise, there are two disjoint links between G_S^i and G_S^j, which contradicts the fact that A and B are the vertices of e_c. If A' and A' belong to G_S^m and G_S^n $(m \neq n \neq i)$ respectively, it means that G_S^m and G_S^n can reach G_S^i through two different cut-edges. Because A and B are inflow nodes, u can reach G_S^i via the two cut-edges, which is contradicted with the definition of G_S^i. To sum up, there is no more than one ingress node in G_S^i. Similarly, it can be proved that there is no more than one egress node in G_S^i. The above conclusions are contradicted with that there are at least two inflow nodes or two outflow nodes in G_S^i, thus lemma 2 is correct.

Lemma 3: For the G_S^i that includes source node u or destination node v, the number of vertices of any $e_c \epsilon E_c^{u,v}$ is no more than 1.

Proof: Supposing that G_S^i contains more than one vertex of e_c. According to lemma 1 and lemma 2, the number of qualified vertices can only be 2 (represented by A and B) and the cut-edges the two vertices belong to connect different subgraphs of G_s. If the source node u is included in G_S^i, u can reach two different subgraphs of G_s through the cut-edges that include A and B respectively. Because the two subgraphs have its own path to the destination node v, it is contradictory to the definition of the two cut-edges, which are belong to $E_c^{u,v}$. Therefore, when the source node u is included in G_S^i, only one vertex of e_c can be included in G_S^i. Similarly, it can be proved that only one vertex of e_c can be included in G_S^i when the destination node v is included. To sum up, lemma 3 is correct.

Lemma 4: $E_c^{u,v} \notin$ any cycle of G.

Proof: Assuming that $E_c^{u,v}$ is included in a cycle of G, thus u and v can be connected by two edge-disjoint paths that contain different subgraphs of G_s. According to lemma 1, u and v cannot belong to the same subgraph of G_s. Therefore, the G_s^i that includes u or v has two vertexes of e_c, which is contradictory to lemma 3. Therefore, lemma 4 is proved.

The traffic patterns and network topology of SD-ABN are changeable, which requires the SD-ABN TS scheme to be updated frequently. The TS algorithm using the global optimization idea will bring enormous network control and computation overhead, making it impractical for SD-ABN. Therefore, MPR-TS is designed as a greedy algorithm formulating traffic strategy for each network flow independently according to the real-time network state. The main idea of MRP-TS is that it decomposes a network flow into multiple reliable transmission paths between the source node and the destination node. Based on the idea, the steps of MPR-TS are shown in algorithm 1. The 2-4 lines of MPR-TS ensures the connectivity of (u, v). The line 5 calculates a most reliable routing path using the MREDP algorithm (described in section V.B). The 6-7 lines calculate $E_c^{u,v}$ which represents the set of cut edges affecting the connectivity of (u, v) and generate G_s which is a set of disjoint subgraphs by deleting $E_c^{u,v}$ from the original network topology graph G. The line 8 assigns source and destination nodes to each disjoint subgraph from u, v, and the vertices of each $e_c \epsilon E_c^{u,v}$. If a subgraph has only one node, this node is assigned as the source and the destination both. Lemma 2 and lemma 3 ensure that each disjoint subgraph has only one pair of source and destination nodes. The 9-25 lines compute edge-disjoint paths between the source node and the destination node allocated for each $G_s^i \epsilon G_s$. The TD-BP algorithm, which is described in section V.C, is used to allocate the bandwidth of f to the calculated transmission paths. The number of the edge-disjoint paths within each $G_s^i \epsilon G_s$ is no more than a configured number K while ensuring the transmission reliability requirement of flow f (presented by f_r). The R_{TS_f} in line 17 is calculated according to (8):

$$R_{TS_f} = \prod_{G_S^i \epsilon G_S} R_{Si,Di} \times \prod_{e \epsilon E_c{}^{u,v}} r_e \tag{8}$$

R_{S_i,D_i} can be calculated according to (7), where S_i and D_i are the source node and the destination node allocated to G_s^i. When calculating R_{S_i,D_i} according to (7), $B_{u,v}$ equals to the bandwidth requirement of f (represented by f_b) and $b_{P_{select}}$ represents the bandwidth allocated on path P_{select} for transmitting f. Finally, the TS strategy for f can be obtained by connecting $E_C^{u,v}$ to the transmission paths calculated for each subgraph, and lemma 3 illustrates the correctness of the operation. Fig. 7 gives an example for MRP-TS.

Algorithm 1 The MRP-TS Algorithm

Input: The information of an arrival flow f; The global network view; The maximum number of allowable edge-disjoin paths K between one pair of nodes;
Output: The TS strategy for f, which is represented by TS_f;

1. **Initialize** $k = 1$, $E_C^{u,v} = \phi$, $G_S = \phi$, $TS_f = \phi$;
2. **If** the source node u and the destination node v are not connected
3.   **Return**;
4. **End if**
5. Calculate a most reliable routing path using the MREDP algorithm;
6. Find $E_C^{u,v}$ based on the network topology G;
7. $G_S = G - E_C^{u,v}$;
8. Allocate a source node and a destination node for each $G_S^i \in G_S$ based on u, v, and the vertices of each $e_c \in E_C^{u,v}$;
9. **While** $k \leq K \&\& G_S \neq \phi$
10.   **For** take a G_S^i from G_S
11.   Calculate $k+1$ edge-disjoint paths between the source node and destination node in G_S^i using the MREDP algorithm;
12.   **If** there is no $k+1$ edge-disjoint paths
13. $G_S = G_S - G_S^i$;
14.   Continue;
15.   **End if**
16.   Divide the flow into the $k+1$ paths within G_S^i according to the TD-BP algorithm, and connect $E_C^{u,v}$ to G_S;
17.   Calculate R_{TS_f} which means the transmission reliability of the flow under the current TS scheme;
18.   **If** $R_{TS_f} <= f_r$
19. $G_S = G_S - G_S^i$;
20.   Continue;
21.   **End if**
22.   Update TS_f;
23.   **End for**
24. $k = k + 1$;
25. **End while**
26. **Return** TS_f;

B. THE MREDP ALGORITHM

The most-reliable edge-disjoint path calculation algorithm (MREDP) is a modified algorithm of the edge-disjoint shortest pair algorithm (EDSP) in [28]. According to our path selection metric, the MREDP algorithm changes the way of calculating the path weight in EDSP, and the weight of an edge is represented by its link stability. To find multiple reliable transmission paths, MREDP uses the multiplication operation instead of the summation operation in EDSP, and the negate operation in EDSP is changed to find the reciprocal value. The detail steps of MREDP is shown in algorithm 2.

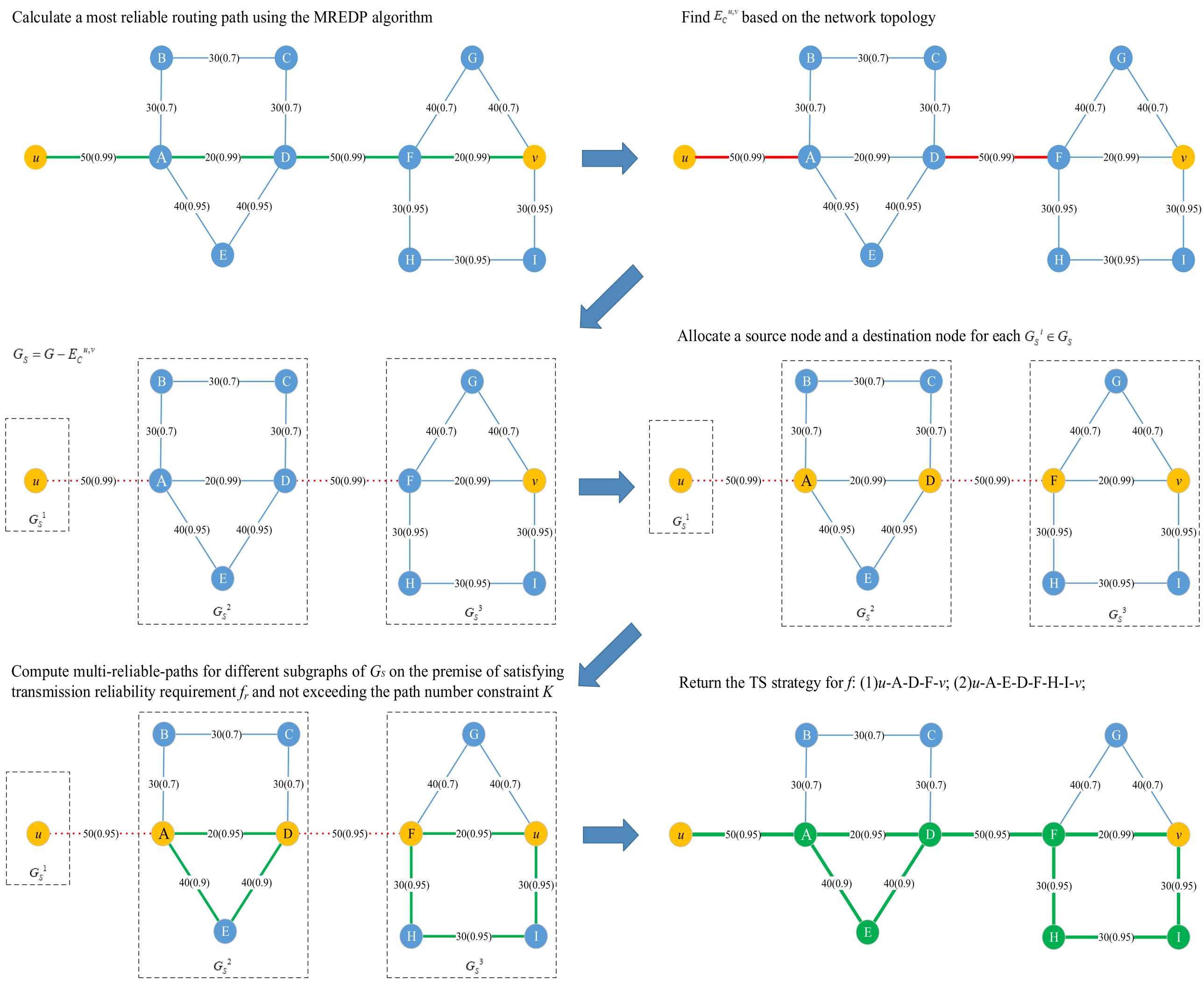


FIGURE 7. An example for MRP-TS ($K = 3, f_r = 0.75, f_b = 50$, source node: u, destination node: v).

MREDP is used to calculate multiple paths that have the maximum sum of reliability value. In line 2, the reliable-first Dijkstra algorithm is designed according to the path selection metric (5) and the modified Dijkstra algorithm in [28] to find the most reliable path between a pair of nodes. The steps of the reliable-first Dijkstra algorithm are the same with that of the modified Dijkstra algorithm, which only changes the weight of each edge and the way of calculating the path weight (The weight of the path is the product of all links' weights on the path). Because the reliable-first Dijkstra algorithm can be easily deduced from other algorithms, it is not described in detail in this paper.

The proof of the correctness of MREDP needs the following 6 definitions:

Definition 1: For the most reliable transmission path P between the source node S and the destination node D, ''Non-decreasing Arc'' is the arc that has a weight value no smaller than 1 and directs from D to S.

Definition 2: ''Decreasing Arc'' is the arc that is not non-decreasing arc.

Definition 3: ''Decreasing Cycle'' is the cycle composed of the decreasing arcs.

Definition 4: If the product of the weight of all arcs in a cycle is not greater than 1, this cycle is defined as the ''Diminished Cycle''.

Definition 5: ''Simple Cycle'' is composed of a set of contiguous non-decreasing arcs and a set of contiguous decreasing arcs. An example of simple cycle is shown in Fig. 8.

Definition 6: ''Mixture Cycle'' is defined to be a cycle composed of the non-decreasing arcs that pertain to P and the decreasing arcs that belong to the rest of the graph. An example for a mixture cycle is shown in Fig. 9.

Lemma 5: Any mixture cycle is diminished cycle.

Proof: For each edge of P, it can be equivalent to the combination of a decreasing arc and a non-decreasing arc, the weight of which are reciprocal to each other.

Algorithm 2 The MREDP Algorithm

Input: The source node S and the destination node D; The global network view; The number of required edge-disjoin paths k;

Output: Multiple edge-disjoin paths between S and D;

1. **Initialize** $n = 1$ and P $= \phi$(P denotes the set of calculated paths);
2. Find the most reliable path between S and D using the reliable-first Dijkstra algorithm and update P;
3. **If** S and D is not connected
4.   **Return**;
5. **End if**
6. **While** $n \leq k$
7.   Replace each edge in P by a single arc directed towards S;
8.   Make the weight of each of the above arcs equal to the reciprocal weight of the corresponding edge;
9.   Run the reliable-first Dijkstra algorithm again from S to D in the above modified graph;
10.   Transform to the original graph, and erase any interlacing edges of the paths found;
11.   Group the remaining edges to obtain n paths between S and D, and update P;
12.   **If** there is no n paths between S and D
13.   **Return**;
14.   **End if**
15. $n = n + 1$;
16. **End while**
17. **Return** multiple edge-disjoin paths between S and D;

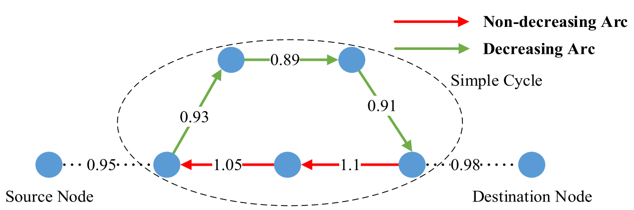


FIGURE 8. An example for a simple cycle.

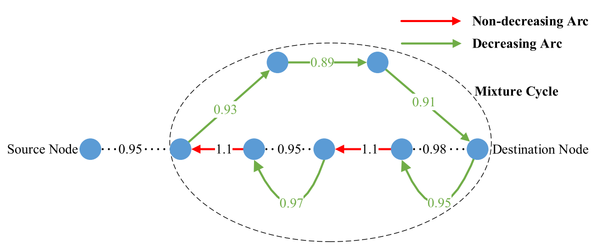


FIGURE 9. An example for a mixture cycle.

Therefore, a given mixture cycle can always be represented as the combination of decreasing cycles and simple cycles. Because P is the most reliable transmission path between S and D, it is obvious that any simple cycle is diminished cycle. Because the decreasing cycle is also diminished cycle, any mixture cycle is diminished cycle as well.

Lemma 6: The MREDP algorithm is correct.

According to lemma 5 and the proofs of the correctness of the modified Dijkstra algorithm in [28], it is easy to prove the correctness of the MREDP algorithm. The detailed proof of lemma 6 is not given in this paper as there are many contents involved, and it is suggested to read the related content in [28] in conjunction with lemma 5.

C. THE TD-BP ALGORITHM

For a new arrival network flow f, the available bandwidth on the transmission path calculated for f may not meet its bandwidth requirement. To satisfy the transmission demand of high-priority flows while taking account of the transmission quality of low-priority flows, we design a bandwidth preemption algorithm that considers traffic decomposition (TD-BP).

Each node of SD-ABN maintains a multi-priority sending queue. Packets in multi-priority queue are sent in order of priority from high to low, which means that the packets in the lower-priority buffers will be sent after the packets in the higher-priority buffers have been sent out, so the transmission demand of higher-priority network flows is guaranteed first. TD-BP is designed based on the priority-based packets transmission mode, and the steps of TD-BP are shown in algorithm 3.

After calculating the transmission paths for f, TD-BP first finds whether all available bandwidth of the transmission paths can satisfy the bandwidth requirement of f. If the bandwidth requirement of f can be satisfied, then f is decomposed according to the proportion of the available bandwidth of different transmission paths. Otherwise, part of the traffic of f is filled with the available bandwidth, and the bandwidth that needs to be preempted by f (presented by) is calculated. After obtaining BP_f, TD-BP calculates the preemptive bandwidth for f (presented by ABP_f), which is the sum of the bandwidth of the flows that have lower priority than f and is transmitted via the same transmission paths as f. If $ABP_f < BP_f$, the preemptive flows are released first and part of BP_f is allocated according to the proportion of the current available bandwidth. The remaining traffic of f is allocated to its most reliable transmission path. If $ABP_f < BP_f$, the preemptive flows are released in turn according to the flow weight function [29] shown in (9) until the available bandwidth can meet BP_f. The flow has a lower weight will be released preferentially. Then, BP_f is allocated according to the proportion of the current available bandwidth of the transmission paths. The information of the released flows is stored in NF, and a released flow will be allocated bandwidth again according to the TD-BP algorithm.

$$W_{f'} = \alpha f'_p + \beta \frac{1}{f'_b} + \eta (f'_b - BP_f)^2 + \delta f'_b \tag{9}$$

In (9), f' denotes a flow whose bandwidth can be preempted by f. α, β, η, and δ are coefficients. $\alpha f'_p$ captures the cost of preempting high priority flows, which can be used to minimize the priority of preempted flows. $\beta \frac{1}{f_{b'}}$ penalizes

Algorithm 3 The TD-BP Algorithm

Input: The information of an arrival flow f; The global network view; The calculated transmission paths for f;
Output: The bandwidth allocation scheme for f;
1. **Initialize** The released bandwidth $B_r = 0$; The new flow set $NF = \phi$;
2. Calculate AB_f which represents all available bandwidth of the transmission paths of f;
3. **If** $AB_f \geq f_b$(f_b means the bandwidth requirement of f)
4. f is decomposed according to the proportion of the available bandwidth of different transmission paths;
5. **Return** the bandwidth allocation scheme for f;
6. **End if**
7. Make part of the traffic of f filled with the available bandwidth;
8. Calculate the bandwidth that need to be preempted by f: $BP_f = f_b - AB_f$;
9. Calculate the preemptive bandwidth of f (represented by ABP_f)
10. **If** $ABP_f < BP_f$
11. All preemptive flows are released;
12. Update the global network view and NF;
13. Part of BP_f is allocated according to the proportion of the current available bandwidth of the transmission paths;
14. The remaining BP_f is allocated to the most reliable transmission path of f;
15. **End if**
16. **If** $ABP_f \geq BP_f$
17. **While** $B_r < BP_f$
18. The preemptive flows are released in turn according to the flow weight function;
19. Update the global network view, NF, and B_r;
20. **End while**
21. BP_f is allocated according to the proportion of the current available bandwidth of the transmission paths;
22. **End if**
23. **Return** the bandwidth allocation scheme for f;

the preemption of low bandwidth flows capturing the cost of preempting a large number of flows, which can be used to minimize the number of preempted flows. $\eta(f_b' - BP_f)^2$ captures the cost of preemption of flows that are much larger or much smaller than BP_f to measure the preempted bandwidth. $\delta f_b'$ captures the cost of preempting large flows to measure the blocking probability.

VI. SIMULATION AND ANALYSIS

In this section, simulation results are given to illustrate the performance of our SR-based TS scheme for SD-ABN. The experiment was conducted based on the network simulator called EXata 5.1. The experiment scenario dimension is set in Cartesian coordinate, where X, Y are both equal to 600km, and the altitude equals 10km above sea level. According to the typical communications range of airborne communications equipment (100km∼300km), we make the primary network size equals to 10 nodes to well coverage our scenario (The number of nodes can vary when discussing the influence of network size on network performance). In the scenario, only one control node is deployed, which manages the network in an out-of-band manner [30], [31]. The moving speed of each node is set to 150m/s-200m/s, and the moving trajectory follows the horse-racing line. A node directionally communicates with other nodes at a transmission rate of 1Gbps, which uses the waveform designed by ourselves. The link layer employs the TDMA protocol with a slot duration of 5ms, and two neighbor nodes occupy transmission slots alternately. Each node maintains a multi-priority sending queue to provide differentiated transmission precedence for flows with different priorities. We make bandwidth preemption an optional function of traffic scheduling, which means that the bandwidth preemption can be enabled as needed. Some important simulation parameters are listed in Table 2.

TABLE 2. Simulation parameters.

Parameter	Value
Network Dimension	$600km \times 600km \times 10km$
Basic Network Size	10 nodes
Number of Control Node	1
Moving Speed	150m/s-200m/s
Number of Beams Per Node	6
Beam Width	$\pi/3$
Link Speed per Beam	1Gbps
Link Layer Protocol	TDMA
Slot Duration	5ms
Number of Priorities	8
Buffer Size for each priority	15000bytes/priority

A. THE PERFORMANCE OF SR IN SD-ABN

We simulate the network updates performance of SR-based two-phase update mechanism in SD-ABN to illustrate the advantages of SR, and the performances of the two-phase update mechanism [32] and the dynamic order update mechanism [33] with OpenFlow as the southbound interface are taken as comparisons. In the experiment, the maximum allowable number of update attempts is set to 5. When using the above three network update mechanisms to respectively schedule a new arrival network flow in SD-ABN, the number of needed update attempts, the network convergence time, and the update success rate are shown in Fig. 10.

As shown in Fig. 10(a), the number of needed update attempts increases with the rise of network size. This is because larger network size increases the number of transmission hops from the control node to the node that needed to be configured, which increases the packet loss rate, and further results in a decrease in the success rate of once update attempt. More transmission hops and update attempts also make it longer for the network to converge to the desired state.

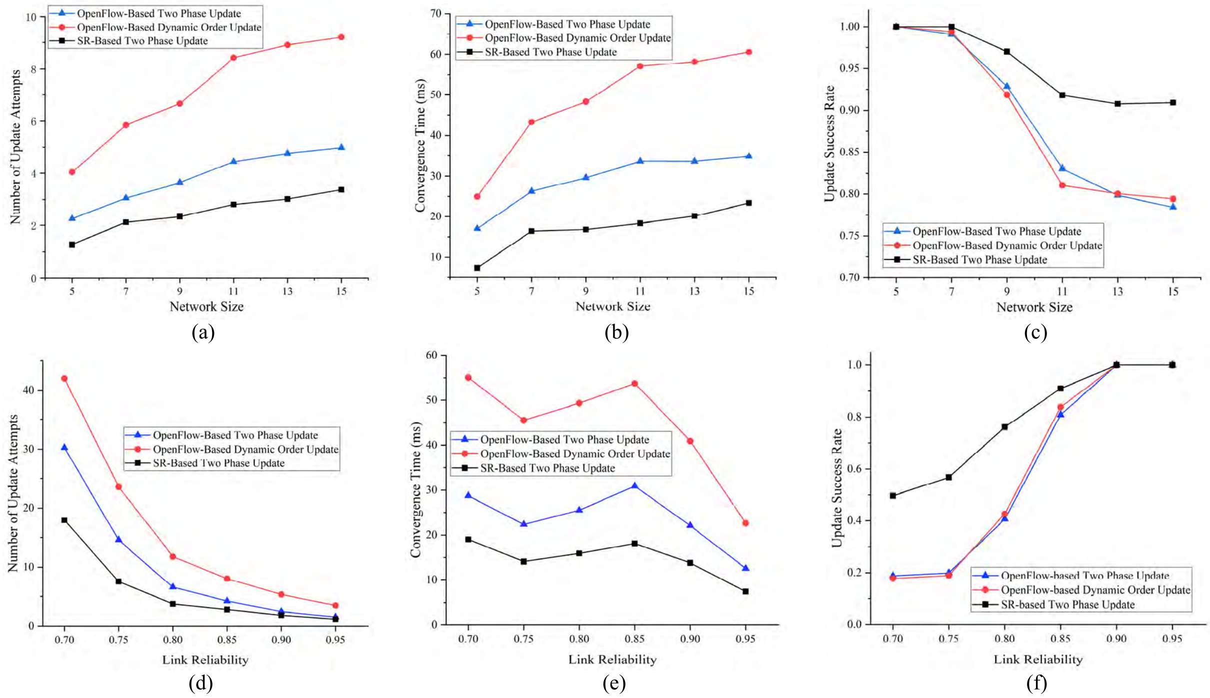


FIGURE 10. Network update performance of SD-ABN. (a) Update times for different network size. (b) Network convergence time for different network size. (c) Update success rate for different network size. (d) Update times for different link reliability. (e) Network convergence time for different link reliability. (f) Update success rate for different link reliability.

Therefore, as shown in Fig. 10(b), the network convergence time also becomes longer with the increase of network size. Meanwhile, because of the limitation of the maximum allowable number of update attempts, as shown in Fig. 10(c), the update success rate becomes lower when network size increases.

As shown in Fig. 10(d), when the transmission link becomes more reliable, the number of update attempts decreases dramatically. The reason is that better link reliability improves the success rate of once update. As shown in Fig. 10(e), with better link reliability, the network convergence time shows a downward trend with a certain degree of fluctuation. The trend is because of the decrease number of update attempts. The fluctuation is because that the increase of link reliability reduces the fluctuation range of the number of update attempts in different topologies. Considering the limitation of the maximum allowable number of update attempts, more network update events need four or five attempts before success, leading to an increase of convergence time. Because the number of update attempts decreases dramatically with the increase of link reliability, as shown in Fig. 10(f), the update success rate increases when the link reliability becomes higher.

Comparing the three network update mechanisms, we can find that, with the increase of network size and link reliability, the SR-based two-phase update mechanism has a fewer number of network update attempts, lower network convergence time, and higher network update success rate. This is because the application of SR in SD-ABN makes the states and forwarding rules of a network flow only maintained and programmed on the ingress node. With the advantage of SR, the SR-based two-phase update mechanism only needs to update the ingress node, which improves the efficiency of network update and alleviates the adverse impact of transmission hops and link reliability.

B. PERFORMANCE OF TRANSMITTING FLOWS IN SD-ABN

To evaluate the transmission performance of SD-ABN, where MRP-TS is used for traffic scheduling, we simulate and analyze the reliability performance, the real-time performance, and the bandwidth utilization of SD-ABN for transmitting random arrival network flows. Because the open shortest path first protocol (OSPF) is an important alternative routing protocol for future airborne backbone network, the performance of OSPF-MDR [34] is taken as a comparison, and the equal-cost multipath routing protocol (ECMP) is employed along with OSPF-MDR to balance the network traffic. Meanwhile, the performance of the reliable-first routing protocol (RFR) which considers transmitting a network flow through the most reliable path instead of the shortest path is also taken as a comparison.

The simulation results of the reliability performance are shown in Fig. 11 (P denotes the priority of the arrival flow, and $p = 7$ represents the highest priority). Comparing the above three traffic scheduling strategies, we can find that the transmission reliability of OSPF+ECMP is always much lower than that of RFR and MRP-TS with the increase of the arrival flow's bandwidth need. This is because OSPF+ECMP

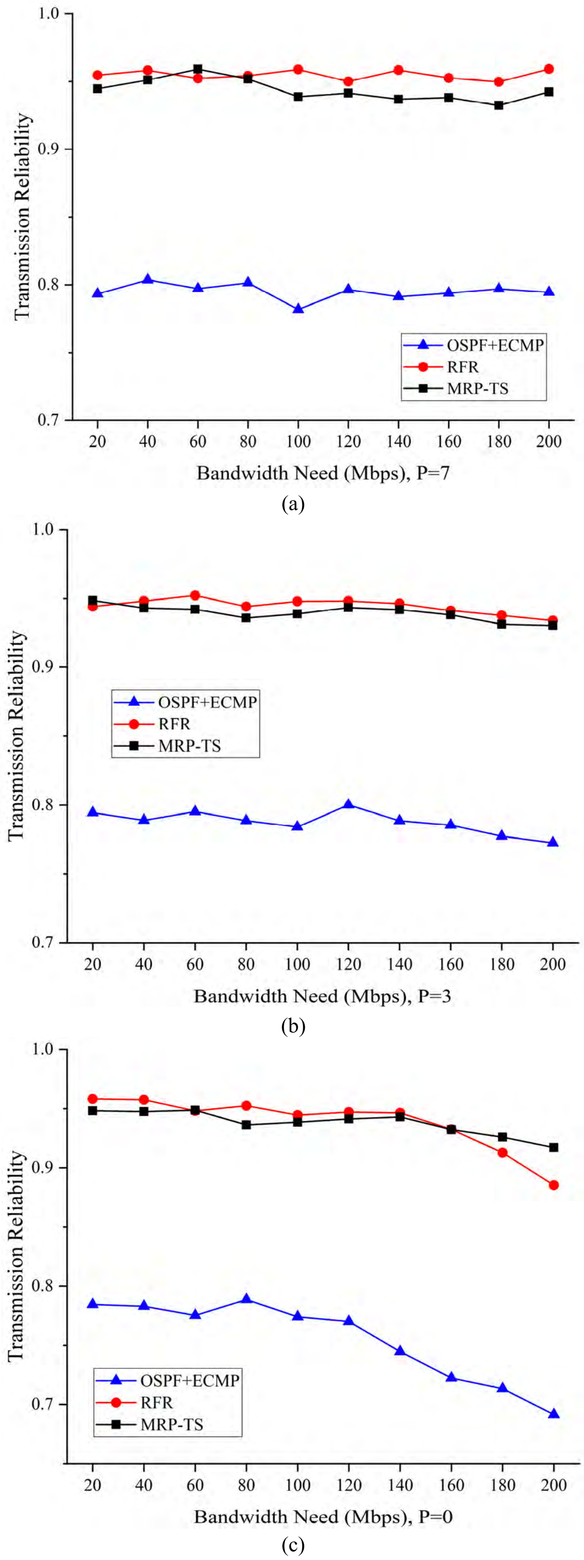


FIGURE 11. **Reliability performance of SD-ABN. (a) Transmission reliability for a flow with P=7. (b) Transmission reliability for a flow P=3. (c) Transmission reliability for a flow with P=0.**

directs the flow to the shortest transmission path without considering the link reliability. Instead, RFR and MRP-TS have considered the link reliability and direct the flow to the reliable transmission paths, making the two traffic scheduling strategies have a good reliability performance. When the priority of the arrival flow is high, the increase of its bandwidth need will not affect its reliability, and this is because the multi-priority sending queue ensures the high-priority flow to be transmitted preferentially. When the priority of the flow becomes lower, the transmission reliability performance of the three traffic scheduling strategies begins to decrease with the increase of bandwidth need. The reason is that more bandwidth need of the flow increases the queuing delay of its packets, resulting in active packet drop (if the packet is invalid) or buffer overflow. In contrast, MRP-TS makes the reliability of the lower-priority arrival flow less affected when the flow's bandwidth need increases. This is because MRP-TS decomposes the traffic more reasonably according to the available bandwidth in the network, which reduces the queuing delay of packets, and thus alleviating the probability of active packet drop or buffer overflow. Moreover, because MRP-TS directs network flows to the path that satisfies its transmission reliability demand rather than the most reliable path, in most cases, the reliability performance of MRP-TS is slightly lower than that of RFR.

Fig. 12 illustrates the real-time performance of SD-ABN. It can be seen that OSPF+ECMP has the best real-time performance in most cases because OSPF+ECMP chooses the shortest path to carry the flow, which significantly reduces the traffic forwarding delay. As shown in Fig. 12(a), when the priority of the arrival flow is the highest ($p = 7$), its real-time performance is not affected by the change of the bandwidth need, and the reason is that the multi-priority sending queue ensures the high-priority flow to be transmitted preferentially. As shown in Fig. 12(b), when the priority of the arrival flow is set to 3, the transmission delay of RFR and OSPF+ECMP increases with the rise of bandwidth need, while the transmission delay of MRP-TS remains stable. This is because MRP-TS enables the flow to preempt the bandwidth of lower-priority flows and balances the traffic more flexible and reasonable than ECMP. As shown in Fig. 12(c), when the priority of the arrival flow is set to 0, the transmission delay of RFR and OSPF+ECMP increases dramatically, and the increase rate of RFR is the largest because RFR cannot balance the traffic. In contrast, with the increase of bandwidth need, the transmission delay of MRP-TS remains the lowest when the bandwidth requirement is greater than 140 Mbps, and MRP-TS increases at a much lower rate than the other two traffic scheduling mechanisms. This is because MRP-TS decomposes the network traffic according to the proportion of available bandwidth, which achieves more flexible and reasonable traffic balance. Although the real-time performance of MRP-TS is not as good as OSPF+ECMP in most cases, it is maintained below 50ms and in the same order of magnitude with OSPF+ECMP, which can well meet the transmission QoS requirements of various real-time tactical flows.

Fig. 13 shows the bandwidth utilization of SD-ABN when using OSPF+ECMP, RFR, and MRP-TS, respectively. It can

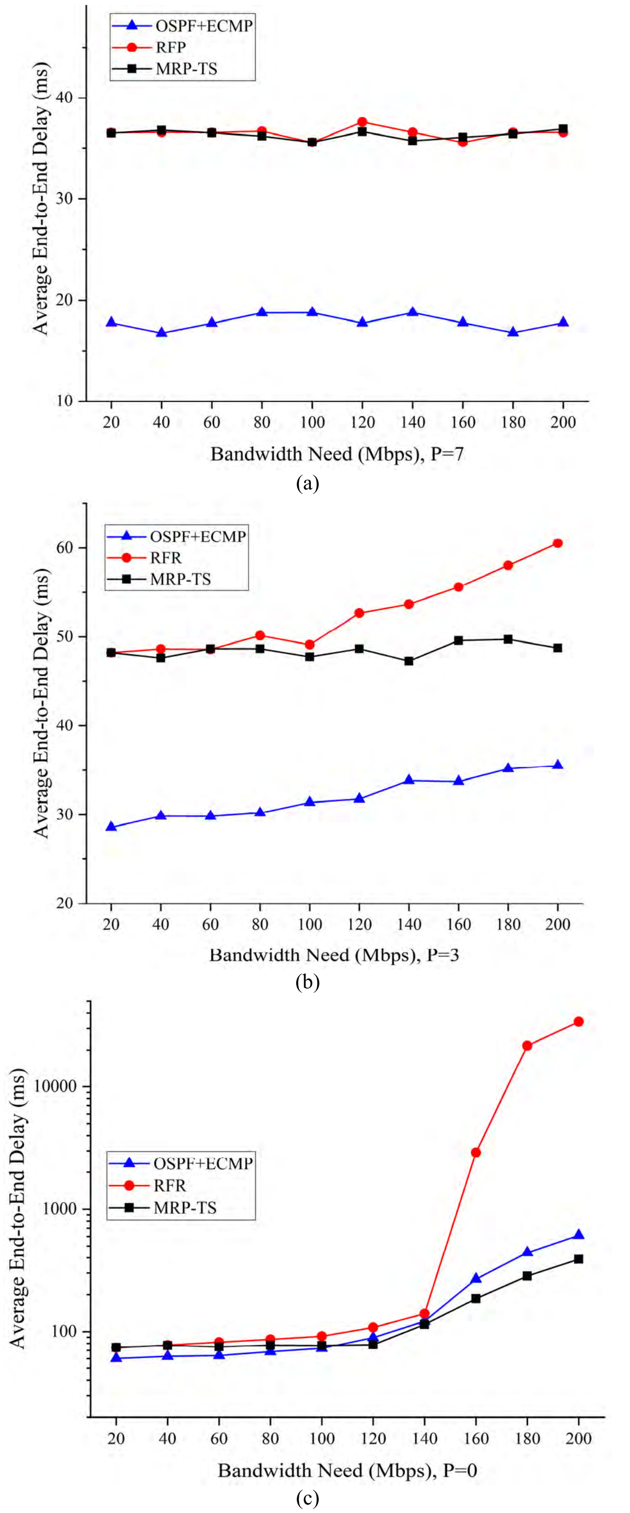


FIGURE 12. Real-time performance of SD-ABN. (a) Transmission delay for a flow with P=7. (b) Transmission delay for a flow with P=3. (c) Transmission delay for a flow with P=0.

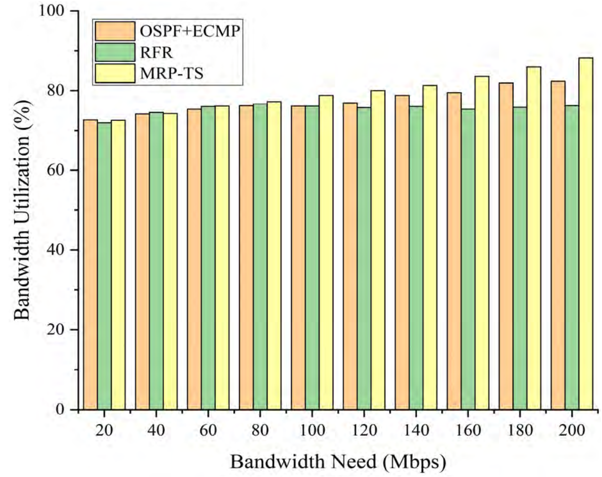


FIGURE 13. Bandwidth utilization of SD-ABN.

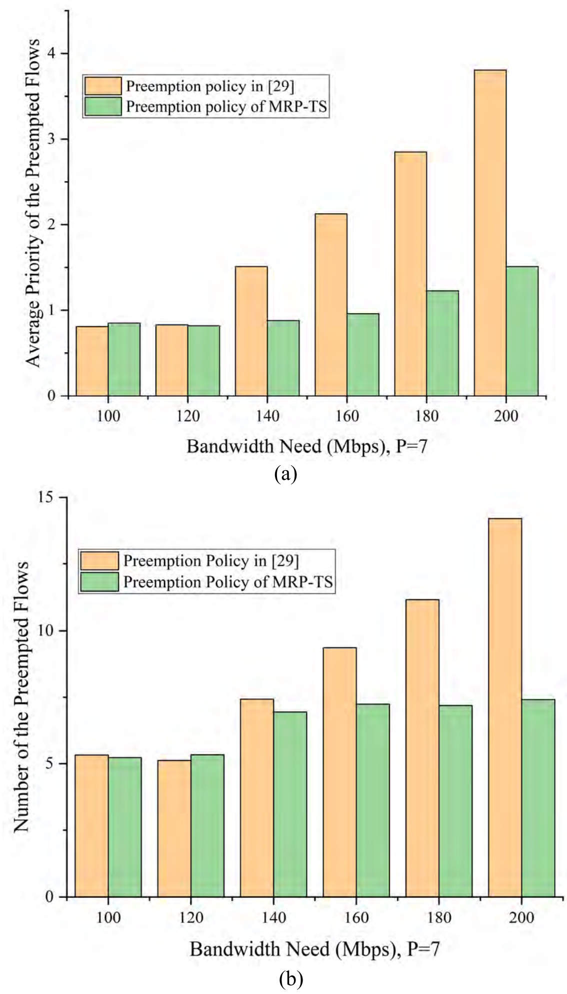


FIGURE 14. Bandwidth preemption performance of SD-ABN. (a) Average priority of the preempted flows. (b) Number of the preempted flows.

be seen that the bandwidth utilization of SD-ABN rises with the increase of bandwidth need of the new arrival flow. This is because the new arrival flow occupies more available bandwidth. MRP-TS can achieve better bandwidth utilization than OSPF+ECMP and RFR because MRP-TS can better balance network traffic and provide available bandwidth for network flows through bandwidth preemption.

C. BANDWIDTH PREEMPTION PERFORMANCE OF SD-ABN

The bandwidth preemption mechanism of MRP-TS guarantees the transmission demand of high-priority flows by preempting the bandwidth of lower-priority flows. This preemption will inevitably affect the transmission quality of the preempted flows. Therefore, it is of great significance to ensure the priority and the number of preempted flows be small enough. Fig. 14 shows the preemption performance of MRP-TS compared with that of [29]. It can be seen that the bandwidth preemption policy of MRP-TS makes the average priority and the number of preempted flows lower than that of the bandwidth preemption policy in [29]. The reason is that the bandwidth preemption policy of MRP-TS considers the decomposition of traffic, which makes better use of the available bandwidth in the network, and thus reduces the preemption bandwidth need.

VII. CONCLUSION

In this study, we proposed the SD-ABN architecture for ABN, which uses the SDN concept to enhance the flexibility, openness, interoperability, and evolvability of an ABN. The advantages of the SD-ABN architecture are comprehensively presented to explain why it can better serve battlefield communications. Based on SD-ABN, segment routing (SR) technology is applied with some practical modifications to make traffic scheduling more concise and more efficient. Moreover, a TS algorithm termed MRP-TS has been proposed, aiming to forward the network traffic of SD-ABN in a reliable and real-time manner. The simulation results have validated the effectiveness of our proposals. In future work, we will standardize the SD-ABN architecture and improve the TS scheme to promote the practicality of our proposals.

REFERENCES

[1] B.-N. Cheng, R. Charland, P. Christensen, L. Veytser, and J. Wheeler, "Evaluation of a multihop airborne IP backbone with heterogeneous radio technologies," *IEEE Trans. Mobile Comput.*, vol. 13, no. 2, pp. 299–310, Feb. 2014.

[2] A. Tiwari, A. Ganguli, A. Sampath, D. S. Anderson, B.-H. Shen, N. Krishnamurthi, J. Yadegar, M. Gerla, and D. Krzysiak, "Mobility aware routing for the airborne network backbone," in *Proc. IEEE Mil. Commun. Conf.*, Nov. 2008, pp. 1–7.

[3] J. Wang, P. Deutsch, A. Coyle, T. Shake, and B. Cheng, "An implementation of a flexible topology management system for aerial high capacity directional networks," in *Proc. IEEE Mil. Commun. Conf.*, Oct. 2015, pp. 991–996.

[4] B.-N. Cheng, F. J. Block, B. R. Hamilton, D. Ripplinger, C. Timmerman, L. Veytser, and A. Narula-Tam, "Design considerations for next-generation airborne tactical networks," *IEEE Commun. Mag.*, vol. 52, no. 5, pp. 138–145, May 2014.

[5] I. F. Akyildiz, A. Lee, P. Wang, M. Luo, and W. Chou, "Research challenges for traffic engineering in software defined networks," *IEEE Netw.*, vol. 30, no. 3, pp. 52–58, May/Jun. 2016.

[6] D. Kreutz, F. M. V. Ramos, and P. E. Verissimo, and C. E. Rothenberg, "Software-defined networking: A comprehensive survey," *Proc. IEEE*, vol. 103, no. 1, pp. 14–76, Jan. 2014.

[7] F. Bannour, S. Souihi, and A. Mellouk, "Distributed SDN control: Survey, taxonomy, and challenges," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 1, pp. 333–354, 1st Quart., 2018.

[8] A. Mendiola, J. Astorga, E. Jacob, and M. Higuero, "A survey on the contributions of software-defined networking to traffic engineering," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 2, pp. 918–953, 2nd Quart., 2016.

[9] C. Y. Hong, S. Mandal, M. Al-Fares, M. Zhu, R. Alimi, K. N. B, C. Bhagat, S. Jain, J. Kaimal, S. Liang, K. Mendelev, S. Padgett, F. Rabe, S. Ray, M. Tewari, M. Tierney, M. Zahn, J. Zolla, J. Ong, and A. Vahdat, "B4 and after: Managing hierarchy, partitioning, and asymmetry for availability and scale in Google's software-defined WAN," in *Proc. SIGCOMM*, Aug. 2018, pp. 74–87.

[10] Z. N. Abdullah, I. Ahmad, and I. Hussain, "Segment routing in software defined networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 1, pp. 464–486, 1st Quart., 2018.

[11] T. I. M. Laberge, C. Filsfils, and P. J. R. Francois, "Tactical traffic engineering based on segment routing policies," U.S. Patent 15 345 049, May 10, 2018.

[12] A. Sen, P. Ghosh, and T. Silva. *Architecture and Algorithms for an Airborne Network*. Accessed: 2010. [Online]. Available: http://netsci.asu.edu/drupal/node/5#tech

[13] J. P. Rohrer, A. Jabbar, E. K. Cetinkaya, E. Perrins, and J. P. Sterbenz, "Highly-dynamic cross-layered aeronautical network architecture," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 47, no. 4, pp. 2742–2765, Oct. 2011.

[14] H. Ahmadi, K. Katzis, and M. Z. Shakir, "A novel airborne self-organising architecture for 5G+ networks," in *Proc. IEEE 86th Veh. Technol. Conf. (VTC-Fall)*, Sep. 2017, pp. 1–5.

[15] L. Gupta, R. Jain, and G. Vaszkun, "Survey of important issues in UAV communication networks," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 2, pp. 1123–1152, 2nd Quart., 2015.

[16] X. Cao, P. Yang, M. Alzenad, X. Xi, D. Wu, and H. Yanikomeroglu, "Airborne communication networks: A survey," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 10, pp. 1907–1926, Sep. 2018.

[17] H. Iqbal, J. Ma, K. Stranc, K. Palmer, and P. Benbenek, "A software-defined networking architecture for aerial network optimization," in *Proc. IEEE NetSoft Conf. Workshops (NetSoft)*, Jun. 2016, pp. 151–155.

[18] I. Zacarias, L. P. Gaspary, A. Kohl, R. Q. A. Fernandes, J. M. Stocchero, and E. P. de Freitas, "Combining software-defined and delay-tolerant approaches in last-mile tactical edge networking," *IEEE Commun. Mag.*, vol. 55, no. 10, pp. 22–29, Oct. 2017.

[19] G. Secinti, P. B. Darian, B. Canberk, and K. R. Chowdhury, "SDNs in the sky: Robust end-to-end connectivity for aerial vehicular networks," *IEEE Commun. Mag.*, vol. 56, no. 1, pp. 16–21, Jan. 2018.

[20] N. Zhang, S. Zhang, P. Yang, O. Alhussein, W. Zhuang, and X. Shen, "Software defined space-air-ground integrated vehicular networks: Challenges and solutions," *IEEE Commun. Mag.*, vol. 55, no. 7, pp. 101–109, Jul. 2017.

[21] Z. Zhou, J. Feng, L. Tan, Y. He, and J. Gong, "An air-ground integration approach for mobile edge computing in IoT," *IEEE Commun. Mag.*, vol. 56, no. 8, pp. 40–47, Aug. 2018.

[22] B.-N. Cheng, J. Wheeler, and L. Veytser, "Radio-to-router interface technology and its applicability on the tactical edge," *IEEE Commun. Mag.*, vol. 50, no. 10, pp. 70–77, Oct. 2012.

[23] A. Bashandy, C. Filsfils, and S. Previdi, "Segment routing with MPLS data plane," IETF draft-ietf-spring-segment-routing-mpls-18, Tech. Rep., 2018.

[24] P. Psenak, S. Previdi, and C. Filsfils, "OSPF extensions for segment routing," IETF draft-ietf-ospf-segment-routing-extensions-27, Tech. Rep., 2018.

[25] F. Paolucci, F. Cugini, A. Giorgetti, N. Sambo, and P. Castoldi, "A survey on the path computation element (PCE) architecture," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 4, pp. 1819–1841, 4th Quart., 2013.

[26] H. Gredler, J. Medved, and S. Previdi, *North-Bound Distribution of Link-State and Traffic Engineering (TE) Information Using BGP*, document RFC 7752, 2016.

[27] S. Sivabalan, C. Filsfils, and J. Tantsura, "PCEP extensions for segment routing," IETF draft-ietf-pce-segment-routing-16, Tech. Rep., 2019.

[28] R. Bhandari, *Survivable Networks: Algorithms for Diverse Routing*. New York, NY, USA: Springer, 1998.

[29] J. de Oliveira, J. P. Vasseur, and L. Chen, *Label Switched Path (LSP) Preemption Policies for MPLS Traffic Engineering*, document RFC 4829, 2007.

[30] K. Chen, N. Lv, and S. Zhao, "A scheme for improving the communications efficiency between the control plane and data plane of the SDN-enabled airborne tactical network," *IEEE Access*, vol. 6, pp. 37286–37301, 2018.

[31] K. Chen, S. Zhao, and N. Lv, "Network monitoring information collection in the SDN-enabled airborne tactical network," *Int. J. Aerosp. Eng.*, vol. 2018, Jul. 2018, Art. no. 1940842.
[32] M. Reitblatt, N. Foster, J. Rexford, C. Schlesinger, and D. Walker, "Abstractions for network update," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 42, no. 4, pp. 323–334, Sep. 2012.
[33] X. Jin, H. H. Liu, R. Gandhi, S. Kandula, R. Mahajan, M. Zhang, J. Rexford, and R. Wattenhofer, "Dynamic scheduling of network updates," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 44, no. 4, pp. 539–550, Oct. 2014.
[34] G. Kuperman, L. Veytser, B.-N. Cheng, S. Moore, and A. Narula-Tam, "A comparison of OLSR and OSPF-MDR for large-scale airborne mobile ad-hoc networks," in *Proc. 3rd ACM Workshop Airborne Netw. Commun.*, Aug. 2014, pp. 17–22.

KEFAN CHEN received the B.S. degree from the University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 2013, and the M.S. degree from Air Force Engineering University, Xi'an, China, in 2016, where he is currently pursuing the Ph.D. degree in aerospace network.

His research interests include airborne tactical networks, software-defined networking, and aviation data link systems.

SHANGHONG ZHAO received the B.S. and M.S. degrees from Physics Department, Lanzhou University, Lanzhou, China, in 1984 and 1989, respectively, and the Ph.D. degree from the Xi'an Institute of Optics and Precision Mechanics, CAS, in 1998.

He is currently a Full Professor with Air Force Engineering University, Xi'an, China. He is also the Head of the Space-Based Information Network Research Center. His research interests include aerospace networks, airborne networks, quantum key distribution, microwave photonics, and optoelectronics. He has authored over 300 academic articles and four books.

NA LV received the B.S. degree in testing technology and instrumentation and the M.S. degree in control theory and applications and the Ph.D. degree in armament science and technology from Northwestern Polytechnical University (NWPU), Xi'an, China, in 1992, 1995, and 2010, respectively.

She is currently a Full Professor with the Air Force Engineering University, Xi'an. Her current research interests include aviation data link systems, military air communications, and software-defined networking.

WEITING GAO received the B.S., M.S., and Ph.D. degrees in communication and information system from Northwestern Polytechnical University (NWPU), Xi'an, China, in 2007, 2011, and 2015, respectively.

He is currently a Lecturer with the Air Force Engineering University, Xi'an. His research interests include airborne networks and communication system design.

XIANG WANG received the B.S., M.S., and Ph.D. degrees in communication and information system from Air Force Engineering University, Xi'an, China, in 2006, 2009, and 2013, respectively, where he is currently a Lecturer.

His research interests include airborne networks and software-defined networking.

XINQING ZOU received the B.S. degree from Hunan University, Changsha, China, in 2017. She is currently pursuing the M.S. degree in communication and information system with Air Force Engineering University, Xi'an, China.

Her research interests include software-defined networking and military aviation communication.