An ICN/SDN-Based Network Architecture and Efficient Content Retrieval for Future Satellite-Terrestrial Integrated Networks

Jian Li, Kaiping Xue, Jianqing Liu, Yongdong Zhang, and Yuguang Fang

Abstract

With the tremendous increase in the amount of network contents and emerging applications, traditional network architecture starts to exhibit management and efficiency problems. The evolving STIN, as a combination of satellite networks and terrestrial networks, also faces similar obstacles. To address this challenge, technologies like ICN and SDN, which were initially proposed for terrestrial networks, could provide viable and promising solutions in STIN. In this article, we propose a novel architecture to provide flexible management and efficient content retrieval for STIN. Our architecture consists of a three-layer satellite network in which satellites of each layer are assigned to different functionalities. In order to achieve compatibility with ICN, we also design a new on-board switch based on protocol oblivious forwarding switch. In so doing, the proposed architecture inherits the advantages of ICN and SDN. Additionally, concerning the issue of the most traffic contribution in content retrieval applications, we propose a simple yet effective cooperative content retrieval scheme for terrestrial users. Furthermore, to make the best use of in-network caching and achieve further efficiency improvement of content retrieval, a cooperative caching scheme and a coded caching scheme are also proposed. The feasibility of these caching schemes is analyzed, and numerical evaluations show that the proposed content retrieval schemes in this architecture can achieve significant traffic load reduction.

Introduction

Current terrestrial networks (e.g., cellular networks) bear the weakness of limited coverage and heavy reliance on infrastructures. What makes these weaknesses more serious is the ever-increasing demand for ubiquitous and seamless services in rural or unreachable areas like oceans and deserts. In light of this, recent research efforts have been devoted to the evolution of conventional terrestrial networks. As one important aspect, Satellite-Terrestrial Integrated Networks (STINs), which consist of the satellite networks and terrestrial networks, has become an attractive research field [1]. By using satellites with different altitudes, that is, Geostationary Earth Orbit (GEO), Medium Earth Orbit (MEO) and Low Earth Orbit (LEO), terrestrial users can benefit from satellites' great characteristics, such as wide coverage and broadcast communication. STIN has the ability to provide seamless and consistent services in extreme environments without terrestrial infrastructures, which makes it a strong supplement to traditional terrestrial networks and a promising direction for future networks.

According to the Cisco's VNI report [2], the demand for multimedia contents has increased tremendously in recent years, and it can be foreseen that content retrieval-based applications will contribute the most Internet traffic. However, the traditional IP-based communication protocol has a number of problems. Specifically, since IP-based networks are designed on the host-to-host communication model, there exists a large number of redundant transmissions in the network which waste the valuable bandwidth of wired/wireless links. Emerging applications also give rise to new requirements such as scalable content distribution, mobility support and so on, while conventional IP-based networks are not designed to address such requirements. Although some functionality patches are introduced to satisfy these new requirements, they inevitably increase the complexity of the overall architecture and are proven to be only temporary solutions [3]. Likewise, adopting the IP-based network protocol will also lead to efficiency and functionality issues in STIN. Therefore, a new network architecture is required not only to overcome the aforementioned drawbacks of the traditional architecture, but also to address the constraints of the resource-limited and highly dynamic environment in STINs.

Some supplemental technologies have been developed which were initially targeted at terrestrial networks, for instance Information-Centric Networking (ICN) and Software Defined Networking (SDN). There are two major characteristics of ICN: in-network caching and routing-by-name. These characteristics enable each node in the network to cache passing-by data and reduce redundant transmissions. In this case, ICN solves the transmission efficiency problem in IP-based networks. Meanwhile, ICN fundamentally decouples information from its sources by means of a clear separation between location and identity. By employing the publish/subscribe communication model [3], it can offer support for high mobility scenarios which naturally leads to applicability in highly dynamic networks, especially STIN. On the other hand, another emerging technology, SDN, has been recognized as a promising architectur-

Digital Object Identifier: 10.1109/MNET.2019.1900138

Jian Li, Kaiping Xue, and Yongdong Zhang are with the University of Science and Technology of China; Jianqing Liu is with the University of Alabama in Huntsville; Yuguang Fang is with the University of Florida, Gainesville.

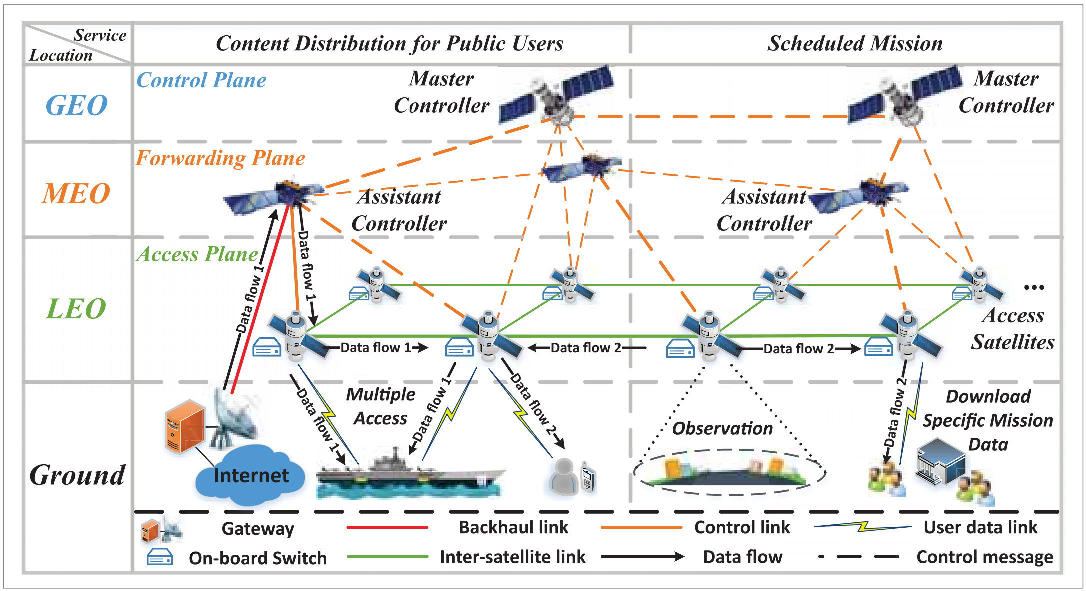

FIGURE 1. Basic components of the proposed architecture in STIN.

al evolution for better network programmability and management. SDN tries to completely decouple the centralized control plane logic from the date-forwarding plane, and it defines a new entity called the controller that centralizes control intelligence of one or more network elements (i.e., switches) [4]. By defining various open interfaces between the control plane and data plane, SDN allows software to be designed independently from the hardware, which greatly simplifies network access, design and operation [5]. Considering various satellite platforms and sophisticated heterogeneous systems in STIN, SDN provides solutions to consolidate different types of network equipment and reduce network complexity. For instance, existing efforts [6, 7] have tried to explore the possibility of bringing SDN to STIN.

Thanks to the rapid development of satellite technologies such as on-board processing and Inter-Satellite Link (ISL) in the last few decades, it becomes possible to apply the mature and newly emerging advanced technologies to satellites [7]. Specifically, considering the aforementioned challenges in current IP-based terrestrial network architecture and the advantages of ICN and SDN, it is worthwhile to integrate ICN and SDN architecture into STIN to achieve more flexible management and efficient communication for future networks. This kind of fusion architecture not only brings advantages in resilience to link disruption and content level security, but also provides possibilities to overcome shortcomings in traditional satellite networks such as the dilemma between dynamic routing and resource conservation, and the difficulty in network upgrade and expansion. In this article, we propose an ICN/SDN-based architecture by leveraging the key technologies of ICN and SDN in STIN and design the specific on-board switch. To show the potential of this novel architecture in STIN, we also propose several caching and transmission schemes for efficient content retrieval. The contributions in this article can be summarized as follows:

- We propose a novel architecture for the future STIN to provide flexible management and efficient content distribution. To implement this architecture in STIN, we design a new on-board switch to support ICN and the specific message format between controller and switch.
- Concerning the issue of the explosive growth of content retrieval, we propose a simple yet effective cooperative content retrieval scheme and introduce the specific retrieval procedure. To make the best use of in-network caching and further reduce the traffic consumption for content retrieval in STIN, we also propose cooperative caching and coded caching schemes.

The rest of this article is organized as follows. At first, an overview of our architecture, the basic components and the specific on-board switch are introduced. Then, a simple yet effective cooperative content retrieval scheme for STIN is given. Two schemes, namely, cooperative caching and coded caching, are introduced for more efficient content retrieval. Then, the performance evaluation is conducted. The final section draws conclusions of our work.

Our Proposed Architecture

Overview

The proposed ICN/SDN-based network architecture is depicted in Fig. 1. We first introduce the basic components of the architecture as well as their specific functionalities. Then, since the traditional SDN switch is incompatible with ICN, we also design a Protocol Oblivious Forwarding (POF) based on-board switch.

Basic Components

As shown in Fig. 1, a multi-layer satellite network is considered as the basic component of our proposed architecture, and it consists of three kinds of satellites, that is, GEO, MEO and LEO satellites. The functionality of each layer is described as follows.

Control Plane: Due to the reliable Inter-Layer Link (ILL), wide coverage, and the nature of broadcasting and stationary to the ground, the GEO satellite is considered as the master controller to manage the LEO satellites. As a master controller, the GEO satellite is in charge of routing strategy calculation, caching strategy update, mobility man-

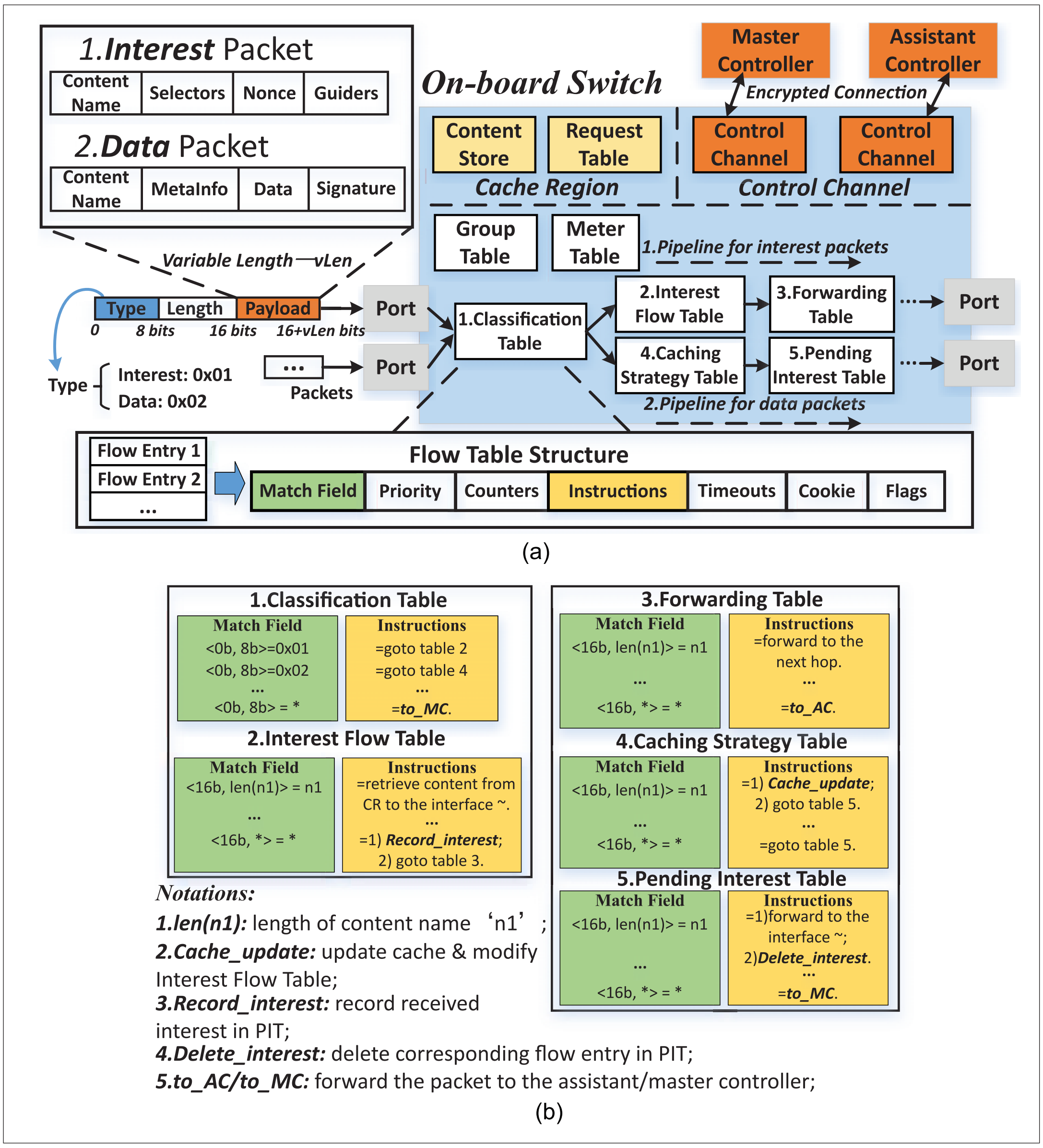

FIGURE 2. The main components of the on-board switch: a) the forwarding procedure in the on-board switch, the packet structure follows a specific ICN project, named data networking [13], and each packet is encoded in Type-Length-Value format; b) the specific flow table.

agement and so on. It also maintains a connection to the Network Operation Control Center (NOCC) on the ground. The NOCC monitors the network status according to the reports from the GEO satellite, and whenever necessary it can also execute complex computation tasks for the controller [7].

Forwarding Plane: Since the distribution of gateway earth stations[1] is heterogeneous and they are separated by thousands of kilometers, LEO satellites with relatively small coverage could not always contact the gateway. To take full advantage of satellite resources and ensure lower propagation delay, MEO satellites are considered as relay nodes which are mainly in charge of data forwarding. Each MEO satellite has a backhaul link to the gateway, which is connected with the terrestrial Internet. In this scenario, the MEO satellite has the ability to retrieve any requested Internet content from the gateway. In order to reduce the burden of the master controller and improve the robustness of the network, the MEO satellite is also considered the assistant controller. It will undertake partial responsibilities as a controller under the command of the master controller and NOCC, and the routing policies for control messages among controllers (i.e., GEO and MEO satellites) are distributed by NOCC according to real-time connectivity.

Access Plane: In 5G systems, satellite access technology has been considered to upgrade performance, reinforce reliability and increase service availability of the current terrestrial networks, and it shall be supported for phase 2. Because of the path-loss propagation, the LEO satellite, which has lower altitude, can achieve better signal quality and lower round-trip delay (up to 50ms) [8, 9]. Thus, the LEO satellite is also preferred by industrial companies to provide worldwide access service, such as the Starlink project of SpaceX. In this case, the LEO satellite is considered the access satellite in our architecture. Each LEO will broadcast access signals periodically, so terrestrial users have the ability to directly communicate with the LEO satel-

[1] For notation convenience, in this article, "gateway" indicates the earth station that provides Internet access for STIN.

lite through portable terminals or moving platform receivers. Although the handover could happen frequently because of the high speed movement of LEO satellites, stable connections can be guaranteed between users and LEO satellites within the coverage period. Normally, handover happens when the LEO satellite moves to another area, and users' incomplete data transmissions should be migrated under the guidance of the controller.

On-Board Switch Design

As the de-facto standard of SDN, OpenFlow abstracts the forwarding operation as a flow table and adopts the match-and-action method to realize flow processing. However, OpenFlow still faces several challenges such as reactive evolving path and almost stateless forwarding plane.

Considering the extension of OpenFlow specification from v1.0 to v1.5 in match fields and actions, the reactive evolving path makes the OpenFlow protocol much more complex. Lacking the capability of stateful network processing, many functionalities can only be realized by the controller which results in scalability and performance issues. Thus, protocol-independent forwarding techniques such as POF [10] and Programming Protocol-independent Packet Processors (P4) [11] have been proposed to further enhance OpenFlow and now they become strong competitors for OpenFlow v2.0. In order to implement our proposed architecture in STIN, we design a new on-board switch based on a POF switch to support ICN functionality. The basic idea of POF is to extract the matching field based on certain {offset, length} tuples from the packet, conduct the table lookups and then execute the associated instructions defined in generic flow instruction sets [12]. The main components of our designed on-board switch are shown in Fig. 2.

At first, in order to achieve in-network caching on each access satellite, each switch is equipped with a Cache Region, which consists of "content store" and "request table" to cache contents and record content popularity, respectively. Then, according to the match-and-action mechanism, we design a specific flow table pipeline to support routing-by-name. As shown in Fig. 2a, each arrival packet enters a "Classification Table" first. The purpose of the "Classification Table" is to classify arrival packets into interest packets and data packets. There are two different pipelines for interest and data packets, which will be introduced as follows.

For an interest packet, the switch first performs a table lookup in the "Interest Flow Table" after the classification. We assume that the "Interest Flow Table" has $M + 1$ flow entries. Since each access satellite is cache-enabled, the first M flow entries in the "Interest Flow Table" are related to M cached contents in the "Content Store," where M is the number of cached contents in the "Content Store." If the packet is matched with one of the first M flow entries in the "Interest Flow Table," the switch will return the related content cached in the "Content Store" backward. Otherwise, the last flow entry with the lowest priority, which is the table-miss flow entry and represented as "$\langle$16 bits,*$\rangle$ = *" in Fig. 2b, will be identified by its match. Then, the interest packet will be forwarded to the nearest assistant satellite for content retrieval from the gateway.

For a data packet, the switch first performs a table lookup in the "Caching Strategy Table" to decide whether the switch updates cached contents in the "Content Store" or not. Similar to the procedure above, if a data packet is matched with one of the first M flow entries, the content with the lowest priority cached in the "Content Store" will be replaced by the content contained in the data packet. Then the data packet will go to the "Pending Interest Table" regardless of whether it is matched with the first M flow entries or not. In the "Pending Interest Table," each data packet should be matched with one of the first M flow entries and follow the reverse path of the interest packet to reach the source. Normally, it should not be matched with the table-miss flow entry in the "Pending Interest Table." If there occurs an error and the packet is matched with the table-miss, it will be sent to the master controller.

Note that the "Interest Flow Table" and "Pending Interest Table" are two stateful flow tables. They are related to the state of the "Content Store" and record the incoming interfaces of received interest packets, respectively. Since the specific implementation of the flow table is optional in POF [10], we consider adopting shared table resources in the on-board switch. Thus, the "Content Store" and "Request Table" in the Cache Region can be considered as two special flow tables occupying the fixed storage space. In this case, the storage space occupied by the Cache Region can be adjusted and recycled according to the specific requirements by the controller. For security reasons, the control signal exchange between the switch and controller can use the default transport layer security protocol in OpenFlow such as SSL, and the switch can also use pre-shared key exchange to authenticate the entities.

Cooperative Content Retrieval Scheme

Concerning the issue of the most traffic contribution in content retrieval applications, we first propose a basic scheme in this section, which is a simple yet effective cooperative content retrieval scheme with one-hop neighbor cache sharing. In the next section, we further propose several improved caching schemes to achieve better performance.

Basic Idea

To introduce the basic idea of the cooperative content retrieval scheme, an example is depicted in Fig. 3. There are two methods to retrieve the requested content in our scheme. One is to forward user's requests to the connected assistant controller through ILL and then retrieve the corresponding content from the gateway. The other is to retrieve the content from a neighbor access satellite with the guidance of the controller. In Fig. 3, *User1* sends requests for contents *A*, *B*, *C* to *LEO1*. Since *LEO1* did not cache these contents before, these requests are forwarded to the gateway through the assistant controller. After the requested contents are retrieved from the gateway, *LEO1* will cache these contents under the caching strategy and send these contents to *User1*. Since the assistant controller has full knowledge of the caching strategies of all access satellites and it is in charge of content retrieval from the gateway, the assistant controller will update the routing strategy of *LEO1*'s neighbor satellite,

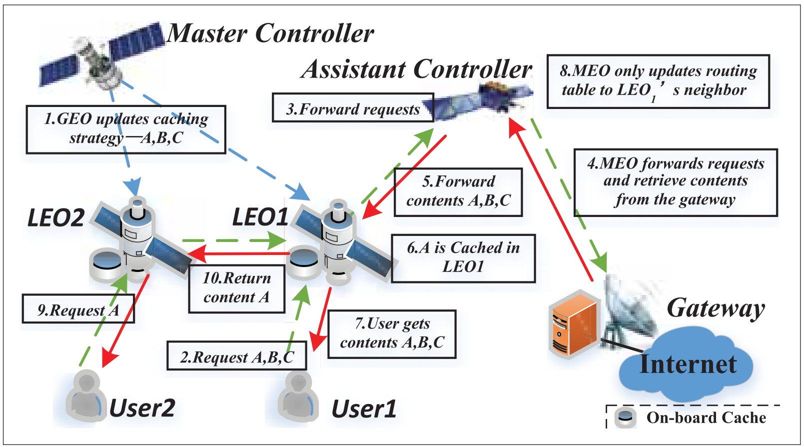

FIGURE 3. Content retrieval scheme in the proposed architecture.

that is, *LEO2*. After that, *User2*'s requests for content *A* will be forwarded to *LEO1* through ISL. In this way, *User2*'s requests will be satisfied within two-hop rather than three-hop forwarded by the assistant controller.

Cooperative Content Retrieval Procedure

The specific procedure of the aforementioned cooperative content retrieval scheme is depicted in Fig. 4. Considering the limited bandwidth of backhaul links, we set an aggregation window T_a at *MEO*, which means that *MEO* will wait time slot T_a after the first arrival request from access satellites (i.e., *LEO1* and *LEO2*). In this way, *MEO* could aggregate requests from access satellites to reduce redundant transmissions. Note that *LEO1* and *LEO2* are one-hop neighbors to each other, and their caches are empty at the beginning of the procedure. The caching strategy of *LEO1* and *LEO2* is to cache content *A*, *B*, *C* and *B*, *C*, *D*, respectively. The specific procedure can be described as follows:

1. According to the historical content popularity information, *GEO* sends a control message to modify the "Caching Strategy Table" in *LEO1* and *LEO2*.
2. *User1* sends requests (i.e., interest packets) to *LEO1* for contents *A*, *B*, *C*. Then these requests are forwarded to *MEO*.
3. *User2* sends requests to *LEO2* for content *A*. The requests are also forwarded to *MEO*.
4. Within the aggregation window T_a, *MEO* receives four requests from *LEO1* and *LEO2*, and two requests for content *A* from *User1* and *User2* are aggregated into one request. Then *MEO* forwards three unique requests to the gateway.
5. After retrieving contents *A*, *B*, *C* from the gateway, *MEO* returns the corresponding contents. According to the "Caching Strategy Table" modified by the master controller, contents *A*, *B*, *C* will be cached when they arrive at *LEO1*.
6. *LEO1* forwards contents *A*, *B*, *C* to *User1*, while *LEO2* forwards content *A* to *User2*.
7. *MEO* has full knowledge of the caching strategy determined by *GEO*. After *MEO* sends contents *A*, *B*, *C* to *LEO1*, it sends a control message to modify the "Forwarding Table" about the routing strategy of contents *A*, *B*, *C* in *LEO2*.
8. *User2* sends a request to *LEO2* for content *B* which is forwarded to *LEO1* according to the routing strategy.
9. *LEO1* returns content *B* and *LEO2* caches content *B* according to the caching strategy. Then *LEO2* returns content *B* to *User2*.
10. Since content *B* is not originated from MEO, after *B* is cached by *LEO2*, *LEO2* sends a status report to inform *MEO* that *LEO2* has updated its "Content Store." In this case, *MEO* always has full knowledge of all cache status in the access satellites.

Note that only one-hop neighbor content retrieval is considered for architecture simplification in this article. The reason is that multi-hop transmissions between access satellites will cause traffic consumption to increase rapidly. Considering that small satellites are preferred as LEO satellites, it is unrealistic to put a heavy traffic load on resource-limited LEO small satellites.

Interaction

In order to update routing strategy and caching strategy, the controller should send control messages to modify the corresponding flow table in on-board switches. As shown in Fig. 4, *GEO* solicits the status of access satellites periodically. Specifically, it sends requests to *LEO1* and *LEO2* for content popularity in terms of symmetric messages in every update period T_c. When *LEO1* and *LEO2* receive the request message, they will return popularity information stored in the "Request Table" to *GEO* immediately. After collecting the popularity information and making caching decisions for *LEO1* and *LEO2*, *GEO* will send another control message to update caching strategy in *LEO1* and *LEO2*. It should be noted that *GEO*, as the master controller, has full access to modify all flow tables of access satellites. The reason is that *MEO* can perceive the change of cache status earlier than *GEO* since it directly retrieves contents from access satellites. By doing so, the number of messages through the control channel for reporting cache status can also be reduced. For specific control message formats, they could be realized by utilizing the "*experimenter*" field based on the OpenFlow protocol specification [14].

Improved Caching Schemes for Efficient Content Retrieval

Cooperative Caching Scheme

In specific ICN projects such as Named Data Networking (NDN) [3], local caching policies like Least Recently Used (LRU) and Least Frequently Used (LFU) are usually considered for scenarios like lack of global content popularity information. However, due to the high-speed relative motion between access satellites and the earth, a satellite's coverage area changes frequently. Meanwhile, considering the inhomogeneous population distribution and variant user preferences, there are also tremendous differences in the number of content requests and content popularity in different geographic regions. In this case, simply adopting local caching policies will lead to frequent cache updates.

In the proposed cooperative content retrieval scheme, since the master controller is aware of request frequency and content popularity in each

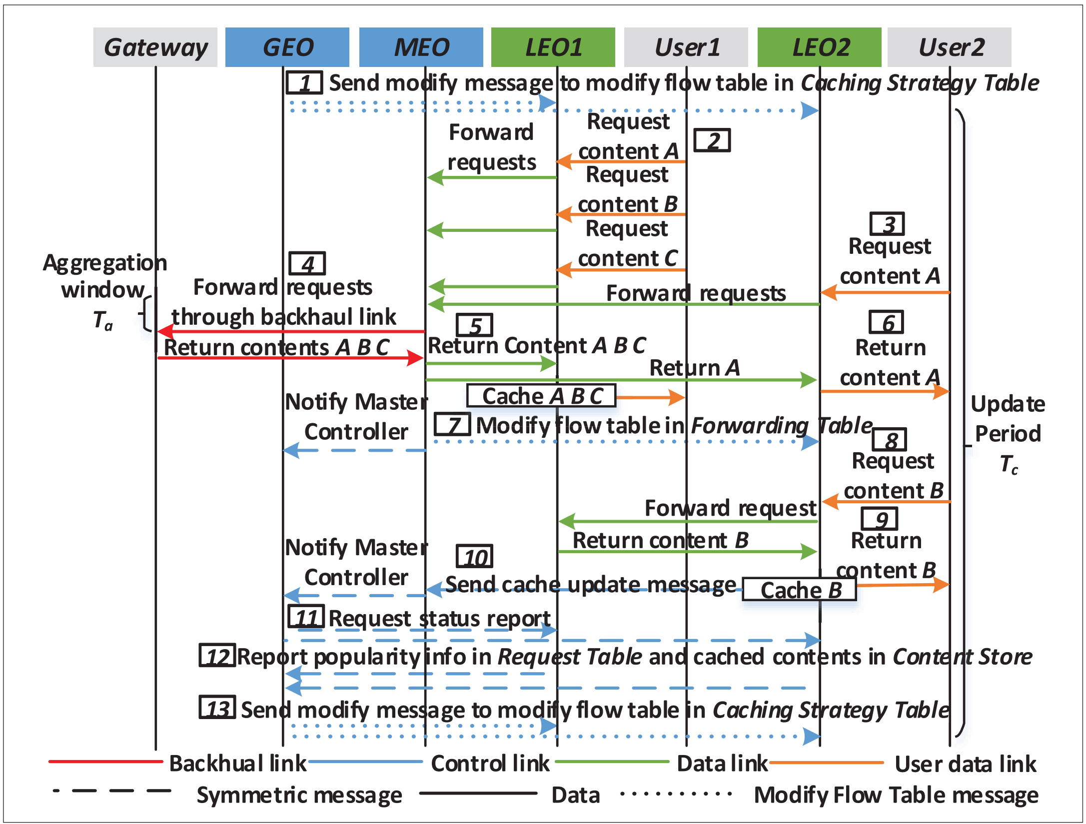

FIGURE 4. The specific procedure of the proposed cooperative content retrieval.

area covered by access satellites through periodical status collection, a novel caching scheme which considers the diversity of users' requests in different geographic regions is preferred. In order to maximize the caching gain in access satellites, we propose a cooperative caching scheme based on our proposed architecture. First of all, the earth is divided into a number of areas and the content popularity will be recorded with respect to each specific area. Then, the popularity of i-th content in the n-th area can be denoted by

$$p_{in} = \frac{R_{it}}{\sum_{t \in T_n^s} R_{it}},$$

where T_n^s is the time duration when satellite s covers the n-th area, and $\Sigma_{t \in T_n} R_{it}$ denotes the historical requests for i-th content.

On the one hand, considering the diversity of user preferences in different areas, we present a relay caching method to reduce caching update costs in access satellites. The basic idea here is that when an access satellite is about to leave the divided area, the controller will notify the access satellite to start the active caching update process. As shown in Fig. 5a, *LEO2* is about to leave *Area2*. Since *LEO1* is the last access satellite which covers *Area1*, *LEO2* will send requests to *LEO1* for its cached contents under the coordination of the master controller. In order to realize this caching update process, the controller should first send control messages to update the caching strategy and routing strategy in *LEO2*. Then *LEO2* generates corresponding requests which are sent to *LEO1* to acquire its cached contents. With the equal division of the area, each access satellite can start the active caching update process one

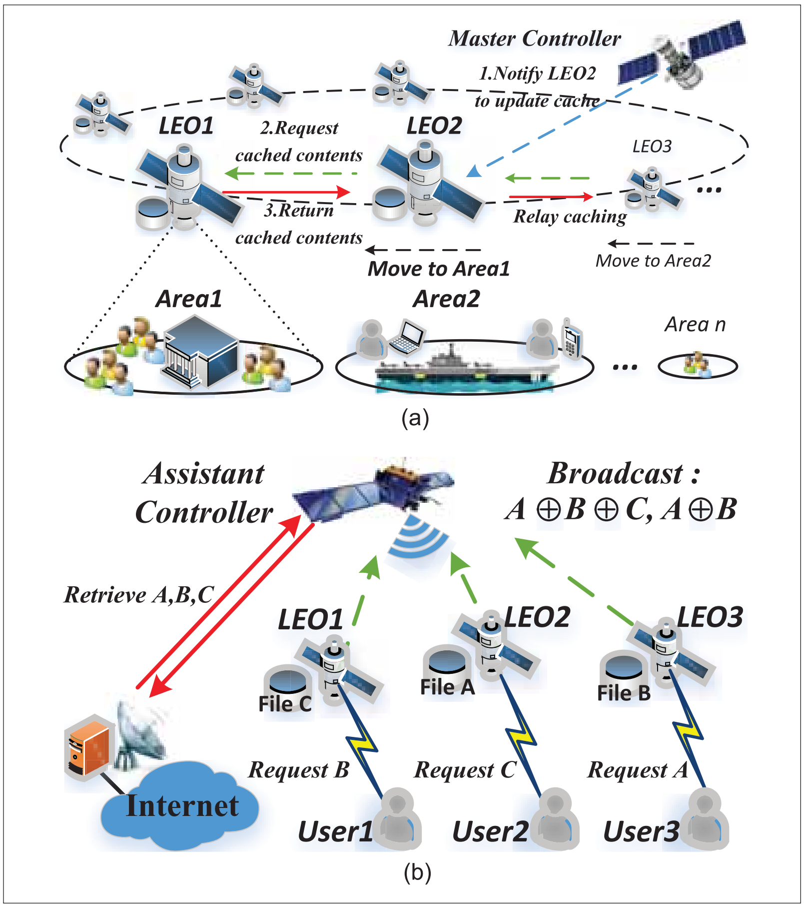

FIGURE 5. Performance improvement schemes for content retrieval in the proposed integrated architecture: a) relay caching method in cooperative caching scheme; b) a coding opportunity in coded caching scheme.

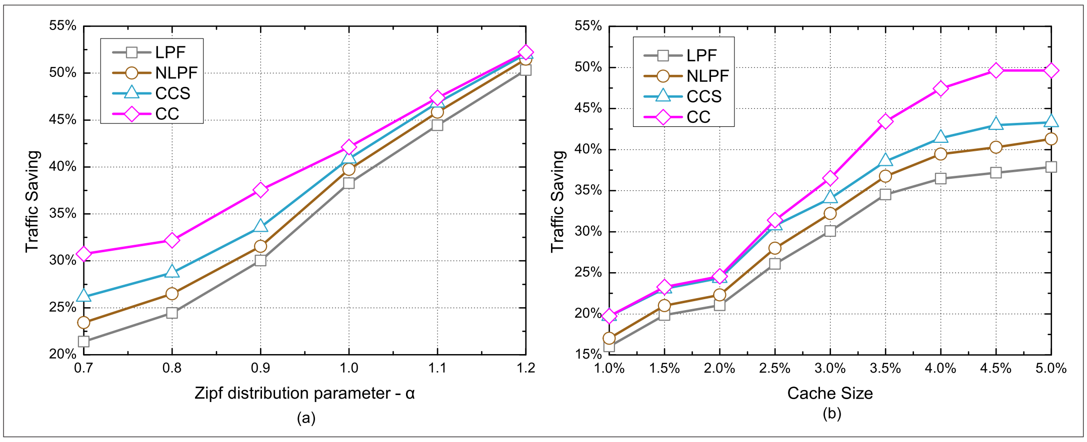

FIGURE 6. Performance comparison between different caching strategy: a) performance comparison versus α (cache size = 3 percent); b) performance comparison versus cache size (α = 0.8).}

by one under the coordination of the master controller. In this case, only ISLs are used and there is a great reduction in the backhaul link.

On the other hand, considering the significant differences in the number of requests due to the inhomogeneous population distribution, a cooperative caching strategy is designed. According to the number of user requests, each area on the earth could be described as a popular area or an ordinary area. In order to minimize total traffic load in STIN, our cooperative caching strategy primarily considers the caching requirements of the dominant traffic contributed by popular areas, which means access satellites covering ordinary areas should help access satellites covering popular areas to cache popular contents. Note that each satellite only helps one-hop neighbor satellites to cache popular contents in this article.

Coded Caching Scheme

In recent years, coded caching has been proposed as an effective approach to the distributed caching problem and can achieve a significant reduction in traffic load. The basic idea of a coded caching scheme is to create coding opportunities at the server in cache-enabled networks through a shared link [15]. The sender is assumed to have access to the database of all files and have the full knowledge of the cache status of connected users. After it receives requests from connected users, it could create coding opportunities at the transmission phase according to users' cached contents to achieve global caching gain.

Our coded caching scheme is shown in Fig. 5b. As aforementioned, a user's request will be forwarded to the assistant controller if the requested content has not been cached in the access satellite. Since the assistant controller has full knowledge of cached contents in each access satellite, there exists coding opportunities after an aggregation window by adopting an online coded caching algorithm [15]. In order to balance the performance improvement and QoS requirement, the size of the aggregation window can be dynamic to satisfy the delay requirement, and our coding scheme should not be applied to specific contents for real-time services. In Fig. 5b, *User1*, *User2* and *User3* request content *B*, *C* and *A*, respectively. Although these contents are already cached in *LEO1*, *LEO2*, and *LEO3* separately, these cached contents cannot be shared if *LEO1*, *LEO2*, and *LEO3* are not one-hop neighbors. In this case, by encoding content packets *A*, *B*, *C* into $A \oplus B \oplus C$ and $A \oplus B$, we successfully save 1/3 bandwidth in ILL, where $\oplus$ denotes XOR operation. After *LEO1*, *LEO2*, and *LEO3* receive coded packets $A \oplus B \oplus C$ and $A \oplus B$, requested content can be easily decoded using another XOR operation with their cached contents. Thus, our coded caching scheme only brings negligible XOR operation overhead to the resource-limited satellite in return for significant bandwidth saving in ILL.

Performance Evaluation

In this section, we evaluate the performance of our proposed content retrieval schemes to show the potential performance of our ICN/SDN-based architecture in future networks. We construct a satellite cluster with five access satellites. The number of total files in the content library is 10^3 and the content popularity in each coverage area follows Zipf distribution, which states that the probability of a request for the *i*-th most popular content is proportional to $1/i^{\alpha}$ with the sharp parameter α. The user requests are generated following Poisson distribution. The performance is evaluated in terms of traffic saving, which is captured as the sum of traffic savings in ISL, ILL and backhaul link. Four schemes are taken in the performance comparison.

Local Popularity First (LPF): The LPF scheme simply caches the most popular contents in its own coverage area.

Neighbor Content Retrieval Enabled LPF (NLPF): The NLPF scheme also caches the most popular contents but enables one-hop neighbor satellites to share their cached contents.

Cooperative Caching Strategy (CCS): Considering popularity distribution and the number of requests in each coverage area, the CCS scheme tries to find the global optimal caching solution which allows one-hop neighbor cache sharing.

Coded Caching Enabled CCS (CC): In the CC scheme, the assistant controller adopts opportunistic network coding within an aggregation window according to cached contents in each access satellite.

We consider the performance of the simple LPF scheme as the baseline in the comparison, and the cache size in Fig. 6 represents the storage capacity of the percentage of the content library. In general, our proposed cooperative content retrieval scheme (i.e., NLPF) can achieve performance improvement in terms of traffic saving in various scales. As shown in Fig. 6, NLPF always outperform the baseline in different settings, which proves the effectiveness of neighbor cache sharing. Since only one-hop neighbor cache sharing is adopted, it also has the potential to acquire further performance improvement with multi-hop neighbor cache sharing.

In Fig. 6a, with the increasing of α, the performance gap among the four schemes is diminishing. This is because users' requests are converging on several popular contents with the increasing of α. Thus, by caching several popular contents, most traffic can be saved. Another result in Fig. 6b shows that the difference between NLPF and the baseline and the difference between CC and the baseline are both enlarging. The reason can be explained as follows. For the NLPF scheme, the hit possibility of one-hop neighbor cached contents is growing with the increasing of the cache size. For the CC scheme, since coding opportunities of online coded caching are dependent on the global cached contents of access satellites, the performance improvement of the CC scheme increases with the increasing of the cache size. Therefore, considering at most 12.5 percent traffic saving compared with the baseline, the efficiency of our proposed content retrieval and caching schemes can be verified.

Conclusion

In this article, we proposed a novel ICN/SDN-based architecture, which can better cope with future challenges such as rapidly increasing access requests and management of heterogeneous networks. Thus, it provides a possible direction for the future design of STIN. We focused on the content retrieval application, which contributes the most traffic to the Internet, and proposed a simple yet effective cooperative content retrieval scheme. By adopting cooperative caching and coded caching, the proposed scheme could achieve further performance improvement in terms of traffic saving. In this case, the proposed heuristic schemes have guiding significance for the design of a practical content retrieval scheme in STIN.

Acknowledgments

This work is partly supported by the National Key Research and Development Program of China under Grant No. 2016YFB0800301; the Youth Innovation Promotion Association CAS under Grant No. 2016394; and the National Natural Science Foundation of China under Grant No. 91538203.

References

[1] Y. Cao *et al.*, "Optimal Satellite Gateway Placement in Space-Ground Integrated Network for Latency Minimization with Reliability Guarantee," *IEEE Network*, vol. 32, no. 5, 2018, pp. 32–37.

[2] Cisco, "Cisco Visual Networking Index: Forecast and Trends, 2017–2022," 2018.

[3] G. Xylomenos *et al.*, "A Survey of Information-Centric Networking Research," *IEEE Communication Surveys & Tutorials*, vol. 16, no. 2, 2014, pp. 1024–49.

[4] L. Bertaux *et al.*, "Software Defined Networking and Virtualization for Broadband Satellite Networks," *IEEE Commun. Mag.*, vol. 53, no. 3, 2015, pp. 54–60.

[5] C. Yang *et al.*, "When ICN meets C-RAN for HetNets: An SDN Approach," *IEEE Commun. Mag.*, vol. 53, no. 11, 2015, pp. 118–25.

[6] J. Liu *et al.*, "Joint Placement of Controllers and Gateways in SDN-Enabled 5G-Satellite Integrated Network," *IEEE JSAC*, vol. 36, no. 2, 2018, pp. 221–32.

[7] J. Bao *et al.*, "OpenSAN: A Software-Defined Satellite Network Architecture," *ACM SIGCOMM Computer Communication*, 2014.

[8] X. Zhu *et al.*, "Cooperative Multigroup Multicast Transmission in Integrated Terrestrial-Satellite Networks," *IEEE JSAC*, vol. 36, no. 5, 2018, pp. 981–92.

[9] "TR 38.913: Study on scenarios and requirements for next generation access technologies," 3GPP, 2018.

[10] H. Song, "Protocol-Oblivious Forwarding: Unleash the Power of SDN through a Future-Proof Forwarding Plane," ACM SIGCOMM HotSDN Workshop, 2013, pp. 127–32.

[11] P. Bosshart *et al.*, "P4: Programming Protocol-Independent Packet Processors," *ACM SIGCOMM Computer Commun. Review*, vol. 44, no. 3, 2014, pp. 87–95.

[12] S. Li *et al.*, "Protocol Oblivious Forwarding (POF): Software-Defined Networking with Enhanced Programmability," *IEEE Network*, vol. 31, no. 2, 2017, pp. 58–66.

[13] L. Zhang *et al.*, "Named Data Networking," *ACM SIGCOMM Computer Commun. Review*, vol. 44, no. 3, 2014, pp. 66–73.

[14] "Openflow switch specification v1.5.1." https://www.opennetworking.org/wp-content/uploads/2014/10/openflow-switch-v1.5.1.pdf/; accessed April 4, 2019.

[15] R. Pedarsani, M. A. Maddah-Ali, and U. Niesen, "Online Coded Caching," *IEEE/ACM Trans. Networking*, vol. 24, no. 2, 2016, pp. 836–45.

Biographies

Jian Li (lijian9@mail.ustc.edu.cn) received his B.S. degree from the Department of Electronics and Information Engineering, Anhui University, Hefei, China, in 2015. He is currently working toward the Ph.D degree in communication and information systems with the Department of Electronic Engineering and Information Science, University of Science and Technology of China. His research interests include satellite networks, cooperative transmission in integrated satellite/terrestrial networks and cache-enabled wireless heterogeneous networks.

Kaiping Xue [M'09, SM'15] (kpxue@ustc.edu.cn) received his B.S. degree from the Department of Information Security, University of Science and Technology of China (USTC), in 2003 and received his Ph.D. degree from the Department of Electronic Engineering and Information Science (EEIS), USTC, in 2007. Currently, he is an associate professor in the Department of Information Security and the Department of EEIS, USTC. His research interests include next-generation Internet, distributed networks and network security. He is the corresponding author of this article.

Jianqing Liu (jianqing.liu@uah.edu) received his B.Eng. degree from the University of Electronic Science and Technology of China, Chengdu, China, in 2013, and received his Ph.D. degree from the Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL, USA, in 2018. Currently, he is an assistant professor in the Department of Electrical and Computer Engineering at the University of Alabama in Huntsville. His research interests include wireless networking and network security in cyber-physical systems.

Yongdong Zhang [A'09, SM'13] (zhyd73@ustc.edu.cn) received the Ph.D. degree in electronic engineering from Tianjin University, Tianjin, China, in 2002. He is currently a professor in the Department of Electronic Engineering and Information Science (EEIS), University of Science and Technology of China (USTC). His current research interests include multimedia content analysis and understanding, multimedia content security, video encoding, and streaming media technology. He is a member of the Editorial Board of the *IEEE Transactions on Multimedia* and *Multimedia Systems Journal*.

Yuguang Fang [S'92, M'97, SM'99, F'08] (fang@ece.ufl.edu) received the M.S. degree from Qufu Normal University, Shandong, China in 1987, the Ph.D. degree from Case Western Reserve University in 1994, and the Ph.D. degree from Boston University in 1997. He joined the Department of Electrical and Computer Engineering at the University of Florida in 2000 and has been a full professor since 2005. He holds a University of Florida Research Foundation Professorship (2017-2020, 2006-2009), a Changjiang Scholar Chair Professorship (Xidian University, Xi'an, China, 2008-2011; Dalian Maritime University, Dalian, China, 2015-2018), and an Overseas Academic Master (Dalian University of Technology, Dalian, China, 2016-2018). He received the NSF CAREER Award in 2001, the ONR Young Investigator Award in 2002, the 2015 IEEE Communications Society CISTC Technical Recognition Award, the 2014 IEEE Communications Society WTC Recognition Award, and the Best Paper Award from IEEE ICNP 2006. He was the editor-in-chief of *IEEE Transactions on Vehicular Technology* (2013-2017), and the editor-in-chief of *IEEE Wireless Communications* (2009-2012). He is a fellow of the IEEE and AAAS.