Multi-QoS adaptive routing algorithm based on SDN for satellite network

Xiaodong Shi*, Yongjun Li, Shanghong Zhao and Weilong Wang
Information and Navigation College, Air Force Engineering University, Xian, China

*Corresponding author e-mail: 2272731016@qq.com

Abstract. With the development of communication services, satellite network services have developed from the traditional single type of services to multiple types of services. Different types of services need to meet different Quality of Service (QoS) requirements. Traditional satellite network algorithm does not distinguish the services, so the high QoS service in the network cannot get timely service. However, the architecture of Software Defined Satellite Networking (SDSN) can solve this problem well. The controller in the SDSN has a global view and can obtain the node and link information of LEO satellites. Therefore, the SDSN can more effectively meet the demand of data service for QoS. In this paper, a satellite network architecture based on Software Defined Networking (SDN) is firstly established. Then an adaptive routing algorithm that can meet various QoS requirements is proposed. The final simulation results show that compared with SDRA algorithm and Dijkstra algorithm, this optimization algorithm has better performance in QoS satisfaction rate and bandwidth utilization.

1. Introduction

Software Defined Networking (SDN) [1,2] is a new network architecture. If combining satellite network with SDN, LEO satellite node can only complete the data forwarding function. The calculation of routing strategy and specific routing in the network is realized by GEO controller. Software Defined Satellite Networking (SDSN) [3] can effectively solve such problems as inflexible configuration update and complicated management of traditional satellite network systems, and improve the performance and QoS of end-user services [4,5].

Due to the characteristics of SDN centralized management mode, the common distributed routing algorithm in satellite networks is no longer applicable to SDSN. The study of routing algorithm has become one of the key technologies in SDSN. The business of satellite network has diversity, different business has different requirement to throughput, packet loss rate or delay. Therefore, Satellite networks need adaptive routing processes that support different Quality of Service (QoS) requirements, rather than simple best effort transmission [6,7]. In the literature [8], Guo A et al. applied SDN technology to LEO satellite network and proposed A centralized shortest path algorithm for LEO satellite network based on SDN. In literature [9], Zhu Y et al. proposed a software-defined routing algorithm (SDRA). This algorithm is a centralized routing algorithm. By collecting the satellite link information in real time, the SDN controller can avoid choosing the congested link during the route calculation, so as to obtain the optimal route path and achieve a better flow distribution. However, they only take a single QoS objective as the optimization point, and can’t adaptively meet the requirements of different service transmission quality.
In order to meet the transmission requirements of different space tasks, this paper proposes a Multi-QoS Adaptive (MQA) routing algorithm for routing packets under multiple QoS constraints. In this paper, an optimization model with multiple QoS constraints including bandwidth, delay and packet loss rate is established. The Lagrange relaxation method \cite{10} is used to relax the multi-constraint problem, and a new link weight formula is obtained. Then a dynamic step size is introduced into the weight factor. By judging whether the selected path meets the demand of QoS, the proportion and influence of time delay and packet loss rate in link weight are changed. After that, the optimal path satisfying the demand of multi-QoS is found through gradual iteration and rapid convergence.

2. Network architecture and problem description

SDN consists of application layer, control layer and infrastructure layer. Its network architecture is shown in figure 1. SDN adopts the mode of centralized control surface and distributed forwarding surface. SDN controller has global view, it enables centralized control of network devices in the infrastructure layer through the southbound interface in the network, while providing flexible programmable capabilities \cite{1,2}.

![Figure 1. The architecture of SDN.](image)

In SDSN, various spatial tasks constitute the application layer of the logical architecture. The control layer is mainly composed of GEO satellite nodes. GEO controller generates a global network view by obtaining node information and link information of all LEO satellites in the entire network. At the same time, the network status information is synchronized between multiple controllers. The infrastructure layer consists of LEO satellite nodes and is mainly responsible for data processing and forwarding \cite{11}. By analyzing the state information of the satellite network, the controller can select the optimal routing path. Through centralized management, the SDSN architecture can optimize routing tables in real time, deploy fine-grained management strategies, and improve the QoS performance of the network. In addition, the satellite nodes will be no different from other common switching nodes. Since the similarities and differences brought by different networks can be unified through modularization, the integration of multiple networks becomes easier.

3. System modeling and algorithm analysis

3.1. Multi-constraint model

When SDN controller chooses the path for the packet, it first needs to satisfy the constraint of QoS and then select an optimal path. According to the classification of space tasks, the following three QoS can be taken as optimization objectives to establish a multi-constraint model:
$$\min \sum w(i,j), (i,j) \in \text{path}(s,d) \quad \text{s.t.}$$

\begin{align}
\text{Remaining bandwidth:} & \quad \min(\text{band}(i,j)) \geq B \\
\text{End-to-end delay:} & \quad \sum \text{delay}(i,j) \leq D \\
\text{Packet loss rate:} & \quad \prod \text{loss}(i,j) \leq L
\end{align}

Where, \((i,j)\) represents the link between node \(i\) and target node \(j\), \(w(i,j)\) represents the link weight of link \((i,j)\), \(\text{delay}(i,j)\) represents the end-to-end delay of link \((i,j)\), \(\text{loss}(i,j)\) represents the link loss rate \((i,j)\), and \(B\), \(D\) and \(L\) respectively represent the space task’s requirements for multi-QoS. The constraint of formula (2) ensures that the path meets the basic requirements of QoS.

### 3.2. Multi-constraint problem relaxation

First, it is necessary to take the logarithm of both sides of the inequality and convert the multiplication in (2) into addition,

$$\sum \log[\text{loss}(i,j)] \leq \log L$$

To solve the above multi-QoS constraint problem, Lagrange relaxation method is required to relax the constraint conditions of (2), and the results are shown as follows:

$$L(\alpha, \beta) = \min[\sum w(i,j) + \alpha(\sum \text{delay}(i,j) - D) + \beta(\sum \log \text{loss}(i,j) - \log L)]$$

$$= \min[\sum w(i,j) + \sum \text{delay}(i,j) + \sum \beta \log \text{loss}(i,j) - \alpha D - \beta \log L]$$

$$= \min[\sum (w(i,j) + \alpha \text{delay}(i,j) + \beta \log \text{loss}(i,j)) - (\alpha D + \beta \log L)]$$

\text{s.t. } \min(\text{band}(i,j)) \geq B, (i,j) \in \text{path}(s,d)

Because when \(\alpha\) and \(\beta\) are constants, \(\alpha D + \beta \log L\) is also a constant. In order to facilitate calculation, the solution \(L(\alpha, \beta)\) in the above equation can be simplified to solve

$$\min \sum (w(i,j) + \alpha \text{delay}(i,j) + \beta \log \text{loss}(i,j))$$

According to the above analysis, link weight can be redefined as

$$W(i,j) = w(i,j) + \alpha \text{delay}(i,j) + \beta \log \text{loss}(i,j)$$

So, we can convert this problem to solving for \(\min \sum W(i,j)\). Therefore, the most important thing to solve this problem is how to calculate \(\alpha\) and \(\beta\).

### 3.3. Optimization Model

Through the analysis of \(W(i,j)\), it can be found that with the gradual increase of weight factor \(\alpha\) and \(\beta\), the proportion of time delay and packet loss in \(W(i,j)\) will also gradually increase. Therefore, we can think of \(\alpha\) and \(\beta\) as both monotone increasing functions. At the same time, considering that the traditional linear programming problem is solved iteratively by gradient method, we can use this method to solve the optimal path in the network. Partial derivatives of function pairs and are calculated as follows: Partial derivatives of \(\alpha\) and \(\beta\) in function \(L(\alpha, \beta)\) are calculated as follows:

$$\frac{\partial L(\alpha, \beta)}{\partial \alpha} = \sum \text{delay}(i,j) - D, \quad \frac{\partial L(\alpha, \beta)}{\partial \beta} = \sum \log \text{loss}(i,j) - \log L$$

Since the values of equations (6) and (7) may be negative, the following equation is used to calculate partial derivatives, denoted as follows:

$$\Delta_{\text{delay}} = \max\{\frac{\partial L(\alpha, \beta)}{\partial \alpha}, 0\}, \Delta_{\text{loss}} = \max\{\frac{\partial L(\alpha, \beta)}{\partial \beta}, 0\}$$
In order to meet the above conditions and make the weight factor increase monotonically, it is necessary to set:

\[ \alpha \leftarrow \alpha + \alpha \Delta_{\text{delay}}, \beta \leftarrow \beta + \gamma \Delta_{\text{loss}} \]  

(9)

Through analysis, weight factors \( \alpha \) and \( \beta \) are obtained, respectively representing the \( t \)-th iteration. Therefore, we can get the result of \( (t+1) \)-th iteration updated as follows:

\[
\alpha_{t+1} = \alpha_t + \sigma (\sum w(i, j) - C_{\text{min}} + 1), \\
\beta_{t+1} = \beta_t + \lambda (\sum w(i, j) - C_{\text{min}} + 1), \\
\sigma_t = \left( \sum \text{delay}(i, j) - D \right)^y + \left( \sum \text{loss}(i, j) - \log L \right)^y, \\
\lambda_t = \left( \sum \text{delay}(i, j) - D \right)^y + \left( \sum \text{loss}(i, j) - \log L \right)^y
\]

(10)

(11)

Where \( C_{\text{min}} \) is the path minimum weight not constrained by QoS. The scalars \( \pi_t \) and \( \lambda_t \) are continuously updated nonnegative constants. \( \left[ \frac{t}{3} \right] \) is the value of rounded to the nearest integer.

\[
\pi_t = \frac{C_{\text{min}}}{D}, \pi_t = \frac{\pi_t}{2}, \lambda_t = \frac{C_{\text{min}}}{\log L}, \lambda_t = \frac{\lambda_t}{2}
\]

(13)

The optimal solution of the original multi-QoS constraint problem can be obtained by using the above method. If the delay or loss rate of the previous path cannot meet the flow demand, the corresponding weight factor can be increased to improve their proportion in the weight of the total link. In this way, the feasible path can be searched through step by step iteration and the constraint condition of formula (2) can be determined.

4. Analysis of simulation results

4.1. The simulation setup

The performance of MQA algorithm is simulated in satellite network. The satellite tool kit (STK) was used in this paper to simulate the network scene, and then the MQA algorithm was simulated by MATLAB.

The simulation environment takes the network scene shown in figure 2 as an example. The system consists of 4 GEO controller and 64 LEO satellites. As the controller, GEO satellite is responsible for monitoring LEO network and obtaining the network topology diagram for routing packets. LEO satellites are responsible for forwarding data streams based on stream table entries.

**Table 1.** The parameters of the system simulation.

| Parameters                  | Value  |
|-----------------------------|--------|
| GEO controller              | 4      |
| LEO satellite               | 64     |
| Number of Orbits            | 16     |
| Number of Satellite/Orbit   | 16     |
| ISL                         | 2 intra/2 inter |
Figure 2. The physical architecture of SDSN.

LEO satellite constellation has a total of four satellite orbits, each of which contains four satellites, and each satellite is connected to its neighboring satellites through inter-satellite link. Inter-satellite links can be divided into two categories: inter-satellite links within orbit and inter-satellite links between orbits \cite{12}, with 12 links respectively. Specific parameters are shown in Table 2.

Table 2. The parameters of the system simulation.

| Parameters     | Value    |
|----------------|----------|
| ISL Bandwidth  | 100 Mbps |
| Inter-ISL Delay| 5ms      |
| Inter-ISL Loss rate | 1% |
| Intra-ISL Delay | 8ms      |
| Intra-ISL Loss rate | 1.5% |

In the simulation scenario, the GEO satellite controller collects node link information in LEO network, and the link bandwidth is configured as 100 Mbps, the average inter-satellite link delay is set as 5ms, and the packet loss rate is set as 1%. The average time delay of inter-orbit satellite link is set as 8ms, and packet loss rate is set as 1.5%. Among them, the inter-link delay is composed of transmission delay, processing delay and queueing delay. Assume that the bandwidth occupied by each transmission is randomly set to 1-10 Mbps, and the requirements of the space task on delay and packet loss rate are set to 30ms and 5%, respectively.

4.2. Analysis of simulation results

Firstly, MQA algorithm is compared with SDRA algorithm and Dijkstra algorithm in QoS satisfaction. Figure 3 compares the QoS satisfaction of the three algorithms. The figure shows that the QoS satisfaction rate of Dijkstra algorithm is the minimum, because when searching the shortest path of data stream, Dijkstra algorithm only chooses the path with the shortest link weight without considering the requirements of delay and packet loss rate. When the bandwidth of a link in the path is completely occupied, the subsequent data cannot be transmitted. The SDRA algorithm bypasses the congested inter-star link when calculating the routing path, so its bandwidth satisfaction is good, but it does not consider the influence of delay and packet loss rate, which are slightly lower. MQA has the highest QoS satisfaction rate among the three algorithms, because it comprehensively considers the requirements of various qualities of service to select the optimal path. MQA algorithm improved 28% compared with SDRA algorithm in terms of delay satisfaction and packet loss rate satisfaction, and
100% compared with Dijkstra algorithm in terms of delay satisfaction, packet loss rate satisfaction and bandwidth satisfaction.

![Figure 3. QoS satisfaction ratio of the three algorithms.](image)

Next, MQA algorithm is compared with SDRA algorithm and Dijkstra algorithm in bandwidth utilization. Figure 4 shows the relationship between bandwidth utilization and business strength of the three algorithms. When the traffic intensity is less than 20, the bandwidth utilization of the three algorithms is almost the same, because the three algorithms are based on the shortest routing algorithm, giving priority to routing the data stream to the path with the least link weight. When the traffic intensity is greater than 20, the bandwidth utilization of Dijkstra algorithm does not increase, because Dijkstra algorithm only chooses one shortest path. In comparison, MQA algorithm and SDRA algorithm have always shown better performance. When the traffic intensity reaches 45, the bandwidth utilization of MQA algorithm is the same as that of SDRA algorithm, which is much higher than that of Dijkstra algorithm.

![Figure 4. Bandwidth utilization of the three schemes in relation to traffic intensity.](image)
5. Conclusion
In this paper, the MQA routing algorithm suitable for software defined satellite networking is proposed. The algorithm can search a shortest path under the constraints of bandwidth, delay and packet loss rate. Firstly, a multi-QoS constrained optimization model was established, and the Lagrange relaxation method was used to optimize the model. The weight factors of delay and packet loss rate were dynamically updated, and the optimal path was searched iteratively through effective feedback of the historical search results. The research results show that MQA algorithm improves performance by 100% compared with Dijkstra algorithm in QoS satisfaction, and by 28% compared with SDRA algorithm in terms of time delay satisfaction and packet loss rate satisfaction. In terms of bandwidth utilization, when the number of traffic intensity reaches 45, the performance of MQA algorithm is improved by 76% compared with Dijkstra algorithm.

6. Acknowledgments
This work was financially supported by the National Natural Science Foundation of China under Grant 61701522 fund.

References
[1] Nunes B A A, Mendonca M, Nguyen X N, et al. A Survey of Software-Defined Networking: Past, Present, and Future of Programmable Networks[J]. IEEE Communications Surveys & Tutorials, 16 (2014) 3 1617-1634.
[2] Feamster N, Rexford J, Zegura E W. The road to SDN: an intellectual history of programmable networks[J]. ACM SIGCOMM Computer Communication Review, 44 (2014) 2 87-98.
[3] Jinzheng Bao, Baokang Zhao, Wanrong Yu, et al. OpenSAN: A Software-defined Satellite Network Architecture[J]. Acm Sigcomm Computer Communication Review, 2014.
[4] Gopal R, Ravishankar C. Software Defined Satellite Networks[C]/ Aiaa International Communications Satellite Systems Conference. 2013.
[5] Bertaux L , Medjiah S , Berthou P , et al. Software defined networking and virtualization for broadband satellite networks[J]. IEEE Communications Magazine, 53(2015) 3 54-60.
[6] Tomovic S , Radusinovic I , Prasad N . [IEEE IEEE EUROCON 2015 - International Conference on Computer as a Tool (EUROCON) - Salamanca, Spain (2015.9.8-2015.9.11)] IEEE EUROCON 2015 - International Conference on Computer as a Tool (EUROCON) - Performance comparison of QoS routing algorithms applicable to large-scale SDN networks[J]. 2015:1-6.
[7] Meng X, Wu L, Yu S, et al. Multi-Topology Routing Algorithms in SDN-Based Space Information Networks[J]. Future Internet, 11(2019)1.
[8] Guo A, Zhao C, Xu F, et al. LEO satellite routing algorithm in software defined space terrestrial integrated network[C]. International symposium on communications and information technologies, 2017: 1-6.
[9] Zhu Y, Qian L, Ding L, et al. Software defined routing algorithm in LEO satellite networks[C]. international conference on electrical engineering and informatics, 2017: 257-262.
[10] Liu K , Cao Y , Liu Y , et al. A novel min-cost Qos routing algorithm for SDN-based wireless mesh network[C]/ IEEE International Conference on Computer & Communications. IEEE, 2017.
[11] Bi, Yuanguo, Han, Guangjie, Xu, Shuang, et al. Software Defined Space-Terrestrial Integrated Networks: Architecture, Challenges, and Solutions[J]. IEEE Network, 33(1):22-28.
[12] DAI Cuiqin, YIN Xiaopan. Ant colony optimization based probabilistic routing algorithm in satellite networks[J]. Journal of Chongqing University of Posts & Telecommunications, 2018.