A Routing Model Based on Multiple-User Requirements and the Optimal Solution

SHENG LIU, DI WU, AND LANYONG ZHANG, (Member, IEEE)
College of Intelligent Systems Science and Engineering, Harbin Engineering University, Harbin 150001, China
Corresponding author: Di Wu (wuxiangpengheu@163.com)

ABSTRACT To promote the development of high-bandwidth IP networks of nongeostationary Earth orbit (NGEO) satellites, this article designs a new benefit measurement model. Moreover, we propose the class-A QoS benefit criterion (QABC), which is established based on a mathematical model involving the remaining queue length and QoS. The benefit model includes three maximum and minimum conflict subgoals and is designed to avoid the problem that the ant colony algorithm easily falls to local optima. This new model is solved with the wolf colony algorithm, which provides both next-hop selection and bottleneck bandwidth reservation mechanisms. Additionally, many burst flows can occur in satellite networks, and they lead to a slow convergence speed and unnecessary overhead. In this article, the Kalman filter algorithm is used to address these issues. Considering the long-term correlation of satellite traffic, it is feasible to smooth the queue length with a Kalman filter. Finally, the Kalman filter wolf colony algorithm (KFWCA) can solve the conflict problem in the new benefit measurement model. The algorithm is applied in the network simulator NS2.35, and the results verify the effectiveness of load balancing and QoS maximization in a satellite network. The results indicate that the KFWCA yields better performance than other algorithms based on traffic allocation, the average delay, the packet loss rate and other factors, especially under high-traffic conditions.

INDEX TERMS Benefit measurement model, KFWCA, QoS support, load balancing.

I. INTRODUCTION

With the increasing demand for satellite technology and communication requirements in near-Earth orbit [1], it is necessary to provide global Internet services. Ethernet, ZigBee and other types of networks are widely used. Moreover, the concepts of the Internet of Things (IoT) and cloud computing are the focus of many on-Earth networks, but both technologies increase the network burden and have limitations due to the lack of cloud access in remote areas [2]. A large number of advanced and reliable network architectures cannot achieve interconnection. Therefore, by using satellites as an important link for cross-regional connections between the ground, air or ocean, satellites can help to expand the ground cloud platform or 5G communication range [3]. In addition, different ground areas have different mission requirements, so the design of satellite routing strategies based on different task requirements is the current research focus. Additionally, with the development of software definition networks (SDNs) and network function virtualization (NFV), network control centers (NCCs) have been configured with reasonable LEO satellite routing algorithms to meet the requirements of different tasks. Overall, the main objective and contribution of this article is to design a satellite routing model and a corresponding optimal solution algorithm.

To achieve effective data transmission based on satellite routing, it is necessary to consider the unbalanced distribution of global IoT traffic [4]. Mountain, ocean and alpine regions tend to have lower traffic demand than other areas, and the demand is closely related to both the degree of development in a region and the environmental characteristics. In addition, the link connectivity or information accessibility among satellites is determined by the constellation topology and has very strong regularity. The shortest path, minimum delay or maximum communication time goals are important considerations in solving satellite routing problems [5]. However,
a satellite may underutilize the nodes in noncongested areas, thus limiting the available resources of some nodes; as a result, communication may remain at a low level, with the loss of a large number of data packets occurring.

Considering the above difficulties, scholars have proposed the concept of load balancing. The existing routing methods can be mainly divided into three types: node-enabled, global-enabled and optimization algorithm-enabled routing. Global-enabled routing was performed in reference [6] to determine the current state of all nodes through satellite links, and this information was used as the basis for bandwidth allocation and route decision making. However, this approach reduced the possibility of a node entering a congested area state at the cost of a low convergence speed and high expenses. In node-enabled routing, traffic allocation is only based on the instantaneous state of the node itself and the information in one hop, as noted in reference [70]; this approach guarantees that the strategy has a faster convergence speed and lower processing costs than other methods, but it will inevitably lead to the satellite network falling to the local optimal path. The third type of methods involves routing based on an optimization algorithm. In reference [8], a routing strategy based on an artificial fish swarm algorithm was designed for global searches while iteratively determining the optimal path. Some common solutions are the ant colony algorithm [9], particle swarm algorithm [10], etc. Although these routing algorithms have some shortcomings related to the long calculation time, these issues can be avoided by establishing an ideal routing cost function or adopting a method for satellite queue length prediction. This concept of integration and complementarity is one of the innovations of this article.

Load balancing combined with prediction technology is also a popular research topic at present. Many studies have shown that satellite orbits have a certain operational trajectory, and the distribution of ground-based networks is characterized by predictability and periodicity throughout the year. Based on these concepts, Ref. [11] established a routing policy based on a realistic physical model. Reference [12] designed a routing strategy for sudden traffic, but unpredictable random traffic will not have a serious impact on satellite networks. If SDSN-enabled switches are updated based on large random flows from the ground terminal, the subsequent overreaction may lead to a slow convergence speed and increased calculation costs [13]. The self-similarity of ground traffic has been widely verified and cannot be completely eliminated with network topologies and service types. Even though self-similarity will be reduced slightly after transmission through a satellite gateway, this characteristic still cannot be ignored [14]. Therefore, on the basis of analyzing the time-varying satellite topology, it is particularly important to consider the traffic characteristics of self-similarity. Self-similarity can be analyzed as a statistical long-term correlation; that is, sudden traffic will be absorbed by a large number of streaming media, which leads to a smoothed streaming media traffic curve. Additionally, the traffic sequence at any time can be expressed as a function of time periods. In this article, the unique long-term correlations in a satellite network are used to predict the remaining queue length.

With the increasing complexity of ground services in the SDSN, QoS support is also the focus of this article. Reference [15] classified businesses with different delay requirements and developed a method to determine whether to bypass the original, take a detour or maintain the original route according to specific network states; to some extent, this approach guarantees the quality of service of high-level streaming media. However, reference [16] noted that other satellite orbit operation states will result in poor quality of service due to the dispersion of traffic. In this article, the QoS guarantee mechanism is implemented in the LEO layer, which does not affect the communication capacity of the MEO/GEO layer. We consider the QoS level in the benefit model, and a mathematical model of different services that change with the remaining queue length is established. Then, a comprehensive evaluation index is designed to guarantee the service quality of satellites and effectively avoid congestion.

The structure of this article is as follows. Section I introduces the motivation of this article, which is to design satellite routing technology. Section II analyzes the existing works. Section III establishes the LEO method and the benefit measurement model, which are designed to ensure QoS and load balancing. In Section VI, the Kalman filtering wolf colony algorithm (KFWCA), in which Kalman filtering is used to eliminate burst traffic and the wolf colony algorithm is used to determine the optimal path, is described. In Section V, the feasibility of the KFWCA is demonstrated in application involving satellite networks. Section VI provides a summary of the article.

II. RELATED WORK

At present, space networks include tens of thousands of satellites, and the movement direction of each satellite and number of adjacent satellites are determined in advance. These networks provide services around the Earth based on certain rules, and the corresponding physical models generally consider the virtual topology and virtual nodes [17]; such models are static processing methods that ignore some dynamic characteristics. In reference [18], the link condition was studied by using a time interval. The satellite topology in each time slot was assumed to be static, and the NCC was used to calculate the routing strategy in advance through a static structure and transfer it to the satellite data layer through a control link. This method greatly reduces the algorithm complexity compared to that of traditional algorithms, but it is not suitable for the sudden failure of satellites or outage maintenance. Under the condition that the inter/intrasatellite link is stable, the virtual node method was used in reference [19] to divide a satellite into different parts according to its orbit; this approach is equivalent to establishing a static satellite motion map. The dynamic characteristics of the satellite in each region
are ignored, and routing calculations are performed; only the virtual nodes in the undirected satellite graph are considered. In this article, the LEO satellite model is built on this basis, and each satellite is numbered, which effectively avoids rerouting due to a large number of dynamic changes and reduces the number of computations for the on-board core processor.

Improve satellite network performance. Table 1 summarizes the existing routing strategies. In reference [20], a routing rule combining deep searching and the Dijkstra algorithm is proposed. SDN and NGEO satellites are associated in the structure to improve the system monitoring management mode and the calculation efficiency. In reference [21], a prior geographic information-based HLBR strategy was proposed for real-time sensing to avoid congestion through satellite detour path calculations. This algorithm is based on the concept of long-distance detours, which result in unsatisfactory adaptive ability and poor predictions. Therefore, if a reasonable algorithm can be designed to predict the flow state in the next stage and mitigate the unreliability caused by time delays, the satisfaction of multidemand users can be greatly improved. In reference [22], by calculating the queue length at the next moment, the system can improve the comprehensive service level of a satellite network and achieve a certain fault diagnosis ability. If the traffic offset of a satellite obviously exceeds the preset threshold value in a given period of time, it can be preliminarily assumed that there is unreliable threat information in the network; this scenario is related to congestion prediction. In reference [23], the FARIMA prediction model was studied considering its nonlinear potential characteristics, but its limitations are reflected by approximate processing in the fitting calculation. Reference [24] verified the feasibility of the wavelet algorithm for prediction; this process involves the decomposition of high-frequency and low-frequency filters, but the corresponding algorithm needs to adopt different wavelet basis functions for different signal waveforms. Therefore, it is difficult to achieve the automatic control of nonhuman intervention. In reference [25], deep learning was used to predict the next hop; this method can not only avoid the complexity of object models but also provide a controller with online learning ability. However, the complexity of the network structure is directly related to the prediction convergence speed, so it is difficult to apply certain methods to communication networks with strict delay requirements and limited computing capacities. Kalman filter technology is applied for the prediction of satellite traffic in this article. Notably, the real-time monitoring queue length of a satellite is combined with the prediction error at the previous time step based on a certain proportional relation. In addition, measurement noise is added to the prediction error to ensure that each satellite node has a smooth and predictable queue length.

In reference [26], with a single-layer satellite network architecture, a dynamic path adjustment routing strategy TLR based on a street light indicator was proposed. The congestion of two nodes along a link was monitored at each satellite output port, and the traffic conditions for the buffered data packets were dynamically adjusted by color changes. Moreover, multiblock technology was used for the temporary storage of excessive data packets. However, the differences in consumer QoS is not considered in that routing strategy. In reference [27], a traffic load optimization algorithm suitable for a multilayer satellite network was established. For a random traffic flow, the original routing table was adjusted twice; moreover, two optimal thresholds and a transmission droop ratio were established to match the load balancing demand. In reference [28], a delay-oriented adaptive routing strategy called DOAR was designed to flexibly adjust the QoS through an information backtracking mechanism. In reference [29], aiming at the network topology of NGEO, a routing indicator based on fuzzy control theory was designed. In the algorithm, multiple kinds of QoS requirements were defined as corresponding fuzzy functions, and local fuzzy rules from the neighboring satellites were combined to obtain a global fuzzy decision. The experiment showed that the algorithm has better congestion handling ability under high-traffic conditions than for other conditions. With the continuous improvements in the routing strategies of ground-based wireless networks, intelligent optimization algorithms such as the bee colony algorithm, ant colony algorithm and particle swarm algorithm have been extended to topological optimization and routing decision making for satellite networks in recent years [8]–[10]. Reference [30] verified the application value of the ant colony algorithm in single-layer NGEO. The algorithm aims to determine the shunting probability of the next routing node through the superposition and release of pheromones. Through the dynamic exploration and feedback mechanisms of ants, the optimal link path satisfying the constraints is continuously optimized. Additionally, many studies have verified that the wolf colony algorithm can solve
combined optimization problems, so its application in routing is feasible.

The contributions and motivation of this article are as follows. To improve the intelligence of the LEO constellation routing strategy, this article establishes a mathematical model of service satisfaction that varies with the remaining queue length, and a new benefit model that improves the comprehensive decision-making abilities in QoS and traffic allocation problems is designed. Moreover, to balance the maximum and minimum conflicts of the three subobjectives, the KFWCA algorithm is proposed, thereby ensuring that satellite routing paths are optimal.

\[ \text{FIGURE 1. LEO network model with virtual topology numbers.} \]

III. LEO MODEL AND NEW BENEFIT FUNCTION

A. SYSTEM MODEL

The system model in this article is based on an iridium-like constellation. As shown in Figure 1, there are four intersatellite links around each satellite, and each link is distributed in the adjacent orbits and original orbit with the structure of an approximate cross. However, the links only maintain connectivity at nonpolar locations. Suppose that there are \( M \times N \) LEO satellites in the orbit model and that they are distributed in \( M \) orbits with certain latitude and longitude angle intervals; additionally, there are \( N \) satellites in each orbit. If the logical position of a virtual node is represented by \((m, n)\), the precise latitude and longitude of each satellite can be expressed as:

\[
\begin{align*}
\Theta (m, n) &= \begin{cases} 
(\omega \times t) \% \Delta \Theta + \Theta_0 (m) + n \Delta \Theta, \\
2 \leq n < N/2 - 1 \\
(\omega \times t) \% \Delta \Theta - 180^\circ - \Theta_0 (m) + n \Delta \Theta, \\
N - 1 \geq n \geq N/2 + 1
\end{cases} \\
\Omega (m, n) &= \Omega_0 + m \Delta \Omega
\end{align*}
\]

(1)

where \( \Delta \Omega = \frac{180^\circ}{M} \), \( \Delta \Theta = \frac{360^\circ}{N} \), \( \Theta_0 \) is the longitude value of the initial satellite, and \( \Theta_0 (m) \) represents the first satellite in orbit \( m \). When \( m \) is odd, \( \Theta_0 (m) = \Theta_1 \in (90^\circ, 90^\circ - \Delta \Theta/2) \); when \( m \) is even, \( \Theta_0 (m) = \Theta_2 \cap |\Theta_2 - \Theta_1| \leq \Delta \Theta/2 \). If the moving speed at the poles is too high, the direction of the satellite will considerably change after passing through the center of the poles. Therefore, this article assumes that communication will be temporarily cut off at high latitudes for the \( 1, N/2, N/2 + 1, N \) satellites, and the \( M \times N \) satellite matrix corresponding to the selected virtual node is used for satellite numbering; the formula is as follows:

\[
\begin{align*}
\text{int} \left( \frac{\omega \times t}{d} |(m_0, n_1) - (m_0, n_0)| \right) &\in [i, i + 1] = n_i \\
\text{SID} &= n_i + (m_i - m_0) N
\end{align*}
\]

(2)

where \( n_i \) indicates that the logical position of a satellite in the same orbit moving at angular speed \( \omega \) remains unchanged in the process \([i, i + 1]\) and the SID representation greatly simplifies the satellite changes in longitude and latitude. According to the numbering principle in this article, every node can be recorded in the routing table, and the existence of a loop can be effectively avoided.

B. BENEFIT MEASUREMENT MODEL BASED ON A QoS GUARANTEE AND LOAD BALANCE

In this section, a new ISL benefit measurement model is presented. The existing network lacks a comprehensive measurement model, so even if the algorithm is excellent, it cannot meet various requirements. In some studies, an improved method is proposed to define the cost measurement by combining queuing and propagation delays. However, this measurement model does not fully consider the needs of different QoS services. Even when the delay remains good, a phenomenon occurs in which the priority of delay-sensitive business is lower than that of non-real-time business. Based on this difference, this section proposes a new objective model to define the cost measurement by combining propagation delay, queue delay, QoS requirements and load balancing four parts.

1) MATHEMATICAL MODEL FOR MULTISERVICE QoS SUPPORT

The existing literature usually adopts the following mechanism to ensure QoS: each node’s output port establishes a buffer in four directions, and the buffer is subdivided by multiple service types. Then, the optimal path is selected by setting the threshold based on the service types. However, there are two disadvantages of this approach: there are no mathematical models that can provide theoretical support for the QoS guarantee mechanism for all types of services, and the algorithm can only determine the QoS according to a single empirical threshold.

Therefore, this article establishes several mathematical models for multiservice QoS under different remaining queues to improve the theoretical level of guaranteed QoS routing. In this article, we define the impact of different remaining queue lengths \( q_{res} \) on the link efficiency \( u(q_{res}) \) with a mathematical model of multiservice satisfaction, as shown in Figure 2. To the best of our knowledge, this is the first study that has normalized the output values of three
different services to produce an index, namely, the class-A QoS Benefit Criterion (QABC).

Class-A services have strict delay requirements, such as remote video and task emergency response requirements, and it is necessary to maintain a stable occupied bandwidth. If the length of the remaining queue of the LEO satellite cache mechanism is less than $q_{\min}$, the quality of this type of service will suddenly be distorted; moreover, the QoS improvement is not obvious when the remaining queue length is excessive. Therefore, the QoS benefit and maximizing the load balance. Then, the QoS values of different services at time $t$ based on the satellite number SID can be described by $U_{SID}(t)$, where $SID = \{1, 2, 3 \cdots \} \in M \times N$, $i = \{A, B, C\}$. Therefore, the QABC proposed in this article can be defined as:

$$U_{SID}(t) = u_{SIDA}(t) + u_{SIDB}(t) + u_{SIDC}(t)$$  \hspace{1cm} (7)

With the change in the queue length and concave to convex transition of the plot of the guaranteed QoS constraint equation, it is impossible to use the enumeration method for analysis, so we innovatively designed a heuristic algorithm to solve the new objective model. It should be noted that the premise of this model is that even though the distribution of the traffic flow in different regions of the country is uneven, the proportions of different traffic flows are approximately the same. It is impossible to have only one business in a target region, as is the case in reality.

2) CONFLICT BENEFIT MEASUREMENT MODEL

The creative link benefit model designed in this article aggregates the functions of minimizing the delay, maximizing the QoS benefit and maximizing the load balance. Then, the weighted combined results for the three conflict sub-objectives can be described as follows:

$$\sum_{j \in P(s, d)} \sum_{i \in P(s, d)} ISL_{benefit_{i,j}}(t) = \sum_{j \in P(s, d)} \sum_{i \in P(s, d)} \mu_{i,j}(t) \left[-w_1 f_{1,i}(t) + w_2 f_{2,i}(t) + w_3 f_{3,i}(t)\right]$$  \hspace{1cm} (8)

$$f_{1,i}(t) = P_{T_{i,j}}(t) + Q_{T_{i,j}}(t)$$

$$f_{2,i}(t) = \frac{1}{N} \int_{\Delta}^{\Delta} \frac{q(a) \times \text{pack}}{c} \, da$$

The length of the LEO satellite cache does not significantly affect the QoS and maintains a high function value for a long time. The applied function is a strict convex function between the remaining cache and QoS. The following equations can describe the class C service:

$$u_{SIDC}(q_{res}) = \begin{cases} 0, & q_{res} < q_{\min} \\ 1, & q_{res} \geq q_{\min} \end{cases} \hspace{1cm} (3)$$

Through the deformation of the exponential function and the point of convexity change on the corresponding curve, the function parameters can be calculated by the following equation group:

$$\begin{cases} u_{SIDB}(q_{\max}) = 1 - e^{-\frac{q_{\max}}{q_{\max} + n}} \\ -mq_{\max}^2 (q_{\max} + 2n)^2 + 2n^2 (q_{\max} + n) = 0 \end{cases} \hspace{1cm} (4)$$

$$\begin{cases} u_{SIDC}(q_{\max}) = 1 - e^{-\frac{q_{\max}}{q_{\max} + m'}} \\ m' = -\ln \left(1 - u_{SIDC}(q_{\max})\right) \end{cases} \hspace{1cm} (5)$$
\[ f_2(t) = U_{SID}(t) = 2/3 - e^{-\frac{m_2(t)}{m_{max}}} - e^{-\frac{m_3(t)}{m_{max}}} \] maximizes the benefit value \( \text{max}QABC \). By using the function model of service satisfaction and the change in the remaining queue length, QABC maximization can ensure high-quality data transmission.

\[ f_3(t) \] maximizes the number of link path nodes, which indicates the number of relay nodes associated with one route from the starting node to the end node. The purpose of this function is to increase the number of relay satellites so that the NCC makes full use of remote and idle satellites in neighborhood regions. By recording the SID and the number of routing nodes, forwarding rules can achieve a load balance in the link objective function, and the loop problem can be effectively avoided.

\[ \mu_{i,j}(t) \] is the distribution proportion coefficient of global users and hosts. Considering the key congestion areas of global traffic centralized between 0° north latitude and 50° east longitude, some remote satellites have never been fully utilized. This operator is introduced to further ensure the load balance and make the description of the routing benefit model optimally reflect the actual situation. If the latitude value of the satellite SID at time \( t \) is represented by \( \text{Lat}_{SID}(t) \), then the route benefit metric can be modified by the coefficient \( \mu_{i,j}(t) \), and the corresponding value can be dynamically represented by the subsection function:

\[
\mu_{i,j}(t) = \begin{cases} 
1, & i = s \\
e^{(\text{Lat}_{SID}(t)/90)\circ}, & (0° < \text{Lat}_{SID}(t) < 50°) \cap i \neq s \\
e^{-|\text{Lat}_{SID}(t)/90|\circ}, & \text{else} 
\end{cases}
\] (9)

Assuming that a class-A traffic transmission has “high quality” or “low quality”. \( U_{SID}(t) \) can be represented by the binary variables “1” and “0”. Therefore, under the condition of unbalanced global traffic, the conflicting optimization model based on the total link delay and QABC is finally established as follows:

\[
\text{Maximum}: \sum_{i \in P(s,d) \cap j \in P(s,d)} ISL\text{benefit}_{i,j}(t) \\
\text{Subject to}: \sum_{i \in P(s,d) \cap j \in P(s,d)} f_{i,j}(t) \leq T_{\text{DELAYmax}} \\
\forall \text{ link } (i, j) U_{SID_{A}}(t) = 1 \cap f_{i,j}(t) \in [0, 1) \\
\cap f_{3}(t) \in (1, M \times N) \\
w_i > 0 \cup \sum_{i=1}^{3} w_i = 1
\] (10)

The new objective function established in this article is a complex optimization problem, which makes it difficult to perform online computing. Reference [30] verified that the ant colony algorithm has some advantages in solving such functions, but the pheromone transfer and updating method has a slow convergence speed and easily falls to the local optimal path, making it unsuitable for the conflict benefit model proposed in this article. Therefore, KFWCA is designed to find the optimal solution, thus guaranteeing the reliability and robustness of complex multidimensional and multipeak functions.

IV. SATELLITE ROUTING STRATEGY BASED ON THE KALMAN FILTER WOLF COLONY ALGORITHM

The algorithm designed in this article is innovative and achieves better routing results through global searching than do traditional methods. If sudden traffic is observed in a network, the emergent data streams will lead to unnecessary changes in the optimal strategy, resulting in frequent route updates and unnecessary overhead. Therefore, this article adopts the Kalman filter algorithm to smooth the traffic flows, which overcomes the long convergence time limitation of the wolf colony algorithm to a large extent. We use the traffic prediction results as the input value of the wolf swarm algorithm, and the overall performance of the system can be improved.

A. SATELLITE TRAFFIC PRETREATMENT

The overall flow of the KFWCA algorithm is shown in Figure 3, and the designed routing mechanism and in-satellite flow process are specifically described in the next section. In this section, the smoothing process of satellite traffic is analyzed, and the traffic output value with smoothing characteristics is taken as the input signal of the subsequent routing process.

According to real data for the Internet traffic distribution in one week provided from reference [31], the data from a 24-hour period on a Monday are extracted for analysis, as shown in the red curve in Figure 3. The general form of the Kalman filter is complex and does not consider the specific case of satellite queue length prediction. In this section, the equations are defined, and new definitions of parameters are given. Therefore, five important equations for satellite queue length prediction are as follows.

1) SATELLITE QUEUE LENGTH PREDICTION

\[ \hat{q}_k = \hat{q}_{k-1} \] (11)

Represents the predicted value of the current \( k \)-time satellite queue length, where “−” represents the predicted value, which is not the optimal result and can be directly replaced by the queue monitoring value at time \( k-1 \). Many existing studies only used this value to assess accuracy; furthermore, the predicted value is further analyzed to improve the performance of the original algorithm.

2) PREDICTION OF THE MEAN SQUARE ERROR OF THE QUEUE LENGTH PREDICTION

\[ P_k = \sqrt{P_{k-1}^2 + Q^2} \] (12)

where \( P_{k-1} \) indicates the predicted queue length error for a given satellite node at the last moment and \( Q \) is the
measurement noise caused by a random burst of traffic. Random traffic may occur between two sampling intervals.

In addition, the formula differs from the general form of the Kalman filter, but the meaning of the equation is the same.

3) KALMAN WEIGHT GAIN CALCULATION

\[
K_k = \sqrt{\frac{(P_k)^2}{(P_k)^2 + Q^2}} \in (0, 1) \tag{13}
\]

The calculation of weight gain is the key factor that affects the prediction accuracy of the satellite queue length and is the core parameter of this algorithm. The purpose of this equation is to determine the degree of influence of the satellite queue length based on the predicted and measured values and the appropriate coefficient weights.

4) THE OPTIMAL ESTIMATION EQUATION AT THE PRESENT K-TIME

\[
\hat{q}_k = \hat{q}_k + K_k (q_k - \hat{q}_k) \tag{14}
\]

where \(\hat{q}_k\) indicates the optimal estimation of the queue length at time \(k\) and \(\hat{q}_k\) is the self-similarity input value of the wolf colony algorithm. In addition, \(q_k\) is the monitoring value of the satellite queue at the current time. In this article, we only need to predict the length of the satellite queue at the next time step, so this equation is a simplification of that in the general Kalman filter.

5) ITERATION PARAMETERS AT TIME \((k+1)\)

\[
P_k = \sqrt{(1 - K_k) \times (P_k)^2} \tag{15}
\]

The optimization of the KFWCA is reflected in the recursion of the above formula. Each time the length of the satellite queue is predicted, the weight gain at the next time step is calculated from the previous prediction error. Therefore, the algorithm only needs the satellite to store the queue value, covariance and monitoring value from the previous time step; then, it can output accurate and smooth prediction curves.

A long-term correlation service is characterized by predictability, and the service provided can be statistically expressed by an approximate Pareto distribution. However, this assumption increases the uncertainty of the model, and the application effect is not as good as that for a simulation. Therefore, we employ a Kalman filter to optimize the queue length of the LEO satellite and obtain a smooth traffic curve without short-term correlation burst traffic. Ultimately, we take the output as the internal part of the satellite to update the routing scheme of each satellite through the KFWCA.

B. INTERNAL SATELLITE PROCESSING

The routing mechanism of the KFWCA is shown in Figure 4. The key components of the algorithm are the head-wolf satellite, the detected-wolf satellite and the fierce-wolf behavior. Fierce-wolf behavior means that data flow is transmitted between two middle head-wolf satellites. The purpose of the detected-wolf satellites is to continuously search for the node with the largest odor concentration. Once the node with the maximum concentration is found, the location of the middle head wolf is updated. The process from the previous section is utilized to pretreat the traffic, which ensures the smoothness of the queue length. Each time the superimposed odor value on path \(path_{c,d}^i\) is estimated, the probability of route selection is redetermined, and the subsequent route adjustment task is completed.

Each satellite in the LEO layer is initialized with a directed graph \(G(V, E, O)\), where \(V, E, and O\) represent the SIDs of satellites, the link topology constraints, and the concentrations between two satellites.
1) ODOR CONCENTRATION OF PRE
The definition of the smell concentration along each candidate path is based on the benefit function value $ISL_{benefit_{x_{i1},x_{i2}}}(t)$ in Section 3; with this function, the delay demand is minimized, and the QoS support and load balancing are maximized. To resolve the conflict among the three subgoals, we treat the delay requirement as a negative value, so the maximum odor concentration can be identified. We define $X_i = \{x_{is}, x_{i1}, x_{i2}, \ldots, x_{ip}, x_{id}\}, p \in (1, M \times N - 2) \in Z$, where $X_i$ represents data for one completely available path collected by the target node; $x_{is}$ and $x_{id}$ are the source node and destination node, respectively; and $x_{ip}$ indicates the middle head-wolf satellite that this routing path passes through. Therefore, the optimal routing result, which is associated with the maximum link benefit function $Best - ISL_{benefit_{x_{is},x_{id}}}(t)$, can be obtained through odor concentration superposition.

2) HEAD-WOLF SATELLITE AND ITS BEHAVIOR
The head-wolf satellite ensures that the data are passed through as many idle nodes as possible in the local area, and its behavior can be described by the Dijkstra algorithm, which is first used to initialize the route. Then, the head-wolf satellite is randomly set on this path. The source node $s$ and the destination node $d$ are irreplaceable head wolves. The traffic for the satellites adjacent to the destination nodes is relatively concentrated. $T_{s \rightarrow d} = [T_{s \rightarrow i}, T_{i \rightarrow j}, \cdots, T_{k \rightarrow d}]$ is used to record the nodes that the route passes through. Furthermore, the odor concentration $\sum_{i \in P(s,d)} ISL_{benefit_{i,j}}(t)$ is stored in each node. It should be noted that a class-A service maintains the initial path to meet delay requirements, and class B and C services need to detour based on the algorithm in this article.

3) DETECTED-WOLF SATELLITE AND ITS BEHAVIOR
The objective of the detected-wolf satellite is to search for the nodes with the highest local concentrations, and its behavior can be described as follows. The nodes within one hop of the head-wolf satellite are regarded as the detected-wolf satellites. A detected wolf randomly searches for the largest benefit function values within a certain number of steps. The concentration formula is shown in (8).

If the detected-wolf satellite finds that the odor concentration at a certain node is higher than that at the current node, it will update the position of the middle head-wolf satellite. Additionally, the detected-wolf satellite needs to adhere to the maximum hop limit constraint. However, if the head-wolf satellite odor is the highest in a local area, the routing algorithm will not be adjusted. Moreover, if calculating the concentration, it is necessary to eliminate the poor-wolf satellites according to formula (10). The purpose of the above operation is to eliminate the satellite nodes that do not meet the QoS guarantee requirements under the specific link queue length; this process is similar to pheromone release along the worst path in the improved ant colony algorithm.

4) FIERCE WOLF AND ITS BEHAVIOR
The fierce-wolf concept is equivalent to data flow along a path. Through the fierce-wolf attack behavior, the route table can be established, and the odor concentration is the standard used to attract fierce wolves. The definition of parameters in this article is shown in Table 2. Inspired by the literature [32], the routing process for fierce wolves is as follows. Equation (16) indicates that the strategy is based on the
TABLE 2. Parameter definitions and descriptions.

| Symbol       | Definition and description                                      |
|--------------|----------------------------------------------------------------|
| $R_{v/d}$   | $R_{v/d} = [V/N]$, a rounding function to represent the row label of a source/destination |
| $C_{s/d}$   | $C_{s/d} = v \mod N$, a function to represent the column label of a source/destination |
| $N$         | $N$ the total number of satellites in the same orbit            |
| $S$         | $S = (R_c, C_c)$ the logical number of the row and column of the source node |
| $D$         | $D = (R_c, C_c)$ the logical number of the row and column of the destination |
| $\text{mind}(S, D)$ | the minimum number of hops between two nodes |
| $w_r, w_c$  | the row/column number difference between the start and end points |
| $P_{\text{next}}$ | the next-hop selection result                                  |
| $u_{\text{up}}(t)$ | the QoS guarantee of the next node for class-A service at time $t$ |
| $\sum_{j \in P_d(s)} f_{i,j}(t)$ | the time delay superimposed on the path $P(s,d)$ |
| $T_r$       | the update period                                               |
| $T_{\text{delaymax}}$ | the maximum delay                                               |
| $\text{SUM}_i$ | the odor concentration of $X_i$                                |
| $P_{(s,d)/i}$ | the probability of choosing $X_i$                              |
| ISLbenefit($t$)   | the odor concentration between two nodes                       |
| $I_{\text{BorC}}$ | the transmission rate of class B or C                           |
| $Q_{\text{max}}$ | the total buffer length                                         |
| $\delta$    | the capacity of a satellite                                     |
| $\delta + d_{\text{max}}$ | the transmission time                                           |

shortest path, and equation (17) gives the routing method.

$$\min d(S, D) = \begin{cases} WR + WC, \\ |R_i - R_d| + \min \{|C_i - C_d|, N - |C_i - C_d|\} \end{cases}$$ (16)

$$P_{\text{next}} = \begin{cases} (R, C + 1), \\ (WR > WC) \cap u_{\text{SIDA}}(t) = 1 \cap \sum_{j \in P_d(s)} f_{i,j}(t) \leq T_{\text{delaymax}}, \\ (R + 1, C), \\ (WR < WC) \cap u_{\text{SIDA}}(t) = 1 \cap \sum_{j \in P_d(s)} f_{i,j}(t) \leq T_{\text{delaymax}}, \\ (R - 1, C - 1), u_{\text{SIDA}}(t) = 0, \\ (R - 1, C + 1), \sum_{j \in P_d(s)} f_{i,j}(t) > T_{\text{delaymax}} \end{cases}$$ (17)

After the routing path is determined by equation (17), we need to overlay the odor concentration along path $X_i$ with equation (18). Then, equation (19) represents the probability of choosing $X_i$. Therefore, the destination satellite selects the path with the highest probability $\max \left( P_{(s-d)/X_i} \right)$ as the optimal result.

$$\text{SUM}_{X_i} = \text{ISLbenefit}_{s,X_i}(t) + \text{ISLbenefit}_{x_2,x_3}(t) + \cdots + \text{ISLbenefit}_{x_p,d}(t)$$ (18)

$$P_{(s-d)/X_i} = \frac{\text{SUM}_{X_i}}{\sum_{i=1}^{n} \text{SUM}_{X_i}}$$ (19)

where $T_c$ is the head-wolf satellite update cycle, $\text{ISLbenefit}_{s,X_i}(t)$ indicates the cumulative concentration value from the source node to the middle head-wolf satellite along path $X_i$; $\text{ISLbenefit}_{x_2,x_3}(t)$ indicates the concentration superposition value of two middle head-wolf satellites on path $X_i$; and $\text{ISLbenefit}_{x_p,d}(t)$ indicates the last concentration accumulation along path $X_i$. Because of the particularity of the source satellite, the position of this satellite does not change to ensure that data traffic is transmitted from the source node to various destinations.

In particular, if the two middle head-wolf satellites are at both ends of path $X_i$, $m$ and $n$ are the start and end nodes. Thus, the path is in a completely idle state, and all data flows do not need to detour to the nodes farther away. In the above case, we need to make the following judgment to determine the transmission of class B and C services:

$$I_{\text{BorC}}(t) = \begin{cases} 0, \\ \frac{q_{\text{min}} Q_{\text{max}}}{P_{\text{avg}}} + \left( I_{\text{BorC}}(t-1) - C \right) \delta + d_{m-n} \geq Q_{\text{max}} \\ I_{\text{BorC}}(t-1), \text{ else} \end{cases}$$ (20)

where $I_{\text{BorC}}(t)$ represents the transmission rate of class B and C services at the current time, $Q_{\text{max}}$ represents the total buffer length, $C$ is the capacity of a satellite, and $(\delta + d_{m-n})$ is the sum of the monitoring update cycle time and the shortest transmission time on path $X_i$. Equation (20) ensures that class B and C services must reserve the remaining $q_{\text{min}} Q_{\text{max}}$ for class-A service, even when the satellite network is not congested. If $Q_{\text{max}}$ is estimated to be exceeded at the next time step, path $X_i$ stops sending class B and C services immediately to ensure the quality of class-A service. Then, class B and C services search for a detour according to the odor concentration.

Finally, the next-hop selection algorithm and the bottleneck reservation mechanism designed in this article are used to transmit the information flow among the wolves with the highest local odor concentrations for class-A services. However, some of the detours are excluded due to the delay constraint, as shown in Table 3.

If the odor concentration value of a satellite is higher than that of the middle head-wolf satellite when tracking the target satellite, the middle head-wolf satellite can be replaced.
In this process, the initialization parameters can be adjusted to improve the global optimal effect of the proposed algorithm.

Finally, with the routing scheme in Table 3 as an example, the path adjustment result is shown in Figure 5.

Figure 5(a) represents the initial state. Figure 5(b), (c) and (d) represent the detour strategy developed in this article. In Figure 5(b), the green curve (class-A service) does not bypass any nodes to meet the corresponding delay requirement, the blue and yellow curves represent the detour paths of other types of services, and the red curves indicate infeasible paths. Figure 5(c) indicates that the optimal route is the initial path, but this path becomes congested at the next time step. Therefore, data transmission for class B and C services is interrupted to ensure the QoS of service A. Furthermore, the algorithm can also consider class B as a delay-sensitive service; then, only class C service is bypassed. Figure 5(d) shows the loop avoidance mechanism when there is a loop in the path.

### C. ALGORITHM STRATEGY DESCRIPTION

In this article, the KFWCA optimal path generation process is shown in **Algorithm 1**. The algorithm needs to determine whether the result meets the accuracy requirements or reaches the maximum number of iterations. The accuracy is defined by the probability of finally selecting a path. If the result does not meet the requirements, the number of wolf satellites will be reset, and the cycle will continue. Steps 1-4 are preparatory steps. Steps 5-6 are iterative constraints. Step 8 is the Kalman filter inside the satellite, which can eliminate instantaneous traffic issues. Steps 9-14 are the wolf colony algorithm, which is to determine all candidate paths. Steps 16-17 select the best path according to the odor concentration as the final result.

### V. SIMULATION VERIFICATION

#### A. EXPERIMENTAL PARAMETER CONFIGURATION

The KFWCA proposed in this article is simulated based on the NS2.35 network simulator. An iridium-like constellation is adopted, and the virtual node detection time is set to 10 s.
Algorithm 1 KFWCA Optimal Path Generation Process

**Pre-set parameters:** $T_{\text{Delay}}$, Kalman measurement noise, $q_{\text{min}}/q_{\text{max}}$ in the QoS model, conflicting subobjective rules $w_i$, wolf update time $\Delta t$, search step $\lambda$, maximum number of iterations $\chi$, and accuracy $\kappa$

**Target:** Find the path $\text{Best} = ISL\text{benefit}_{x_1,x_d}(t)$ with the highest odor

**Routing process:**
1. Set $G(V,E,O)$, where $O$ is the odor concentration at each node
2. Calculate the coefficients $m$, $n$, and $n'$ in the QoS model
3. Use $q_{\text{res}}$ to update $u_{\text{SID}A}, u_{\text{SID}B}, u_{\text{SID}C}$, and obtain the QABC $U_{\text{SID}}(t)$
4. Initialize $\text{path}_{s \rightarrow d}^i = \{x_{i1}, x_{i2}, \cdots, x_{ip}, x_{id}\}$ and $\text{ISLbenefit}_{x_1,x_d}(t) = 0$
5. If $\chi_0 \geq \chi$
6. while $\chi_0 \geq \chi$, do:
7. for each node $i$, do:
8. $\hat{q}_k = \hat{q}_k - 1$
9. $+\sqrt{\sum_{j=1}^{P_x} \hat{q}_k - 1 + Q^2}$
10. Calculate the reserved capacity $I_{\text{BorC}}(t)$ through (20)
11. Record relay nodes and loop avoidance information
12. Determine whether the detour satisfies the constraint in equation (10)
13. end for
14. Calculate the accumulated odor concentration $\sum_{j \in P_x} ISL\text{benefit}_{i,j}(t)$
15. Calculate the path selection probability $P_{(s \rightarrow d)x_i}$ according to formula (19)
16. end while $\chi$ meets specific requirements
17. Output the optimal path $\text{Best} = ISL\text{benefit}_{x_1,x_d}(t)$
18. Output the optimal path $\text{Best} = ISL\text{benefit}_{x_1,x_d}(t)$

The capacity of all links is 20 Mbps, and the buffer of each node is 80 packets of 1 kb. The three traffic distributions are 5%, 25%, and 70%, and the routing table is updated in a 40 s period. The monitoring interval of the queue is set to 10 ms, the Kalman measurement noise $Q = 2$ packets, $q_{\text{min}} = 20%$, $q_{\text{max}} = 80%$ in the QoS mathematical model, $T_{\text{Delay}} = 300$ ms, the update time of the head wolf is 3 s, the search step length of a detected wolf is 2 hops, the maximum number of iterations is 5, the accuracy constraint is set to 0.8 (probability), and $w_1 = 0.4$, $w_2 = 0.5$ and $w_3 = 0.1$ indicate the weights of the time delay and QoS support optimization. Each ground base station adopts Pareto-distributed control on-off flow with a period of 400 ms and a shape parameter of 1.3. According to the data in reference [15], a global traffic flow table is established, and the traffic flow over the course of a day is 1.6 Tb; the average value over 10 repetitions for 24 hours is the experimental result. In this article, the Dijkstra, TLR [26], and LBRA-CP [30] algorithms are compared with the KFWCA. The route update time of each algorithm is 40 s. A route optimization strategy based on ant pheromones was proposed in the literature [30]. The algorithm continuously adjusts the probability of each node through forward and reverse ant movements and finally completes the route table update based on load balancing.

**B. SIMULATION RESULTS AND ANALYSIS**

1) AVERAGE ROUTE DELAY

The delay performance is analyzed by determining the average end-to-end routing delay in the system. The simulation results of the four algorithms are shown in Figure 6. The LBRA-CP algorithm and KFWCA yield higher average delays than the other algorithm in low-traffic states because the new cost routing function designed in this article achieves load balancing and QoS support at the cost of using long-distance satellites, so the efficiency is temporarily poor. However, as the network traffic burden increases, the delay of the Dijkstra algorithm increases significantly along the single considered path. Similarly, the local detour strategy of the TLR algorithm has better initial performance to some extent, but after the load pressure reaches a certain level, the performance will decrease due to the increase in the queuing delay. The average delays of the LBRA-CP algorithm and KFWCA are comparatively small under high-traffic conditions and intersect at 2.5 Tb of the daily traffic load; this result suggests that wolf colony optimization in the KFWCA displays better global search performance than LBRA-CP through the initial bypass mechanism of class B and C services.
2) PACKET LOSS RATE
The simulation of the packet loss rate in one day is shown in Figure 7. Compared with the other three algorithms, the KFWCA obtains the lowest packet loss rate, which verifies the effectiveness of the algorithm. Notably, the benefit function of the algorithm considers the transmission and queuing delays in the global network, thus effectively avoiding the packet loss caused by queue overflow. The Dijkstra algorithm does not consider route detours, so the number of packets lost increases rapidly with increased traffic; additionally, even in the initial state, some packets are directly lost. The TLR strategy avoids the long queue phenomenon by utilizing a unique traffic congestion discrimination mechanism. TLR avoids the loss of data packets through the explicit analysis of the node state during transmission. However, routing only takes place in the local area of a congested node. The local adjustment strategy cannot meet the needs of network transmission when the congestion intensifies to the shortest path, resulting in the loss of data packets. The LBRA-CP algorithm has made a great breakthrough compared with the other two algorithms, and it predicts the node queue length through the EWMA mechanism. Both the theoretical analysis and simulation results indicate that the effect of EWMA prediction is not as good as that for the KFWCA proposed in this article.

3) FLOW DISPERSION INDEX
From the simulation results in Figure 8, we can assess the load balancing ability of the algorithm and use the common distribution coefficient for analysis. The range of this value is [0,1], and the larger the distribution coefficient is, the better the distribution effect. Through the flow distribution index for different traffic conditions throughout a day, we can find that the TLR algorithm, LBRA-CP algorithm and KFWCA perform better than the Dijkstra algorithm because they all apply certain adaptive adjustment strategies. The daily traffic load of 3.4 Tb is the threshold at which the two optimization algorithms notably differ in performance; specifically, our algorithm and the LBRA-CP algorithm achieve good shunting performance under low-traffic conditions, and the search model in this article can disperse the traffic to unused nodes under high-traffic conditions.

4) NETWORK THROUGHPUT
The throughput obtained through the simulations of the four algorithms is shown in Figure 9. Notably, idle satellites cannot be fully utilized in the routing procedure due to the lack of a reasonable adjustment mechanism in the Dijkstra algorithm. Notably, in the case of high traffic, a large amount of data is directly lost due to the increase in the queuing delay, which indirectly leads to the low throughput of the network for a long time. The TLR and LBRA-CP algorithms adopt a detour strategy to improve the throughput to some extent. However, the congestion avoidance mechanisms are only determined by analyzing nearby satellites, so the throughput increase of
the LBRA-CP algorithm is limited. The throughput of the KFWCA is obviously better than that of the other algorithms because the demand is determined through a random search process from a global perspective with a limited step size, and the routing table is adjusted by the algorithm to improve the corresponding global characteristics.

Because the newly designed measurement model may fall to local optima, this article innovatively designs the KFWCA to solve three conflicting subproblems. First, an intrasatellite Kalman filter is used to process instantaneous traffic. Then, the wolf colony algorithm is used to find the path with the maximum odor concentration. Finally, the path with the largest odor concentration is the optimal routing solution. Moreover, the algorithm adopts mechanisms for loop avoidance and bottleneck reservation. Through the NS2.35 simulation tools, the Dijkstra, TLR, LBRA-CP and KFWC algorithms are compared. The simulation results show that the proposed algorithm displays superior performance based on traffic allocation, the routing delay, the packet loss rate and computational resource use, especially in the case of high-traffic conditions.

This article does not consider satellite storage resources, battery resources, or antenna resources, which are also important factors that influence routing. Subsequently, the concept of resource allocation management and deep learning technology can be introduced into routing methods to improve the practical application value of these methods.

VI. CONCLUSION AND FUTURE RESEARCH

To improve the theory and application level of NGEO satellite network load balancing and QoS support, for the first time, this article establishes a mathematical model of different services considering changes in the remaining queue length. We design a weighted routing benefit measurement model based on the class-A QoS criterion, the network delay, the number of path nodes and the load balancing factor.

REFERENCES

[1] Y. Su, Y. Liu, Y. Zhou, J. Yuan, H. Cao, and J. Shi, “Broadband LEO satellite communications: Architectures and key technologies,” IEEE Wireless Commun., vol. 26, no. 2, pp. 55–61, Apr. 2019.
[2] W. Liu, Y. Tao, and L. Liu, “Load-balancing routing algorithm based on segment routing for traffic return in LEO satellite networks,” IEEE Access, vol. 7, pp. 112044–112053, 2019.
[3] F. Babich, M. Comiso, A. Cuttin, M. Marchese, and F. Patrone, “Nanosatellite-5G integration in the millimeter wave domain: A full top-down approach,” IEEE Trans. Mobile Comput., vol. 19, no. 2, pp. 390–404, Feb. 2020.
[4] W. Zhou, Y. Zhu, Y. Li, Q. Li, and Q. Yu, “Research on hierarchical architecture and routing of satellite constellation with IGSO-GEO-MEO network,” Int. J. Satell. Commun. Netw., vol. 38, no. 2, pp. 162–176, Mar. 2020.
[5] B. Feng, G. Li, G. Li, Y. Zhang, H. Zhou, and S. Yu, “Enabling efficient service function chains at terrestrial-satellite hybrid cloud networks,” IEEE Netw., vol. 33, no. 6, pp. 94–99, Nov. 2019.
[6] Z. Wu, G. Hu, F. Jin, Y. Song, Y. Fu, and G. Ni, “A novel routing design in the IP-based GEO/LEO hybrid satellite networks,” Int. J. Satell. Commun. Netw., vol. 35, no. 3, pp. 179–199, May 2017.
[7] J. Chen, L. Liu, and X. Hu, “Towards a throughput-optimal routing algorithm for data collection on satellite networks,” Int. J. Distrib. Sensor Netw., vol. 12, no. 7, Jul. 2016, Art. no. 15501471665860.
[8] Q. Liu, H.-P. Ren, R.-J. Tang, and J.-L. Yao, “Optimizing co-existing multicast routing trees in IP network via discrete artificial fish school algorithm,” Knowl.-Based Syst., vol. 191, Mar. 2020, Art. no. 105276.
[9] N. Yarr and M. Ceriotti, “Optimization of intersatellite routing for real-time data download,” IEEE Trans. Aerosp. Electron. Syst., vol. 54, no. 5, pp. 2356–2369, Oct. 2018.
[10] D. CQ, S. QY, and L. Guo, “An intelligent computing method for contact plan design in the multi-layer spatial node-based Internet of Things,” Sensors, vol. 18, no. 9, pp. 2852–2870, Sep. 2018.
[11] W. Han, B. Wang, Z. Feng, B. Zhao, W. Yu, and Z. Tang, “Exploring the gateway-based distributed location management schemes in LEO satellite networks,” IEICE Trans. Commun., vol. E101.B, no. 3, pp. 825–834, 2018.
[12] G. Sato, N. Uchida, and Y. Shibata, “Resilient disaster network based on software defined cognitive wireless network technology,” Mobile Inf. Syst., vol. 2015, Mar. 2015, Art. no. 308194.
[13] C. Sun, B. Yin, Z. Dou, D. Jia, and C. Pang, “A routing protocol combining link state and distance vector for GEO-GEO satellite backbone network,” Mobile Netw. Appl., vol. 24, no. 6, pp. 1937–1946, Dec. 2019.
[14] Z. Na, Y. Liu, Y. Cui, and Q. Guo, “Research on aggregation and propagation of self-similar traffic in satellite network,” *Int. J. Hybrid Inf. Technol.*, vol. 8, no. 1, pp. 325–338, Jan. 2015.

[15] H. Nishiya, D. Kudoh, N. Kato, and N. Kadowaki, “Load balancing and QoS provisioning based on congestion prediction for GEO/LEO hybrid satellite networks,” *Proc. IEEE*, vol. 99, no. 11, pp. 1998–2007, Nov. 2011.

[16] H. Nishiya, Y. Tada, N. Kato, N. Yoshimura, M. Toyoshima, and N. Kadowaki, “Toward optimized traffic distribution for efficient network capacity utilization in two-layered satellite networks,” *IEEE Trans. Veh. Technol.*, vol. 62, no. 3, pp. 1303–1313, Mar. 2013.

[17] L. Song, Z. Youjian, S. Fuchun, and L. Hongbo, “Low-load survivable routing algorithm based on SDN,” *Mobile Netw. Appl.*, vol. 23, no. 2, pp. 285–300, Apr. 2018.

[18] P. Liu, H. Chen, S. Wei, L. Li, and Z. Zhu, “Hybrid-traffic-detour based load balancing for onboard routing in LEO satellite networks,” *China Commun.*, vol. 15, no. 6, pp. 28–41, Jun. 2018.

[19] B. T. Thumati and S. Jagannathan, “A model-based fault-detection and prediction scheme for nonlinear multivariable discrete-time systems with asymptotic stability guarantees,” *IEEE Trans. Neural Netw.*, vol. 21, no. 3, pp. 404–423, Mar. 2010.

[20] C. Katris and S. Daskalaki, “Dynamic bandwidth allocation for video traffic using FARIMA-based forecasting models,” *J. Netw. Syst. Manage.*, vol. 27, no. 1, pp. 39–65, Jan. 2019.

[21] Y. Lu, F. Sun, and Z. Zhao, “Virtual topology for LEO satellite networks based on Earth-fixed footprint mode,” *IEEE Commun. Lett.*, vol. 17, no. 2, pp. 357–360, Feb. 2013.

[22] Q. Chen, J. Guo, L. Yang, X. Liu, and X. Chen, “Topology virtualization and dynamics shielding method for LEO satellite networks,” *IEEE Commun. Lett.*, vol. 24, no. 2, pp. 433–437, Feb. 2020.

[23] M. Jia, S. Zhu, L. Wang, Q. Guo, H. Wang, and Z. Liu, “Routing algorithm with virtual topology toward to huge numbers of LEO mobile satellite network based on SDN,” *Mobile Netw. Appl.*, vol. 23, no. 2, pp. 285–300, Apr. 2018.

[24] K. Zhang, Y. Wang, and W. Wang, “A two-hops state-aware routing strategy based on deep reinforcement learning for LEO satellite networks,” *Electronics*, vol. 8, no. 9, p. 920, Aug. 2019.

[25] G. Song, M. Chao, B. Yang, and Y. Zheng, “TLR: A traffic-light-based intelligent routing strategy for NGEO satellite IP networks,” *IEEE Trans. Wireless Commun.*, vol. 13, no. 6, pp. 3393–3399, Jun. 2014.

[26] L. Liu, T. Zhang, and Y. Lu, “A novel adaptive routing algorithm for delay-sensitive service in multihop LEO satellite network,” *KSII Trans. Internet Inf. Syst.*, vol. 10, no. 68, pp. 3551–3567, Aug. 2016.

[27] Z. Shang, C. Liu, S. He, C. Li, and Q. Lu, “A QoS routing strategy using fuzzy logic for NGEO satellite IP networks,” *Wireless Netw.*, vol. 24, no. 1, pp. 295–307, Jan. 2018.

[28] H. Wang, G. Wen, N. Liu, J. Zhang, and Y. Tao, “A load balanced routing algorithm based on congestion prediction for LEO satellite networks,” *Cluster Comput.*, vol. 22, no. S4, pp. 8025–8033, Jul. 2019.

[29] S. Mz, “A first look at cellular machine-to-machine traffic: Large scale measurement and characterization,” *IEEE ACM Trans. Netw.*, vol. 21, no. 6, pp. 1960–1973, Jan. 2013.

[30] Q. Chen, X. Chen, L. Yang, S. Wu, and X. Tao, “A distributed congestion avoidance routing algorithm in mega-constellation network with multi-gateway,” *Acta Astronautica*, vol. 162, pp. 376–387, Sep. 2019.

---

**LANYONG ZHANG** (Member, IEEE) received the Ph.D. degree in control science and engineering from Harbin Engineering University, in 2011. He is currently a master’s Supervisor with Harbin Engineering University. His main research interests include electromagnetic compatibility prediction and measurement, satellite communication, stochastic signal processing, fast transient analysis and modeling of field-excited multiconductor networks, communication network EMC, electromagnetic interaction with advanced composite materials, and satellite routing strategy. In 2009, he received the Excellent Gradate Award from Harbin Engineering University.

---

**DI WU** was born in Harbin, China, in 1994. He received the master’s degree in electrical engineering from Harbin Engineering University, in 2019, where he is currently pursuing the Ph.D. degree in control science and engineering. His main research interests include networked control, multilayer satellite network architecture, satellite routing strategy, and satellite-air-terminal integrated networks.

---

**SHENG LIU** was born in Bacheng, China, in 1957. He received the B.Eng. degree in industrial automation from the Harbin University of Civil Engineering and Architecture, Harbin, China, in 1982, and the master’s and Ph.D. degrees in theory and control engineering from Harbin Engineering University, Harbin, in 1982 and 2000, respectively. He is currently a Professor with Harbin Engineering University. His current research interests include remote network control, satellite-ground integrated networks, TCP network control, multimedia streaming technology, network security encryption, underwater sonar communication, and ship control systems.