Multi-Constraint Routing Algorithm for Satellite Networks Based on SDN and NDN (Postprint)

Authors: Liu Zhiguo, Yao Qiaoyu, Pan Chengsheng

Date: 2022-04-07T00:00:00+00:00

Abstract

To address the problems of high content transmission delay, high packet loss rate, and low request hit rate in NDN satellite networks, we propose a multi-constraint routing algorithm for satellite networks based on SDN and NDN, named SNMcRA (Satellite Networks Multi-constraint Routing Algorithm). By leveraging SDN's centralized control and global view, a multi-constraint routing model is established that integrates link multi-constraint information with an ant colony algorithm to solve for the minimum-cost path satisfying constraints on delay, bandwidth, and packet loss rate. Nodes dynamically construct the Forwarding Information Base (FIB) and Pending Interest Table (PIT) during packet forwarding. Experimental results demonstrate that, compared with the DSP algorithm, the proposed algorithm reduces delay by 35%, improves bandwidth utilization by 29%, decreases packet loss rate by 17%, and also exhibits significant advantages in request hit rate.

Full Text

Preamble

Satellite Network Multi-Constraint Routing Algorithm Based on SDN and NDN

Liu Zhiguo1,2†, Yao Qiaoyu1,2, Pan Chengsheng3

1. School of Information Engineering, Dalian University, Dalian Liaoning 116600, China;
2. Key Laboratory of Communication & Network, Dalian Liaoning 116600, China;
3. School of Electronics & Information Engineering, Nanjing University of Information Science & Technology, Nanjing 211800, China

Abstract: To address the problems of high content transmission delay, high packet loss rate, and low request hit rate in NDN satellite networks, this paper proposes a multi-constraint routing algorithm for satellite networks based on SDN and NDN, named SNMcRA (Satellite Networks Multi-constraint Routing Algorithm). Leveraging SDN's centralized control and global view, the algorithm establishes a multi-constraint routing model that integrates link multi-constraint information with an ant colony algorithm to solve for the least-cost path satisfying constraints on delay, bandwidth, and packet loss rate. Nodes dynamically construct the Forwarding Information Base (FIB) and Pending Interest Table (PIT) during packet forwarding. Experimental results demonstrate that compared with the DSP algorithm, SNMcRA reduces delay by 35%, improves bandwidth utilization by 29%, reduces packet loss rate by 17%, and offers significant advantages in request hit rate.

Key words: SDN; NDN; satellite network; routing algorithm

Received Date: 2021-12-21; Revised Date: 2022-02-17
Funding: National Natural Science Foundation of China (61931004)

Author Biographies:
Liu Zhiguo (1974–), male, professor, master's supervisor, Ph.D., from Shenyang, Liaoning, China. His research interests include network architecture, protocol technology, and network management (liuzhiguo_{dldx}@163.com).
Yao Qiaoyu (1997–), female, master's student, from Xingtai, Hebei, China. Her research focuses on named data networking and software-defined satellite network communications.
Pan Chengsheng (1962–), male, professor, doctoral supervisor, Ph.D., from Yixing, Jiangsu, China. His main research direction is satellite network communications.

---

0 Introduction

With the rapid development of space technology, satellite networks have become a critical component of global communications. As the core of satellite network communication protocols, routing protocols play a significant role in improving data transmission efficiency and reliability. Meanwhile, user demand for multimedia content such as video and voice is growing exponentially. However, IP-based satellite networks suffer from inherent limitations in content transmission: the binding of identity and location causes repeated transmission of identical content within the network, wasting substantial on-board bandwidth resources. Therefore, a new network architecture is needed to overcome these challenges in satellite network environments.

Named Data Networking (NDN), a data-centric future network architecture, decouples content from location and supports in-network caching, alleviating the low transmission efficiency issues of traditional TCP/IP networks. NDN employs a receiver-driven communication model with two packet types: Interest packets and Data packets. Packet forwarding is managed through three data structures: the Content Store (CS), Pending Interest Table (PIT), and Forwarding Information Base (FIB).

Existing NDN forwarding methods offer certain advantages in stable topologies. For instance, Wang L et al. proposed an OSPF-based NDN routing protocol that distributes routing messages through flooding to achieve content delivery. Zhang L created network topology through periodic broadcast of link-state information and used hop-by-hop forwarding instead of OSPF-based periodic prefix announcement flooding. However, in satellite networks with highly dynamic topology, existing static mapping and flooding routing or link-state broadcast-based routing mechanisms are unsuitable due to two key problems: (1) frequent broadcasting or multicasting of Interest packets causes multiple data sources to respond to requests simultaneously, resulting in redundant data transmission; and (2) frequent satellite link disruptions may prevent Data packets from returning via the reverse path of Interest packet transmission.

To address these issues, Hasan M A Islam et al. combined NDN with DTN to solve the problem of Data packets failing to receive timely replies due to frequent disruptions. Liu Di et al. modeled the predictable satellite link switching status using time-varying graphs to dynamically calculate time-dependent fastest paths for packet transmission, but this approach only considered single-objective optimization and suffered from low routing efficiency due to high topology variability. Zhou Y et al. employed a connection graph routing method based on satellite connection plans to determine packet forwarding paths, using optimal and suboptimal paths simultaneously, but this method lacked dynamic adaptability and imposed high requirements on satellite nodes.

Given these limitations, this paper proposes SNMcRA (Satellite Networks Multi-constraint Routing Algorithm), a multi-constraint routing algorithm based on SDN and NDN for satellite networks. Compared with traditional flooding routing and single-objective optimization methods based on connection graphs, SNMcRA leverages SDN's centralized control and uses an improved ant colony algorithm to obtain the least-cost path satisfying multiple constraints on delay, bandwidth, and packet loss rate, enabling efficient packet transmission.

---

1 SDN-Based Satellite Network System Architecture

The SDN (Software Defined Network) architecture separates the network control plane from the forwarding plane, effectively improving network management, control, and data processing. This paper designs a multi-layer satellite network architecture based on SDN called MsnSDN (multi-layer satellite network based on SDN), as shown in [Figure 1: see original paper]. Three GEO satellites serve as local controllers, responsible for obtaining real-time status information of LEO satellites and performing routing calculations and network management.

The forwarding layer consists of LEO satellite nodes that forward data according to flow tables issued by GEO satellites. The ground control center acts as a global controller for centralized control and management of the entire satellite network.

Based on the periodicity and predictability of satellite network topology, this paper adopts a virtual node strategy to "shield" the high dynamics of satellites. According to the regional positions divided by ground stations on the Earth's surface, the virtual satellite node above a region remains unchanged, while actual satellites continuously move and change their logical addresses based on regional positions, thereby forming a globally covered virtual fixed topology.

3 SNMcRA-Based Routing Calculation

In NDN, Interest packets and Data packets are forwarded through FIB and PIT tables. Therefore, to achieve efficient routing in NDN-based satellite networks, FIB and PIT tables must first be constructed. This paper establishes a multi-constraint routing model where GEO satellite controllers obtain global network status information and execute the SNMcRA algorithm to find the least-cost path satisfying constraints on delay, bandwidth, and packet loss rate, thereby enabling dynamic construction of FIB and PIT during packet forwarding.

2.1 SNMcRA-Based FIB Construction Process

In satellite network scenarios, broadcast-based forwarding of Interest packets causes multiple data sources to respond to requests simultaneously, resulting in redundant data transmission and wasting on-board bandwidth resources. Therefore, this paper forwards Interest packets to the GEO satellite controller to obtain the content source satellite node and executes multi-constraint routing calculations to determine the optimal forwarding path, thereby updating/modifying the FIB table, as shown in [Figure 2: see original paper].

When a satellite node receives an Interest packet, it first checks the CS. If the content is found in CS, the Data packet containing the content returns along the original path, satisfying the user request. Otherwise, the node checks the PIT. If a PIT entry for this content exists, the incoming interface information is added to the corresponding entry. If no PIT entry exists, the node continues to check the FIB. If forwarding interface information for this content is found in FIB, the packet is forwarded according to the interface information. If no FIB entry exists, the Interest packet is forwarded to the GEO satellite controller. The controller obtains the content source satellite node based on the parsed content name and executes multi-constraint routing calculations to determine the optimal forwarding path for the Interest packet. It then issues flow tables to the corresponding LEO satellites to complete forwarding. If calculation fails, the Interest packet is traced back or discarded.

2.2 SNMcRA-Based PIT Construction Process

In stable network topologies, Data packets return along the reverse path of Interest packet transmission. However, satellite network topology changes dynamically, and the reverse path of the Interest packet may no longer exist by the time the Data packet returns. Therefore, PIT tables must be constructed dynamically. Since NDN natively supports content-level rerouting, when PIT tables need updating, the GEO satellite controller simply executes SNMcRA routing calculations, as shown in [Figure 3: see original paper].

When a satellite node receives a Data packet, it first checks whether the packet exists in CS. If present, the Data packet is discarded. Otherwise, the node checks the PIT. If the recorded Interest packet transmission link in the PIT still exists, the Data packet is forwarded and cached according to the PIT. If the link no longer exists, the node requests the GEO satellite controller to execute multi-constraint routing calculations for the optimal Data packet forwarding path. If the calculation succeeds, the LEO satellite node forwards the Data packet according to the flow table and caches it following the appropriate caching strategy. If calculation fails, the node returns a NACK.

3.1 Multi-Constraint Model

The LEO satellite system is modeled as a graph $G(V, E)$, where V is the set of satellite nodes and E is the set of inter-satellite links. Let $P(s, d)$ denote a path from source node s to destination node d, and $l_{k,l}$ denote the link between node k and node l. This paper selects the most important path constraints as follows:

1) **Communication Delay**: The sum of transmission delay and node queuing delay along the path, calculated as:

$$delay(P(s,d)) = delay_{tr}(P(s,d)) + delay_{qu}(P(s,d)) = \sum_{l_{k,l} \in P(s,d)} delay_{tr}(l_{k,l}) + \sum_{v \in P(s,d)} delay_{qu}(v)$$

where $delay_{tr}(l_{k,l})$ is the transmission delay of link $l_{k,l}$ and $delay_{qu}(v)$ is the queuing delay of node v.

2) **Remaining Available Bandwidth**: The difference between total link bandwidth and used bandwidth, belonging to concave parameters:

$$band(l_{k,l}) = B(l_{k,l}) - B_{used}(l_{k,l})$$

where $B(l_{k,l})$ is the total bandwidth of link $l_{k,l}$ and $B_{used}(l_{k,l})$ is the used bandwidth.

3) **Packet Loss Rate**: The ratio of lost packets to total transmitted packets, belonging to multiplicative parameters:

$$loss(P(s,d)) = 1 - \prod_{l_{k,l} \in P(s,d)} (1 - loss(l_{k,l}))$$

where $loss(l_{k,l})$ is the packet loss rate of link $l_{k,l}$.

This paper defines the path cost $cost(P(s,d))$ as the weighted sum of communication delay, available bandwidth, and packet loss rate:

$$cost(P(s,d)) = \omega_1 \frac{delay(P(s,d))}{D_{max}} + \omega_2 \frac{B_{min}}{band(P(s,d))} + \omega_3 \frac{loss(P(s,d))}{L_{min}}$$

where D_{max} is the minimum communication delay in the current satellite network, B_{min} is the maximum available bandwidth among current satellite network links, L_{min} is the minimum packet loss rate in the current satellite network, and $\omega_1, \omega_2, \omega_3$ are relative weights for delay, available bandwidth, and packet loss rate, respectively, with $\sum_{i=1}^{3} \omega_i = 1$.

The algorithm's objective is to find a path with minimum cost for each content request while satisfying delay, bandwidth, and packet loss rate constraints. The multi-constraint routing model is established as follows:

$$\begin{aligned} \min \quad & cost(P(s,d)) \\ \text{s.t.} \quad & delay(P(s,d)) \leq D_{max} \\ & band(P(s,d)) \geq B_{min} \\ & loss(P(s,d)) \leq L_{max} \end{aligned}$$

where D_{max}, B_{min}, and L_{max} represent the constraint thresholds for communication delay, available bandwidth, and packet loss rate, respectively.

3.2 Model Solving Based on Improved Ant Colony Algorithm

The ant colony algorithm simulates ants releasing pheromones along their paths during foraging. Shorter paths attract more ants, resulting in higher pheromone concentrations and more ants selecting those paths, ultimately finding the shortest path. However, the basic ant colony algorithm aims only to find the shortest path and easily falls into local optima. This paper combines multi-constraint conditions with the ant colony algorithm, fully considering link multi-constraint information during pathfinding to efficiently solve for optimal paths satisfying delay, bandwidth, and packet loss rate constraints.

3.2.1 State Transition Rule Based on Prior Knowledge and Probability Drive The basic ant state transition formula is:

$$P_q(k,l) = \begin{cases} \frac{[\tau_{k,l}(t)]^{\alpha} \cdot [\eta_{k,l}(t)]^{\beta}}{\sum_{s \in \text{prec}(q)} [\tau_{k,s}(t)]^{\alpha} \cdot [\eta_{k,s}(t)]^{\beta}}, & l \in \text{prec}(q) \\ 0, & l \notin \text{prec}(q) \end{cases}$$

where $P_q(k,l)$ is the probability of ant q transferring from satellite k to satellite l, $\text{prec}(q)$ is the set of nodes waiting to be visited by ant q, $\tau_{k,l}(t)$ is the pheromone concentration on link $l_{k,l}$ at time t, α is the pheromone heuristic factor reflecting the influence of pheromone concentration on the transition rule, $\eta_{k,l}(t)$ is the heuristic information on the link from node k to node l at time t, and β is the heuristic function factor reflecting the influence of heuristic information on the transition rule.

The basic ant state transition rule only selects the next hop based on probability, resulting in high randomness and slow convergence. Therefore, this paper employs prior knowledge selection combined with probability drive to determine the ant's next movement direction, better utilizing the positive feedback mechanism of ants than the basic ant colony algorithm. The improved state transition rule is:

$$l = \begin{cases} \arg\max_{l\in \text{prec}(q)}\{[\tau_{k,l}(t)]^{\alpha}\cdot[\eta_{k,l}(t)]^{\beta}\}, & p \le p_0 \\ \text{random selection based on } P_q(k,l), & p > p_0 \end{cases}$$

where p is a random number uniformly distributed in $[0,1]$, p_0 is the state transition factor calculated as:

$$p_0 = 1 - \frac{N_c}{N_{max}}$$

where N_{max} is the maximum number of iterations and N_c is the current iteration number. When $p \le p_0$, the algorithm uses prior knowledge for non-random search, transferring to the node with the maximum product of pheromone and heuristic function. When $p > p_0$, it calculates random transition probabilities for all nodes satisfying constraints and selects the node with higher probability.

In the early iterations, p_0 is larger, making nodes more likely to select deterministic transfers and accelerating local optimal path search. In later iterations, p_0 becomes smaller to increase random transition probability and prevent falling into local optima. Thus, the improved state transition rule dynamically adjusts the state transition factor, enriching the selectability of next-hop nodes and preventing the algorithm from getting trapped in local optima.

3.2.2 Pheromone Update Rule Based on Multi-Constraint Link Cost

The basic pheromone update method is:

$$\tau_{k,l}(t+1) = (1-\rho)\cdot\tau_{k,l}(t) + \Delta\tau_{k,l}(t)$$

where ρ is the pheromone volatilization factor ($0 < \rho < 1$), $\tau_{k,l}(t)$ is the pheromone amount on link $l_{k,l}$ at time t, and $\Delta\tau_{k,l}(t)$ is the pheromone increment left by ants on link $l_{k,l}$:

$$\Delta\tau_{k,l}(t) = \sum_{q=1}^{m}\Delta\tau_{k,l}^{q}(t)$$

where Q is a constant representing pheromone intensity and L_q is the path length traveled by ant q in the current cycle.

The basic pheromone update method only considers single path length factor and is unsuitable for solving multi-constraint routing paths. Therefore, this paper combines multi-constraint conditions with pheromone update, enabling the ant colony to perceive path parameters such as delay, bandwidth, and packet loss rate in real-time and adjust path search strategies promptly. The improved pheromone update rule is:

$$\tau_{k,l}(t+1) = (1-\rho) \cdot \tau_{k,l}(t) + \Delta\tau_{k,l}(t)$$

where $\Delta\tau_{k,l}(t)$ is calculated as:

$$\Delta\tau_{k,l}(t) = \begin{cases} \frac{1}{cost(P(s,d))}, & l_{k,l} \in P(s,d) \\ 0, & l_{k,l} \notin P(s,d) \end{cases}$$

and $cost(P(s,d))$ is the path cost value of the current optimal solution ant, as defined in Equation (4). The smaller the cost of the selected path, the more pheromone is added to that path, guiding more ants to choose it.

To avoid excessively high or low pheromone concentrations causing premature local optima or search stagnation, this paper restricts pheromone amounts on all optimized paths to the range $[\tau_{min}, \tau_{max}]$. When exceeding this range, the pheromone amount is forcibly limited to τ_{min} or τ_{max}:

$$\tau_{k,l}(t+1) = \begin{cases} \tau_{min}, & \tau_{k,l}(t+1) < \tau_{min} \\ (1-\rho) \cdot \tau_{k,l}(t) + \Delta\tau_{k,l}(t), & \tau_{min} \le \tau_{k,l}(t+1) \le \tau_{max} \\ \tau_{max}, & \tau_{k,l}(t+1) > \tau_{max} \end{cases}$$

The steps for solving the optimal multi-constraint path in satellite networks using the improved ant colony algorithm are described as follows:

Step 1: Initialize multi-constraint condition parameters for network nodes and links, as well as basic parameters in the ant colony algorithm.

Step 2: Remove links that do not satisfy constraints from the network based on multi-constraint conditions to obtain a new network topology. Then, based on the new network topology, set the iteration count $N_c = 1$.

Step 3: Set the source node s as the ant's current node and add it to the tabu table.

Step 4: The ant selects the next-hop node based on the state transition rule (Equation (8)) and multi-constraint conditions, adding the selected node to the tabu table.

Step 5: The ant checks whether the current node is the destination node. If yes, the pathfinding is declared successful and the algorithm jumps to Step 7; otherwise, it proceeds to Step 6.

Step 6: The ant checks whether the set of unvisited nodes for the current node is empty. If empty, pathfinding is declared failed; otherwise, the algorithm returns to Step 4.

Step 7: The destination node d selects an optimal path based on path cost $cost(P(s,d))$. The ant returns along the original path and updates pheromones according to Equation (13).

Step 8: If $N_c < N_{max}$, set $N_c = N_c + 1$ and return to Step 3; otherwise, the algorithm loop ends and outputs the optimal solution.

4 Performance Evaluation

4.1 Simulation Scenario and Parameter Settings

In the simulation, three GEO satellites serve as controllers to manage the entire network in real-time, while LEO satellites simulate the forwarding layer using the Iridium constellation. The GEO satellite controllers collect node and link information from the LEO network, with specific parameters shown in .

Table 1 Satellite Network Orbit Parameters

Parameter	GEO Layer	LEO Layer
Satellites per orbit	3	66
Orbit altitude (km)	35786	780
Orbit inclination (°)	0	86.4
Minimum elevation (°)	-	8.2

In the simulation, considering constraints on communication delay, remaining bandwidth, and packet loss rate for transmission services, this paper sets the thresholds for delay, bandwidth, and packet loss rate as $D_{max} = 100$ ms, $B_{min} = 10 \sim 30$ Mbps, and $L_{max} = 3\%$, respectively. Using the eigenvector method, the weights for delay, available bandwidth, and packet loss rate in the cost function are calculated as $\omega = (0.55, 0.25, 0.2)$, with a consistency ratio $CR = 0.04 < 0.1$, indicating the weight vector is acceptable.

The parameter values in the ant colony algorithm significantly affect performance. Regarding the ant quantity parameter, while a large number of ants (approaching problem scale) improves search stability and globality, it slows convergence. Generally, the optimal number of ants is chosen as two-thirds of the number of nodes, so this paper selects $m = 45$. Referencing literature [22], considering that excessively low initial pheromone concentration increases algorithm loop count, this paper sets $\tau_0 = 20$. To prevent premature search termination due to excessively low pheromone concentration, $\tau_{min} = 10$ is set. When pheromone values are within 100, problem scale has minimal impact on pheromone value selection, so $\tau_{max} = 100$ is chosen. To ensure optimal algorithm performance, the maximum iteration count is set to $N_{max} = 50$.

The values of α, β, and ρ are determined experimentally by measuring their impact on the number of iterations required to converge to the optimal solution while fixing other parameters. The experimental results are shown in [Figure 4: see original paper] and [Figure 5: see original paper].

Figure 4 The influence of α and β on the number of iterations

The experiments show that as α and β values increase, the number of algorithm iterations gradually increases. Therefore, this paper selects $\alpha = 1$ and $\beta = 2$, which yield the fewest iterations to converge to the optimal solution. After determining α and β, the experimental results for ρ are shown in [Figure 5: see original paper].

Figure 5 The influence of parameter ρ on the number of iterations

The results indicate that when $\rho = 0.5$, the algorithm requires the fewest iterations to converge to the optimal solution, so $\rho = 0.5$ is selected.

4.2 Simulation Results Analysis

This section compares the convergence performance of the proposed SNMcRA algorithm with the basic Ant Colony Optimization (ACO) algorithm.

[Figure 6: see original paper] shows the algorithm convergence comparison. As the number of ants increases, the iteration count required for the proposed algorithm to reach the optimal solution is consistently lower than that of the ACO algorithm. When the number of ants is 45, the proposed algorithm converges to the optimal solution in approximately 6 iterations. This is because SNMcRA improves upon the basic ant colony optimization algorithm by selecting next-hop nodes through a combination of prior knowledge and probability selection, accelerating local optimal path search. Additionally, it optimizes pheromone update by incorporating link multi-constraint information and sets upper and lower bounds for pheromone concentration to prevent the algorithm from falling into local optima or stopping search prematurely. Consequently, the algorithm achieves faster convergence.

Figure 6 Algorithm convergence

Simulations were also conducted to compare ACO, DSP (Dijkstra Shortest Path), and SDN-SPA (SDN-based Shortest Path Algorithm) under the same scenario, with detailed analysis and comparison of network transmission delay, link bandwidth utilization, and packet loss rate.

Combining the virtual topology strategy, this paper simulates transmission delays of different routing algorithms across 100 static network topologies, as shown in [Figure 7: see original paper]. The results demonstrate that the proposed algorithm consistently achieves lower transmission delay than the other four algorithms. Specifically, the average transmission delays of ACO, DSP, and SDN-SPA algorithms are 0.10s, 0.12s, and 0.11s, respectively, while the proposed algorithm's average transmission delay is 0.076s, representing a maximum reduction of 35% compared to other algorithms. When network load increases, other algorithms that select paths based solely on distance 容易导致链

路拥塞，因此时延较大。而本文算法将时延作为优化目标，更倾向于选择较低时延的链路，因此路径时延性能较好。

Figure 7 Network transmission delay comparison

[Figure 8: see original paper] shows the bandwidth utilization comparison simulation. When the number of service requests is less than 250, the bandwidth utilization of the four algorithms is similar. When the number of requests exceeds 250, the bandwidth utilization growth trends of SDN-SPA and DSP algorithms slow down because both are based on shortest path algorithms that prioritize routing data flows to a single shortest path. In contrast, ACO and SNMcRA comprehensively consider link bandwidth factors, bypassing congested links during path calculation and utilizing more links for path selection, thus achieving better bandwidth utilization performance. However, the traditional ACO algorithm is prone to local optima, resulting in slightly worse solution quality. The proposed algorithm improves the ant colony algorithm by incorporating link information, enabling faster optimal solution finding. When the number of service requests reaches the maximum, the proposed algorithm's bandwidth utilization improves by up to 29% compared to other algorithms.

Figure 8 Bandwidth utilization comparison

[Figure 9: see original paper] shows the trend of average packet loss rate varying with network load. As network load increases, the packet loss rates of all four routing algorithms gradually increase. Under light network load, the packet loss rates are similar across algorithms. As load increases, the proposed SNMcRA algorithm achieves the lowest packet loss rate, with a maximum reduction of 26% compared to other algorithms. This is because ACO, DSP, and SDN-SPA algorithms do not consider link packet loss rate conditions when searching for shortest paths, selecting only the shortest distance path, which easily leads to network congestion and increased packet loss. The proposed algorithm comprehensively considers link packet loss rate, preferring non-blocking paths with lower packet loss rates during path selection, thus achieving lower packet loss rates.

Figure 9 Comparison of network average packet loss rate

[Figure 10: see original paper] compares request hit rates under stable network conditions across the MsnSDN architecture, SWIMNDN architecture, and SDN satellite network architecture. In the SDN satellite network architecture, forwarding nodes lack caching functionality and content providers are designated by source nodes, resulting in relatively low request hit probability. Under the SWIMNDN architecture, introducing caching functionality at forwarding nodes fully utilizes in-network node caches, effectively improving request hit rates in satellite networks. However, the MsnSDN architecture's global view and centralized control via controllers enable better utilization of content caching for content lookup and retrieval, thus achieving higher request hit rates.

Figure 10 Request hit rate comparison

To analyze the time complexity of the SNMcRA algorithm, this paper breaks down the algorithm's execution steps and analyzes their time complexity progressively, as shown in , where n is the number of LEO satellite nodes and m is the number of ants.

Table 2 SNMcRA Algorithm Time Complexity

Step	Time Complexity
Initialize parameters	$O(n^2)$
Set ant tabu table	$O(m)$
Each ant constructs solution individually	$O(m \cdot n^2)$
All ants complete cycle and calculate path cost	$O(m \cdot n)$
Update link pheromone concentration	$O(n^2)$
Save optimal solution setting	$O(1)$

From the table, when the ant traversal loop count is N_c at algorithm termination, the overall time complexity is $O(c \cdot N_c \cdot n \cdot m)$. According to theoretical and practical requirements, after improving the state transition probability rules and pheromone update method of the traditional ant colony algorithm, the time complexity of SNMcRA is acceptable.

---

5 Conclusion

This paper proposes SNMcRA, a satellite network multi-constraint routing algorithm based on SDN and NDN. By establishing a multi-constraint routing model that integrates multi-constraint information with the ant colony algorithm, the algorithm optimizes the ant state transition rules and pheromone update method in the basic ant colony algorithm, improving convergence speed and optimization capability to efficiently solve for the least-cost path satisfying constraints on delay, bandwidth, and packet loss rate. Simulation experiments demonstrate that the proposed algorithm achieves faster convergence and offers significant advantages over ACO and DSP algorithms in terms of transmission delay, bandwidth utilization, packet loss rate, and request hit rate.

Future work will focus on research into content request aggregation to better leverage the advantages of NDN content distribution.

References

[1] Feng Xuan, Tan Qiang, Liang Zongchuang. The development of satellite mobile communication system [C]// Proc of the 2011 6th International ICST Conference on Communications and Networking in China. Piscataway: IEEE Computer Society, 2011: 1110-1114.

[2] Kodheli O, Andrenacci S, Maturo N, et al. Resource allocation approach for differential doppler reduction in NB-IoT over LEO satellite [C]// 2018 9th Advanced Satellite Multimedia Systems Conference and the 15th Signal Processing for Space Communications Workshop. Piscataway: Institute of Electrical and Electronics Engineers Inc, 2018: 1-6.

[3] Du J, Jiang C, Wang J, et al. Resource allocation in space multi-access systems [J]. IEEE Transactions on Aerospace and Electronic Systems, 2017, 53 (2): 598-618.

[4] Din I U, Hassan S, Khan M K, et al. Caching in Information Centric Networking: strategies, challenges, and future research directions [J]. IEEE Communications Surveys & Tutorials, 2018, 20 (2): 1443-1474.

[5] Liu Di, Huang Chuanhe, Chen Xi, et al. Distributed dynamic routing method of multilayer satellite network based on NDN [J]. Acta Electronic Journal, 2017, 45 (11): 2769-2778.

[6] KIM S Y, LEE J W. Content allocation in Information Centric Network [C]// IEEE International Conference on Information Networking. Korean: IEEE Computer Society, 2016: 135-140.

[7] Hu Xiaoyan, Gong Jian. Inter-domain multi-path routing mechanism of named data network NDN [J]. Journal of Communications, 2015, 36 (10): 211-223.

[8] Zhang L, Afanasyev A, Burke J, et al. Named data networking [J]. Acm Sigcomm Computer Communication Review, 2014, 44 (4): 66-73.

[9] Yi C, Afanasyev A, Wang L, et al. Adaptive forwarding in named data networking [J]. Acm Sigcomm Computer Communication Review, 2012, 42 (3): 62-67.

[10] Yuan H, Crowley P. Scalable pending interest table design: From principles to practice [C]// IEEE Conference on Computer Communications. Piscataway: IEEE, 2014: 2049-2057.

[11] Guo Xian, Yang Shengya, Cao Laicheng, et al. A Novel Routing Based on OLSR for NDN-MANET [C]// Advances in Intelligent Systems and Computing. Berlin: Springer, 2020: 698-714.

[12] Hoque A K M, Amin S O, Alyyan A. NLSR: Named-data link state routing protocol [C]// ACM SIGCOMM Workshop on Information-Centric Networking. Hong Kong: ACM, 2013: 15-20.

[13] Islam H M A, Lukyanenko A, Tarkoma S, et al. Towards disruption tolerant ICN [C]// 2015 IEEE Symposium on Computers and Communication (ISCC). Piscataway: IEEE, 2015: 212-219.

[14] Liu Di, Huang Chuanhe, Chen Xi, et al. Supporting producer mobility via named data networking in space terrestrial integrated networks [J]. Lecture Notes in Computer Science, 2017, 10251: 829-841.

[15] Zhou Yongqi, Li Wenfeng, Zhao Kanglian. Information Centric Networking for Future Deep Space Networks [C]// Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering. Berlin: Springer Verlag, 2019: 437-448.

[16] Xia Wenfeng, Wen Yonggang, Heng F C, et al. A survey on software defined networking [J]. IEEE Communications Surveys and Tutorials, 2015, 17 (1): 27-51.

[17] Bu C, Wang X, Huang M, et al. SDNFV-based Dynamic Network Function Deployment: Model and Mechanism [J]. IEEE Communications Letters, 2018, 22 (1): 93-96.

[18] 齐小刚, 马久龙, 刘立芳. 基于拓扑控制的卫星网络路由优化 [J]. 通信学报, 2018, 39 (02): 11-20. (Qi Xiaogang, Ma Jiulong, Liu Lifang. Routing Optimization of Satellite Network Based on Topology Control [J]. Journal of Communication, 2018, 39 (02): 11-20.)

[19] LIU Z, LI J, WANG Y, et al. A Hybrid Global-local load balancing Routing Scheme for the Internet of Things through Satellite Networks [J]. International Journal of Distributed Sensor Networks, 2017, 13 (3): 69-81.

[20] 魏德宾, 刘健, 潘成胜, 等. 卫星网络中基于多 QoS 约束的蚁群优化路由算法 [J]. 计算机工程, 2019, 45 (07): 114-120. (Wei Debin, Liu Jian, Pan Chengsheng, et al. Ant Colony Optimization Routing Algorithm Based on Multiple QoS Constraints in Satellite Networks [J]. Computer Engineering, 2019, 45 (07): 114-120.)

[21] Li Sicong, Cai Saihua, Li Li, et al. CAAS: a novel collective action-based ant system algorithm for solving TSP problem [J]. SOFT COMPUTING, 2020, 24 (12): 9257-9278.

[22] 陈颖杰, 高茂庭. 基于信息素初始分配和动态更新的蚁群算法 [J]. 计算机工程与应用, 2022, 58 (02): 95-101. (Chen Yingjie, Gao Maoting. Ant Colony Algorithm Based on Pheromone Initial Distribution and Dynamic Update [J]. Computer Engineering and Applications, 2022, 58 (02): 95-101.)

[23] GUO A, ZHAO C, XU F, et al. LEO satellite routing algorithm in software defined space terrestrial integrated network [C]// 2017 17th International Symposium on Communications and Information Technologies (ISCIT). Institute of Electrical and Electronics Engineers Inc, 2017: 1-6.

Note: Figure translations are in progress. See original paper for figures.