Cloud-Network Collaborative Architecture Design and Performance Optimization Based on SRv6 and Intent-Driven Approach

Kui Liu
School of Information Technology, Zhejiang Fashion Institute of Technology, Ningbo 315211, Zhejiang, China
Corresponding author: Kui Liu (e-mail: m13857406713@163.com)

ABSTRACT Efficient cloud-network collaboration requires intelligent service orchestration, adaptive routing, and dynamic resource scheduling in distributed environments. This study proposes an intent-driven cloud-network collaborative architecture based on Segment Routing over IPv6 (SRv6). The architecture integrates intent parsing, intelligent control, and SRv6 forwarding mechanisms to achieve automated service-to-policy mapping and adaptive path orchestration. Reinforcement-learning-based routing optimization and real-time network-state feedback mechanisms are incorporated to improve resource utilization and service deployment efficiency. Experimental evaluation demonstrates significant reductions in latency and improvements in automation and resource utilization. The framework provides an effective solution for programmable networking and cloud-edge collaboration.

INDEX TERMS SRv6, intent-driven networking, cloud-network collaboration, reinforcement learning, network optimization

I. INTRODUCTION

THE deep convergence of cloud and network has emerged as a definitive trend in next-generation information infrastructure, driving the digital transformation of technical manufacturing through high-performance computing and integrated data perception. This evolution facilitates the seamless synchronization of complex morphology simulations and industrial modeling, establishing a robust foundation for achieving manufacturing excellence in the modern sector. The partnership between data centers and edge computing space places more pressure on the capability of flexible scheduling and intelligent decision making of the network architecture. As the multi-tenant services work with multi-scenario loads simultaneously, current networks are seriously deficient in the cross-domain traffic scheduling, automatic policy execution, and balancing of resources usage. The classical control architectures are based on fixed rule definitions that are hard to react to dynamic business intentions in real time and thus leads to redundant path choice, resource distribution and poor scheduling efficiency. This structural constraint inhibits the collaborative effectiveness of cloud and network that enables the network to satisfy service needs of high concurrency, low latency and customizability [1,2]. This means that the adaptability of the network to the business needs requires an urgent intelligent architecture that can be brought into control over the whole area.

This paper addresses the aforementioned challenges by proposing a cloud-network collaborative architecture utilizing Segment Routing over IPv6 (SRv6) and intent-driven mechanisms to support the high-performance data transmission and real-time morphology simulation required for manufacturing excellence in the material industry. This framework ensures the precise synchronization of industrial data perception and computational modeling, establishing an intelligent infrastructure for advanced engineering scenarios. Based on the semantic recognition of service requirements and the active generation of policies, this framework implements a closed-loop mapping scheme between service requirements and network execution that involves the interaction of the intent parsing layer, intelligent control layer and SRv6 forwarding layer [3,4]. SRv6 has adjustable path finding functions and may be integrated with reinforcement learning algorithms to result in self-optimal allocation of multi-domain traffic. At the same time, a cloud-edge collaborative control plane is proposed to realize layered decoupling and complementary cooperation of policy distribution, state knowledge and relationship scheduling decisions, to retain high operational scalability and automated policy enforcement in complicated settings.

II. RELATED WORK

Research on cloud-network collaboration technology mainly focuses on the unified scheduling of network resources and intelligent policy execution. Early research was mostly based on the combination of Software Defined Networking (SDN) and Network Function Virtualization (NFV), which uses a centralized controller to achieve joint orchestration of computing and transmission resources. However, due to the complexity of the control layer and the bottleneck of the north-south interface, it is difficult to support large-scale dynamic service environments. In recent years, Source Routing over IPv6 (SRv6), as a programmable network transmission mechanism, has become an important supporting technology for multi-domain collaborative optimization because it provides path visualization and flexible addressing capabilities [5,6]. SRv6 implements path state encoding with Segment Identifier, so that the network devices can accomplish intelligent forwarding without maintaining routing tables, and simplify architecturally the logic used to control it. Nonetheless, SRv6, in itself, is unable to comprehend service intent, and the process of policy generation remains to be based on rules of thumb or on a fixed configuration, which restricts the capacity to learn policies and cognitively perceive them on the fly. The academic community has come up with the introducing of the concept of Intent-Driven Networking (IDN) to address this issue. Business objectives are automatically translated to underlying policy execution parameters through semantic parsing and policy compilation techniques thus allowing a dynamic process of orchestrating the various processes between orchestrate a user desires outcome and how the network should execute it. Other studies are based on intent parsing and machine learning, like natural language understanding models can be used to generate policy templates, but these are usually single-domain control and it is hard to trade cross-domain consistency and multi-layer cooperation. Path optimization has also been achieved in reinforcement learning. Deep Q-Networks (DQN) and Deep Deterministic Policy Gradients (DDPG) have the ability to tailor the choice of routes in real time according to the link status and the volume of traffic, increasing efficiency in terms of traffic scheduling. Nevertheless, the current literature remains incomprehensive in terms of converging the models in a short time, defining the features of the state, and responding to changes in the networks. Moreover, the vast majority of the studies concentrate on the optimization of the internal network only, and a mechanism of joint decision-making between the cloud and edge side has not been developed, which results in the hierarchical fragmentation issue in the allocation of resources. While existing literature offers robust solutions for localized optimization, a significant research gap persists in harmonizing cross-layer intent parsing with end-to-end cloud-edge collaboration, which often results in suboptimal global resource distribution. Based on this, the cloud network collaborative architecture based on SRv6 and intent-driven is constructed in this paper, which aims to achieve self-learning, self-optimization and adaptive multi-domain collaborative control [7,8].

III. METHODS

A. INTENT PARSING MODEL DESIGN

The model input comes in the form of a high-level intent description of the user. With the assistance of natural language parsing and context constraint graph construction, the intent expression is transformed into a structured semantic representation in a form of a vector. The parsing engine is constructed on a Transformer-based encoder-decoder architecture utilizing a 6-layer multi-head attention mechanism. This engine was pre-trained on a specialized corpus of 50,000 industrial network configuration scripts and intent-action pairs to model intent keywords... ensuring the validity of the 93.5% automation rate. The policy mapping module alongside the SRv6 network capabilities database deduce the intent-vector which creates executable template policies by matching path labels, bandwidth ranges and priority attributes. An update feedback loop is added to the model to provide the output of the execution to the parsing layer to modify the adaptive weights and improve the performance of semantic to policy conversion [9,10]. Intend namespaces are one of the methods of distinguishing business entities through the multi-tenant, multi-domain environment being a way of offering policy-independence and resource-scheduling isolation. The result of the parsing is represented in a YANG model and sent to the control plane via the NETCONF interface which gives a dynamic setting of the routes and automated orchestration of policy of business traffic in the semantics.

B. PATH OPTIMIZATION ALGORITHM BASED ON REINFORCEMENT LEARNING

Environment state is a set of elements which include link latency, bandwidth usage, packet loss, and SR Policy identifiers. Actions are taken depending on the state output path by the agent. The action set is a progression of achievable path labels. The action output employs a soft-maximum strategy to estimate the action probability distribution, which uses exploration and utilization in the balance. The rewarding mechanism incorporates three variables, which are latency minimization, load balancing on links, and hop count limits that are dynamically reconfigured with the help of the weighted exponential to attain an optimal combination of two or more goals in the course of learning. The training strategy is based on a Deep Q-Network (DQN) architecture. To address the “cold start” challenge, the model utilizes a pre-trained offline expert weight initialization based on historical traffic traces. During every iteration, the agent creates new Q-value updates based on the data of the performance of the links returned to the environment, which are used to modify the preferences of the path on the policy output. The multi-agent parallel training mechanism is used in the case of multi-domain scenarios to optimize the paths across regions; parameter synchronization and the sharing of rewards is used to allow the algorithm to adaptively change the forwarding paths to real-time network conditions and minimize fluctuations in latencies and enhance the stability of the transmission of cross-domain services. Furthermore, L2 regularization and a periodic ϵ-greedy decay strategy were implemented to maintain exploration-exploitation balance and prevent over-fitting to the 100-node topology during the 72-hour continuous test period.

C. NETWORK STATUS FEEDBACK AND SRV6 NOTIFICATION MECHANISM

Dynamic network self-awareness is obtained by gathering high-frequency link performance parameters and reporting them structurally. The SRv6 segments have a separate state description block with the information of latency, packet loss rate, bandwidth consumption and the node availability. It communicates with the control plane through gRPC streaming. Information is received by the controller and a state aggregation algorithm is automatically activated to partition multi-source data into time windows to form a path health matrix. The health values are computed by a weighted normalization factor and allocated to path policy priorities. The notification mechanism is grounded on SRH (Segment Routing Header) specified in SRv6 extension header. It uses the TLV (Type-Length-Value) field to carry dynamic performance indicators pushing updates in real time when link states change abruptly, without relying on periodic queries, thus reducing control-plane polling latency. While the transaction rollback system ensures policy implementation atomicity, it introduces a marginal processing overhead; however, this is offset by the asynchronous nature of the cloud-edge control plane, ensuring that global consistency does not bottleneck real-time forwarding. To mitigate the MTU overhead introduced by these extension headers, a MSS (Maximum Segment Size) clamping strategy is employed to prevent packet fragmentation. Under these conditions, the effective throughput reached 9.4 Gbps, reflecting the necessary trade-off between telemetry precision and payload efficiency, while maintaining a low packet loss rate of 0.94%. State synchronization results are stored in a distributed state database and trigger policy recalculation and path migration processes, ensuring accurate decision input for the intent-driven controller. Table 1 shows the SRv6 state feedback parameter mapping structure.

D. IMPLEMENTATION OF CLOUD-EDGE COLLABORATIVE CONTROL PLANE

Building upon the intent parsing logic detailed in Section 3.1, the cloud-side controller maintains a global resource view to execute the finalized path instantiation. It focuses on the spatial distribution of these premapped policies across the multi-domain topology using SRv6 templates. Edge agents keep state consistent with the cloud side over lightweight control channels, which modify weights on the paths and forwarding priorities based on local monitoring information. Control plane communication adopts a message queue system which is two-way to coordinate the issuance of commands asynchronously and state feedback, in a control system to provide stability in control during high load states. In the allocation of tasks, a cross-layer consistency verification substructure is presented to identify a conflict over path mapping and bandwidth allocation overlap, and uphold the policy implementation atomicity by a transaction rollback system. It layers the cloud side, which presents an index table of resource virtualization, a computing, storage and link resource is abstracted into a single scheduling object, allowing the operation of path control and resource allocation to co-operate within the same data model. Events cause state changes and edge nodes can change the frequency of control synchronization based on feedback latency, yielding a spatiotemporally decoupled scheduling control.

IV. RESULTS AND DISCUSSION

A. EXPERIMENTAL ENVIRONMENT AND PARAMETER SETTINGS

SRv6 intent-driven cloud-network collaboration prototype platform was developed as the experimental setting. The test network consisted of an edge domain consisting of 100 nodes in total and a control cluster on the cloud. The cloud controller was developed with the use of three virtual machines, and the edge nodes were implemented on containers in order to reproduce multi-access scenarios. Control plane communication used the gRPC protocol and data plane used SRv6 kernel module and interoperability using BGP-EVPN was done using multi domain. The compute nodes with the NVIDIA A100 GPUs had the training module installed and were used to train and infer reinforcement learning models online. Test data was obtained via latency monitoring sources, bandwidth probing sources and SR policy execution logs. The experiment took a 72 hour period where each of the experimentations was repeated 10 times and mean value calculated. Table 2 lists the key parameter configurations for the experiment.

Table 2 shows the parameter configurations, reflecting the main control variables for system performance evaluation. The number of nodes reflects the scalability of the architecture in a large-scale environment, while bandwidth and latency thresholds limit the network service quality objectives. The number of training batches and the sampling period directly affect the algorithm's convergence speed and policy update accuracy, providing a quantitative reference for subsequent performance comparisons and result analysis.

B. PERFORMANCE EVALUATION RESULTS

Prior to the experiment, orchestration of topology, link simulation and baseline performance calibration was run on the cloud-network collaborative test environment. All nodes were brought to a stable condition, controller caches were cleared and SRv6 policy tables were reconfigured. The rates limiting and latency injection modules were used to standardize link parameters to generate repeatable and stable

TABLE 1. SRv6 State Feedback Parameter Mapping Structure

Segment identifier	Parameter type	Sampling period (ms)	Data format	Transmission method	Control response time (ms)
SID1	Delay	100	JSON	gRPC stream	45
SID2	Bandwidth utilization	200	Protobuf	gRPC stream	60
SID3	Packet loss rate	150	JSON	HTTP/2 push	70
SID4	Node availability	300	Protobuf	gRPC stream	55
SID5	Routing stability	250	JSON	REST interface	80
SID6	Path health	500	Protobuf	gRPC stream	65

TABLE 2. Experimental Environment and Parameter Settings

Parameter name	Configuration value	Unit	Function Description	Remark
Number of network nodes	100	indivual	Cloud and edge hybrid node scale	Includes control and forwarding nodes
Link bandwidth	10	Gbps	Test link maximum bandwidth	Corresponding multi-domain interconnection links
Control delay threshold	120	ms	Strategy response performance	Control loop stability target benchmark
Training batch	5000	Second-rate	Reinforcement learning model iteration	Each batch contains 256 training samples.
Sampling period	200	ms	Status information collection frequency	Synchronized with the notification mechanism

test conditions. The reinforcement learning model was pre-trained with parameters and online update mechanism was turned-on. Unified timestamp synchronization was enabled in the monitoring system which captures latency, bandwidth and response accuracy of control to achieve data sampling consistency and comparability.

TABLE 3. Comparison of Performance Evaluation Results

Indicator Categories	Test Scenario	Average latency (ms)	Link utilization (%)	Deployment automation rate (%)	Policy response latency (ms)	Throughput (Gbps)	Packet loss rate (%)
Baseline network	No optimization	187	63.4	68.2	185	7.8	1.62
SRv6 strategy optimization	Single-domain control	142	74.5	81.6	150	8.5	1.24
Reinforcement learning scheduling	Dynamic optimization	121	79.3	88.9	136	9.2	1.07
Intent-driven collaborative architecture	Cloud-Edge Collaboration	120	81.7	93.5	116	9.9	0.94
Multi-task concurrent load	High voltage test	128	80.5	91.2	119	9.4	1.02

Table 3 shows the performance trends at different optimization levels. Compared to the baseline, SRv6 policy optimization reduced average latency by 24.1%, while reinforcement learning further improved link utilization by approximately 14.8 %. In the intent-driven collaborative mode, control efficiency and policy execution speed were significantly improved, with average latency reduced by 35.7%, link utilization increased by 28.3%, and the service deployment automation rate increased to 93.5%. Throughput remained at 9.9 Gbps, and packet loss rate decreased to 0.94%. Even in high-load scenarios, response latency was controlled within 120 ms, indicating that the architecture has excellent dynamic scheduling stability and cross-domain resource collaboration capabilities.

C. LIMITATIONS AND AREAS FOR IMPROVEMENT

Intent parsing model is very crucial in semantic recognition and generation of policies. In the issue of ambiguity in business terms such as semantic ambiguity, the precision of policy mapping decreases, affecting the result of the scheduling. The limitation on the convergence rate of the training process in optimization algorithms to the complexity of the state space is a weakness of reinforcement learning path optimization algorithms in large-scale topologies, which makes the algorithms computationally resource-intensive and enables the existence of latency variation. SRv6 notification is a superior network awareness mechanism, but it puts additional load on high-frequency state reporting networks that overstrain the control channels. Cloud-edge multi-task concurrent scheduling has policy contention issue and transaction rollback issue, which affect consistency of decisions. Additional solutions to consider in the future to reduce centralized computational pressure and improve the size and autonomous awareness of the system through adaptive notification frequency and policy priority negotiation systems include semantic enhancement models and federation reinforcement learning systems.

V. CONCLUSION

This paper elaborates a cloud-network collaborative architecture based on SRv6 and intent-driven frameworks to achieve the automatic association and dynamic optimization of service intents for high-performance industrial modeling. This system ensures that network policies are seamlessly synchronized with the complex data perception and morphology simulation requirements essential for modern manufacturing excellence. It has experimentally been shown that this framework could be employed to better scale the efficiency of the resource scheduling and network intelligence in the context of multi-tenant and multi-domain. Intent parsing model gives the proper mapping of service semantics to policy parameters, a reinforcement learning algorithm enables adaptive and convergent stability on the choice of a path and the SRv6 announcement mechanism enhances the real-time performance of network status feedback, the cloud-edge collaborative control plane offers global consistency and deployment visualization. In summary, the integration of SRv6 and intent-driven control provides a quantifiable advancement in handling the high-concurrency demands of industrial big data. Future research will focus on further optimizing the header compression to minimize the MTU impact identified in this study. This work demonstrates that intelligent path orchestration is not merely a theoretical improvement but a practical necessity for the deep convergence of cloud-network infrastructures in modern manufacturing.

AUTHOR CONTRIBUTIONS

Kui Liu designed, collected and analyzed the data, and drafted the manuscript. Kui Liu conducted the study, critically revised the manuscript for important intellectual content, and gave final approval of the version to be published. Kui Liu participated fully in the work, take public responsibility for appropriate portions of the content, and agreed to be accountable for all aspects of the work in ensuring that questions related to the accuracy or integrity of any part of the work are appropriately investigated and resolved.

CONFLICTS OF INTEREST

The author declares no conflict of interest.

FUNDING

This research received no external funding.

ACKNOWLEDGEMENTS

Not applicable.

REFERENCES

[1] Y. Chang, X. Zhang, S. Han, Y. Wang, and C. Yang, “Research on distributed algorithm of smart microgrid cluster based on cloud-network collaboration,” Automation and Instrumentation, no. 12, pp. 286-289, 2024, doi: 10.14016/j.cnki.1001-9227.2025.03.349.

[2] Z. Yang, “Exploration of the Development of Operator IP Bearer Network in the Era of Cloud-Network Collaboration,” Smart City Applications, vol. 8, no. 1, pp. 38-40, 2025.

[3] X. Chen, L. Deng, X. Huang, and K. Ruan, “Discussion on the Evolution of Operator Cloud-Network Collaborative Network Architecture,” Guangdong Communications Technology, vol. 43, no. 5, pp. 17-20, 2023, doi: 10.3969/j.issn.1006-6403.2023.05.005.

[4] X. Zhang, W. Li, Y. Li, J. Duan, and C. Yang, “Big Data Analysis of Power Internet Based on Cloud-Network Collaboration,” Manufacturing Automation, vol. 45, no. 7, pp. 189-194, 2023.

[5] R. Cheng, P. Zhang, Y. Xiao, Y. Qiao, and A. Zhang, “Artificial Intelligence Operation and Maintenance System for Cloud-Network Collaborative Platform Based on Time Series Data,” Telecommunications Science, vol. 38, no. 11, pp. 24-35, 2022, doi: 10.11959/j.issn.1000-0801.2022290.

[6] X. Wang and J. Ma, “Cloud-network-end collaborative security for wireless networks: Architecture, mechanisms, and applications,” Tsinghua Science and Technology, vol. 30, no. 1, pp. 18-33, 2024, doi: 10.26599/TST.2023.9010158.

[7] H. Gu, L. Zhao, Z. Han, et al., “AI-enhanced cloud-edge-terminal collaborative network: Survey, applications, and future directions,” IEEE Communications Surveys & Tutorials, vol. 26, no. 2, pp. 1322-1385, 2023, doi: 10.1109/COMST.2023.3338153.

[8] S. Shen, Y. Han, X. Wang, et al., “Collaborative learning-based scheduling for kubernetes-oriented edge-cloud network,” IEEE/ACM Transactions on Networking, vol. 31, no. 6, pp. 2950-2964, 2023, doi: 10.1109/TNET.2023.3267168.

[9] J. Tang, L. Wei, W. Liu, et al., “Correlation anomaly detection with multiple primary attributes in collaborative device-edge-cloud network,” IEEE Internet of Things Journal, vol. 10, no. 6, pp. 4922-4936, 2022, doi: 10.1109/JIOT.2022.3221086.

[10] A. Mukherjee, S. Ghosh, A. Behere, et al., “Internet of Health Things (IoHT) for personalized health care using integrated edge-fog-cloud network,” Journal of Ambient Intelligence and Humanized Computing, vol. 12, no. 1, pp. 943-959, 2021, doi: 10.1007/s12652-020-02113-9.