ANIMA
Internet-Draft
Intended status: Informational
Expires: 18 June 2026

S. Jiang
L. Zhu
BUPT

15 December 2025

The Use Case of Autonomic Traffic Management in the Artificial Intelligence Data Center
draft-jiang-autonomic-traffic-management-in-aidc-00

Abstract

This document describes the use case of autonomic traffic management in the Artificial Intelligence Data Center (AIDC), including the requirements and two management mechanisms, based on the distributed model and the centralized controlling model. It proposed to use the IETF GRASP protocol for the information exchanging, resource negotiation, control signalling, and etc.

Table of Contents

1. Introduction

Against the sweeping backdrop of the artificial intelligence technology entering the "Large Model Era", the AIDC has emerged as the new bedrock of the digital economy. With the continuous enrichment of business scenarios in AIDCs, the loads they bear show significant dynamics and complexity. For example, during large-model training, PB-level training data will be generated for high-frequency interaction between computing nodes, and the proportion of east-west traffic has increased from 30% in traditional data centers to more than 80%. Distributed inference services, on the other hand, need to ensure low-latency responses and have extremely high requirements for

the stability of traffic transmission. Its core mission has fundamentally shifted from traditional general-purpose cloud computing services to hosting the training and inference of models with trillions of parameters. This fundamental transformation in business morphology presents unprecedented and severe challenges to the network architecture of AIDCs. In a massive cluster composed of tens of thousands of high-performance GPU cards, the network is no longer merely a conduit for data transmission; it has become a critical component determining the overall efficiency ("Goodput") of the computing cluster. Unlike the massive volume of small flows characteristic of internet traffic, AI training traffic exhibits distinct features: significant periodic pulses, extremely high bandwidth occupation, and extreme sensitivity to latency and packet loss. In typical scenarios of distributed training, such as All-reduce collective communication, thousands of computing nodes must complete the exchange and update of massive gradients within a millisecond-level synchronization window. This instantaneous burst of traffic often fills switch buffers instantly, creating extremely violent "micro-bursts." For lossless networks employing RDMA (Remote Direct Memory Access) protocols, any minute packet loss or congestion can trigger a PFC (Priority Flow Control) storm or retransmission mechanisms, leading to the stagnation of the entire training task and resulting in the exorbitant waste of computing resources. Consequently, the AIDC imposes nearly harsh requirements on network stability, throughput, and response speed. This critical necessity urgently calls for an automated traffic management mechanism capable of operating without human intervention and possessing millisecond-level response capabilities.

In the construction of current ultra-large-scale AIDCs, the design of networking architecture directly determines the efficiency of computing power scheduling and service bearing capacity. Among various architectures, Spine-Leaf and Fat-Tree topologies have become mainstream choices due to their unique advantages. While both have their merits, they also face distinct bottlenecks in traffic management. The Spine-Leaf architecture is confronted with an inherent load balancing dilemma, where uneven traffic distribution across spine nodes may lead to bottlenecks in data transmission efficiency. The Fat-Tree topology suffers from significant operational complexity, primarily stemming from its multi-layered structure and the intricate configuration requirements for link redundancy and fault tolerance.

In summary, it can be concluded that traditional network management models based on static planning and manual operations are no longer adequate for coping with the dynamic and volatile traffic characteristics of AIDCs. Therefore, building an autonomic and dynamic traffic planning and management mechanism has become the key

to breaking the performance bottleneck of AIDC. Such mechanisms are expected to enable networks to perceive their own operational state in near real time, react promptly to transient traffic dynamics, and autonomously make and adjust control decisions accordingly. Motivated by these requirements, this document examines the use cases of autonomic traffic management in AIDCs and discusses two representative management approaches: a distributed autonomic approach and a centralized controller-led hybrid approach. In both cases, the Autonomic Networking architecture, along with a set of autonomic network tools and protocols defined by the IETF ANIMA Working Group, provides a robust foundation that enables self-perception, self-configuration, self-optimization, and self-healing. In particular, the ANIMA GRASP[RFC8990] can be extended to support information exchange, resource negotiation, and control signaling among autonomic entities.

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. Distributed Autonomic Traffic Management Mechanism in AIDC

Distributed autonomic network technology [RFC8993]provides a refined automated traffic management solution for AIDCs with its characteristics of "decentralized control and autonomous node collaboration". As the core signaling protocol of autonomic networks, GRASP boasts flexible goal-oriented interaction capabilities. Through protocol extensions tailored to AIDC scenarios, such as adding computing power status fields and link quality parameter identifiers, it can span five key phases: topology awareness, multi-path exploration, resource negotiation and reservation, traffic deployment, and resource release. This can form a standardized closed-loop management mechanism, addressing the compatibility and efficiency issues of autonomous inter-node communication.

In the topology awareness phase, each computing node periodically send extended GRASP discovery messages that contains its own identity, computing power status, network interface parameters and other information to surrounding devices. At the same time, the node will receive and parse feedback messages from other computing nodes, switches, and routers, quickly construct and update the local "neighbor node list" in real time, and clarify the available next-hop computing nodes and network devices in the current network, laying

the foundation for subsequent traffic scheduling. This process does not require the intervention of a centralized controller, and the node can update the topology information in milliseconds, adapting to dynamic scenarios such as device plugging and unplugging and temporary failures.

After the topology awareness is completed, the node will start the multi-path exploration process. Based on the locally stored topology information, the node will generate multiple potential transmission paths through intelligent algorithms such as Dijkstra's algorithm or ant colony optimization algorithm, combined with real-time indicators such as link bandwidth, transmission delay, packet loss rate, and jitter. For example, when connecting to a target computing node, it will not only explore the shortest path of "local Leaf node - same-area Spine node - target Leaf node", but also plan the redundant path of "local Leaf node - cross-area Spine node - standby Leaf node", ensuring rapid switching when the main path is congested.

After the path is determined, the node will conduct resource negotiation with all switches and routers on the path by extended GRASP messages. The node will send resource request messages to various network devices, specifying parameters such as bandwidth and queue priority required for traffic transmission. The network devices will respond according to their current resource occupancy (such as remaining bandwidth and number of reserved queues). If resources are sufficient, they will return confirmation information and lock the corresponding resources; if resources are insufficient, they will feed back the available resource quota for the node to adjust the path selection. For example, when the remaining bandwidth of a core switch on a certain path can only meet 60% of the demand, the node will automatically switch to a redundant path for re-negotiation until resource reservation is completed.

After the resource reservation is completed, the system will automatically deploy traffic forwarding rules: on the one hand, configure ACL rules and QoS queues on various network devices to ensure that traffic is transmitted according to the reserved resource priority; on the other hand, configure traffic encapsulation and decapsulation parameters on the source node and target node to ensure the security of data transmission.

After the traffic starts to be transmitted and the migration is completed, the node will send resource release instructions to all network devices on the path, unlock the relevant bandwidth and queues, and reintroduce the resources into the available pool for other traffic scheduling, realizing the circular and efficient utilization of network resources.

4. Traffic Orchestration and Management by Centralized Controllers in AIDC

Furthermore, to solve the limitations of distributed autonomic networks in global resource optimization, AIDCs could introduce centralized controllers and traffic orchestrators to build a hybrid architecture of "distributed execution + centralized decision-making", realizing global collaborative scheduling of computing power and network resources. GRASP could also play as the key communication link connecting distributed nodes and centralized controllers, enabling global collaborative scheduling of computing power and network resources. This model performs particularly prominently in scenarios such as large-model training task iteration and computing node load balancing. The bidirectional interaction capability of the GRASP protocol not only guarantees the flexibility of autonomous inter-node communication but also ensures the centralized controller’s precise grasp of the global state.

When a computing node initiates a traffic migration request due to high load (such as a single-node GPU utilization rate exceeding 90%), hardware maintenance (such as the need to replace a faulty hard disk), or task iteration (such as the need to adjust the scale of the computing cluster when the training task enters the next phase), it will first submit an extended GRASP request message containing the type of demand, data volume, QoS requirements (such as delay upper limit, packet loss rate threshold), and target computing power range to the centralized controller. As the "global command center", the centralized controller will immediately start a multi-dimensional data collection process: collect the operating status of each computing node (including validity, GPU utilization rate, memory occupancy rate, temperature, etc.) through the computing power monitoring module; collect the dynamic occupancy information of the entire link (such as bandwidth utilization rate, delay, fault status of each link) through the network monitoring module; and obtain the priority information of all currently running tasks through the business management module to ensure the comprehensiveness of the decision-making basis.

Based on the collected global data, the centralized controller will make decisions through global optimization algorithms such as integer programming and genetic algorithms: in terms of computing node planning, it will screen out target nodes that meet the requirements based on task requirements and computing node status. For example, it will give priority to allocating nodes with larger GPU memory for high-precision inference tasks, and select target nodes in the same Leaf domain as the source node for low-latency tasks; in terms of path planning, it will comprehensively consider link load balancing and transmission efficiency to avoid concentrating a large amount of

traffic on several core links. For example, when the link utilization rate of a certain Spine node reaches 75%, it will give priority to planning paths passing through other low-load Spine nodes; in terms of resource reservation, it will uniformly calculate the forwarding resources required by each path according to the traffic scale and negotiate and lock them with relevant network devices in advance to avoid resource competition that may occur in distributed negotiation.

After the decision is made, the centralized controller will issue instructions to relevant devices by the extended GRASP: send computing power migration preparation instructions to the source node and target node to complete data backup and receiving preparation; send resource configuration instructions to the switches and routers on the path to complete operations such as bandwidth reservation, queue priority setting, and forwarding rule configuration; send scheduling instructions to the traffic orchestrator to clarify the start time and rate control parameters of traffic migration. During the traffic migration process, the centralized controller will monitor the transmission status in real time. If sudden congestion is found on a certain link, it will immediately trigger a dynamic adjustment mechanism, re-plan the path and update the resource configuration to ensure uninterrupted migration.

After the traffic migration is completed and the target node starts the business normally, the centralized controller will issue resource release instructions by the extended GRASP to unlock the resource reservation on the original path and update the global resource status information. Subsequently, each computing node enters the next round of computing and data transmission process, and the centralized controller continues to monitor the global network status and waits to receive new scheduling requests, forming a complete closed loop of "demand submission - global decision-making - execution scheduling - status update" to ensure the dynamic balance of the overall computing power and network resources of the AIDC.

5. Summary

This document elaborates on the application of autonomic network technology in traffic planning and management of AIDCs. Aiming at the autonomic traffic management demands in AIDCs, this document proposes two solutions: distributed autonomic network and centralized controller-led hybrid architecture. The distributed solution realizes autonomous traffic management through topology awareness, multi-path exploration and other closed-loop processes; the hybrid architecture achieves global resource optimization via centralized decision-making and distributed execution. Both solutions effectively address the challenges of high dynamics and complexity of

traffic in AIDCs, ensuring efficient utilization of computing power and network resources.

6. IANA Considerations

This informational document does not define any IANA items.

7. Security Considerations

This informational document proposes using GRASP for the information exchanging, resource negotiation, etc. These mechanisms shall reuse the security mechanims that has defined by the IETF ANIMA WG. The detailed secure considations shall be given and resoved in the standard documents that define these executable detailed mechanisms and protocol extenstions.

8. References

8.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8990] Bormann, C., Carpenter, B., Ed., and B. Liu, Ed., "GeneRic Autonomic Signaling Protocol (GRASP)", RFC 8990, DOI 10.17487/RFC8990, May 2021, <https://www.rfc-editor.org/info/rfc8990>.

8.2. Informative References

[RFC8993] Behringer, M., Ed., Carpenter, B., Eckert, T., Ciavaglia, L., and J. Nobre, "A Reference Model for Autonomic Networking", RFC 8993, DOI 10.17487/RFC8993, May 2021, <https://www.rfc-editor.org/info/rfc8993>.

Authors' Addresses

Sheng Jiang
Beijing University of Posts and Telecommunications
No. 10 Xitucheng Road
Haidian District, Beijing
China

Email: shengjiang@bupt.edu.cn

Longwei Zhu
Beijing University of Posts and Telecommunications
No. 10 Xitucheng Road
Haidian District, Beijing
China
Email: lwzhu@bupt.edu.cn