Research on Traffic Dispatching Scheme of Multi-service Data Center for 5G Smart Grid

Yanzhi Sun*, Yuming Liu, Feng Tian, Quan Wen and Liqiong Jiang
Yunnan Electric Power Dispatching Control Center, Kunming, Yunnan

*Corresponding author email: 501212807@qq.com

Abstract. Aiming at the flow scheduling problem of 5G power network multi-service data center, this paper designs and proposes a multi-service-oriented multi-factor flow scheduling scheme. Based on the shortest path algorithm, this paper improves and perfects the traffic scheduling strategy, and introduces a set of scoring mechanism. The number of hops, the number of large flows and the remaining bandwidth of each path are used as scoring indicators, and the path with the highest comprehensive score is selected. As the forwarding path of traffic. Based on the Mininet simulation platform, use Ryu to control to generate probability model traffic to simulate data center network traffic. The experimental results show that the scheme improves the network throughput and bandwidth utilization, and improves the performance of the data center network.

Keywords: 5G Smart Grid, SDN, Traffic Dispatching Scheme, Multi-service Data Center, MFFC.

1. Introduction
Yunnan Power Grid Corporation actively promotes the construction of a safe, reliable, green, and efficient smart grid. The backbone communication network side of 35kV and above has a complete and efficient all-optical data network. On the distribution communication network side, due to the many points, wide areas, and massive equipment, and with the rapid development of large-scale distribution automation, low-voltage centralized reading, distributed energy access, and two-way user interaction, various types of grid equipment. The communication demands of terminals and power customers have exploded. Existing technologies cannot effectively support the considerable, measurable, and controllable distribution network terminals.

The safe, stable and efficient operation of the smart grid is of great significance to the country's sustainable development. Smart grid is built on the basis of an integrated, high-speed two-way communication network. The development of smart grids poses new challenges to the power communication network. It is necessary to adopt advanced and efficient emerging communication technologies to enrich the communication access methods, transform from simple passive satisfaction of business needs to active leadership of business needs, and provide more ubiquitous terminal access capabilities, Strong carrying capacity for diversified services, differentiated security isolation capabilities and more efficient and flexible operation management capabilities.

In response to the company’s urgent need to build a ubiquitous distribution communication network with safe and reliable access, flexible access, and two-way real-time interaction, based on the typical applications of 5G smart grids,
2. The Concept And Architecture of SDN for 5G Smart Grid

2.1. The Concept of SDN for 5G Smart Grid

Software-defined networking (SDN) is a transformative network architecture, which brings hope for breaking through the limitations of current network infrastructure. First, it breaks the vertical integration of traditional devices by decoupling the control plane and data plane of the network. Secondly, with the decoupling of the control plane and the data plane, the network switch becomes a mere forwarding device, and the control plane is implemented in the controller, which simplifies the implementation of policies and the reconfiguration of the network. Table 1 summarizes the related architecture and design features of the SDN controller and control platform, in order to provide a reference for the design of the 5G grid controller.

| Component               | OpenDaylight                  | OpenContrail                  | HP VAN SDN                      | Onix                     | Beacon                     |
|-------------------------|-------------------------------|-------------------------------|---------------------------------|--------------------------|----------------------------|
| Base network services   | Topology/Stats/SwitchManager, Host Tracker, Shortest Path Forwarding | Routing, Tenant Isolation    | Audit Log, Alerts, Topology, Discovery | Discovery, Multi-consistency Storage, Read | Topology, device manager, and |
| East/Westbound APIs     | —                             | —                             | Sync API                       | Distribution I/O module  | Not present                |
| Integration on Plugin   | OpenStack Neutron            | CloudStack, OpenStack        | OpenStack                      | —                        | —                          |
| Management              | GUI/CLI, REST API             | GUI/CLI                       | REST API, Shell / GUI          | —                        | Web                        |
| Northbound APIs         | REST, REST-CONF, Java APIs   | REST APIs (config-ration, operational) | REST API, GUI Shell            | Onix API (general purpose) | API based on               |
| Service abstraction layers | Service Abstraction Layer (SAL) | —                             | Device Abstraction API         | Network Information Base (NIB) Graph with | —                          |
| Southbound APIs         | OpenFlow, OVSDB, SNMP         | —                             | OpenFlow, L3 Agent, L2         | OpenFlow, OVSDB          | OpenFlow                   |

2.2. The Architecture of SDN for 5G Smart Grid

The topological structure of the data center network has two major types, namely the switch type topology and the server type topology. Fattree, VL2, c-Through and Helios are the classic solutions of the former, and the network connection and routing functions are completed by a large number of switches; the typical topology of the latter is DCell, FiConn, BCube And uFix, etc., only one switch is connected to a large number of servers to form a layer 0 network. In order to build a multi-path traffic scheduling environment in the data center, this paper chooses the classic Fattree topology. The FatTree topology is proposed by Fares et al. of the Massachusetts Institute of Technology (MIT) on the basis of improving the performance of the traditional tree structure. The entire topological network is divided into three levels: from top to bottom are the access layer (edge), the aggregation layer (aggregate) and the core layer (core). The aggregation layer switch and the access layer switch form a region (Pod). The switching equipment is all commercial switching equipment.

3. The Multi-factor Flow Control (MFFC) Strategy

This paper proposes a Multi-factor flow control (MFFC) strategy for the data center-oriented traffic dispatch of SDN in the smart grid. The core of the strategy is to find the best path for the data flow. The main steps are as follows: First, use the KSP (k takes 5) algorithm based on the number of hops to calculate 5 candidate paths, and consider the number of hops, bandwidth, and the number of large flows in the 5-hop path for path selection. After that, the controller starts a statistics thread for sending port information requests and flow table information requests regularly, and processing the port information responses and flow table information responses returned by the OpenFlow switch. The port information contains the id and port number of the switch, the amount of data sent by the port, and the amount of data received by the port. The system takes the sum of the amount of data sent and received by the port.
as the port forwarding amount, and takes 1 second as the request interval. The port forwarding amount in the two adjacent response messages is calculated to obtain the bandwidth occupation of the corresponding port (in Mbit/s).

After obtaining the bandwidth occupancy of each link and the number of large flows, the solution calculates the maximum bandwidth occupancy and the maximum number of large flows of all links on the path for each path in the alternative path, as the path bandwidth occupancy value and the number of large flows. After obtaining the bandwidth occupancy value of each path and the number of large flows, each path is traversed. If the hop count of the path is less than or equal to the hop count of \( x \) other paths, the path score is increased by \( x \); if the bandwidth occupancy value of the path is less than or equal to the occupancy value of \( y \) other large flows, the score of the path is increased by \( y \); if the number of large flows is less than or equal to the number of \( z \) other large flows, the score of the path is increased by \( z \). Finally, the path with the highest score is selected. If multiple paths have the same score, the path with the lower subscript value in the result list of the KSP algorithm and the front position is selected.

4. Simulation Environment Settings

4.1. Simulation Parameter Setting

In order to be consistent with other variables in the control experiment, each strategy uses the same experimental settings. The specific experimental settings are as follows:

1. Topological scale: Fat tree network with \( K=4 \);
2. Number of simulated hosts: 16;
3. Number of simulated switches: 20;
4. The bandwidth of the network link: 10Mbit/s;
5. The number of candidate paths of KSP algorithm (K): 5;
6. The network load monitoring cycle of the SDN controller: 2 seconds;
7. Flow entry timeout time: idle_timeout=5 seconds, hard_timeout=0 seconds;
8. Network performance test duration: 60 seconds.

4.2. Comparative Traffic Scheduling Strategy

In addition to the traffic scheduling strategy introduced in this paper, three other very well-known traffic scheduling strategies are also selected. The following briefly introduces these three traffic scheduling strategies:

1. ECMP: This experiment will implement ECMP by using flow entries and Select type group tables.
2. Hedera: According to the existing experimental verification, its performance is better than ECMP.
3. PureSDN: This scheme is the most traditional traffic scheduling strategy that uses SDN to implement the optimal path calculation scheme based on the number of path hops. It first uses KSP to select \( k \) paths with the least number of hops, and then selects the one with the lightest load as the best path.

4.3. Flow Model

This experiment simulated 8 traffic models of data center networks: random, stag_0.1_0.2, stag_0.2_0.3, stag_0.3_0.3, stag_0.4_0.3, stag_0.5_0.3, stag_0.6_0.2, stag_0.7_0.2. "Random" means a traffic model formed by the host that initiates a large flow by selecting the opposite end in a completely random manner; other traffic models take stag_0.1_0.2 as an example, and stag_0.1_0.2 means the one below the access layer switch. The traffic (that is, the traffic in the Pod) accounts for 10% of the total traffic, the traffic between different access layer switches in the Pod accounts for 20%, and the traffic between Pods accounts for 70%. Repeat the experiment 20 times for each flow model, and finally take the average of the experimental results as the final data.

In the experiment, IperfTCP is used to simulate a large flow, and each host sends a large flow. It is equivalent to that each host is the client of the traffic, and then the server is selected according to the probability set by different traffic models.
5. Simulation Result Analysis

Throughput rate represents the actual amount of data that a network passes through a certain network in a unit time, which best reflects the performance of the network (Fig. 1). When the proportion of traffic in the Pod is high, the average throughput of the four schemes is roughly the same as the standardized total throughput. But when the traffic in the Pod is relatively low, the performance of our proposed strategy is better than the other three strategies in terms of two indicators.

The ratio of the number of links that have transmitted data to the total number of links in the network (Fig. 2). The higher the utilization of the link, the more utilization of link resources, but when the utilization of the link is too high, it may cause excessive delay. In the simulation experiment of this article, because the network link is generally two-way, when calculating the link utilization and bandwidth utilization, a path is considered as two opposite directions, and the Pod and the access layer switch are not calculated separately. Direct links are included. In the case of low traffic in the Pod, the link utilization rate of ECMP is the lowest, and the MFFC scheme proposed in this paper is close to the link utilization rate of the Hedera and PureSDN strategies.

![Figure 1. Average throughput of the network.](image1)

![Figure 2. Network link utilization.](image2)

The bandwidth utilization rate of the network refers to the ratio of the bandwidth actually used by the link during data transmission to the total bandwidth. In general, low bandwidth utilization means that link resources are not fully utilized, and high utilization means that the link is very prone to congestion. As shown in Figure 3, in the case of high traffic in the Pod, the bandwidth utilization of the four traffic scheduling strategies is low. In the case of low traffic in the Pod, the links with lower utilization in the link bandwidth of ECMP account for a relatively high proportion, indicating that its link bandwidth resources are wasted a lot, while PureSDN and the MFFC strategy proposed in this paper are in the majority. In this case, there are more links with link bandwidth utilization between 40% and 60%, which shows that the strategy proposed in this paper has better performance than ECMP and Hedera algorithms under the performance index of bandwidth utilization.
Figure 3. Network link utilization.

6. Conclusion
This paper proposes a multi-service-oriented multi-factor traffic scheduling scheme. It improves and perfects the traffic scheduling strategy, and introduces a set of scoring mechanism. Based on the Mininet simulation platform, use Ryu to control to generate probability model traffic to simulate data center network traffic. The experimental results show that the scheme improves the network throughput and bandwidth utilization, and improves the performance of the data center network.

References
[1] X. Jingwen, W. Muqing and H. Xiaolan, "A Traffic Scheduling Scheme for Data Center Networks Based on SDN," 2019 IEEE 5th International Conference on Computer and Communications (ICCC), Chengdu, China, 2019, pp. 1417-1422.
[2] L. Cong and W. Yong-Hao, "Strategy of Data Manage Center Network Traffic Scheduling Based on SDN," 2016 International Conference on Intelligent Transportation, Big Data & Smart City (ICITBS), Changsha, 2016, pp. 29-34.
[3] L. Peizhe et al., "A Network Traffic Scheduling Strategy for Energy Storage Data Centers Based on SDN," 2019 IEEE 19th International Conference on Communication Technology (ICCT), Xi'an, China, 2019, pp. 1413-1417.
[4] H. Yao and W. Muqing, "SDN-based routing mechanism for cloud data centers," 2017 IEEE 3rd Information Technology and Mechatronics Engineering Conference (ITOEC), Chongqing, 2017, pp. 400-404.
[5] M. Amiri, H. Al Osman and S. Shirmohammadi, "Game-Aware and SDN-Assisted Bandwidth Allocation for Data Center Networks," 2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR), Miami, FL, 2018, pp. 86-91.
[6] S. Raghul, T. Subashri and K. R. Vimal, "Literature survey on traffic-based server load balancing using SDN and open flow," 2017 Fourth International Conference on Signal Processing, Communication and Networking (ICSCN), Chennai, 2017, pp. 1-6.

[7] T. A. Nguyen, D. Min, E. Choi and T. D. Tran, "Reliability and Availability Evaluation for Cloud Data Center Networks Using Hierarchical Models," in IEEE Access, vol. 7, pp. 9273-9313, 2019.

[8] E. Dutisseuil et al., "Traffic management in SDN-enabled optical packet switching intra-datacenter network," 2017 Opto-Electronics and Communications Conference (OECC) and Photonics Global Conference (PGC), Singapore, 2017, pp. 1-4.

[9] H. Yao, H. Li, C. Liu, M. Xiong, D. Zeng and G. Li, "Joint Optimization of VM Placement and Rule Placement towards Energy Efficient Software-Defined Data Centers," 2016 IEEE International Conference on Computer and Information Technology (CIT), Nadi, 2016, pp. 204-209.

[10] Maity, S. Misra and C. Mandal, "Traffic-Aware Consistent Flow Migration in SDN," ICC 2020 - 2020 IEEE International Conference on Communications (ICC), Dublin, Ireland, 2020, pp. 1-6.