DIN: A Bio-Inspired Distributed Intelligence Networking

Yufeng Li[1], Yankang Du[2](✉), Chenhong Cao[1], and Han Qiu[2]

[1] School of Computer Engineering and Science, Shanghai University, Shanghai 200023, China

[2] National Digital Switching System Engineering and Technology R&D Center, Zhengzhou 450002, China

duyankang@163.com

Abstract. Software-Defined Networking (SDN) is a promising method to simplify network management and facilitate network evolution. However, SDN is a logically centralized technology with global network-wide view. It faces the problem of scalability and reliability. In this paper, we propose a novel method termed as Distributed Intelligence Networking (DIN). DIN optimizes network management based on distributed coordination of multiple forwarding nodes like the coordination in bird flocking motion, it is a fully physically and logically distributed structure based on neighbor network-wide view. This architecture naturally has the advantage of scalability and reliability.

Keywords: Software-Defined Networking · Distributed Intelligence Networking · Neighbor network-wide view

1 Introduction

Software Defined Networking (SDN) offers the chance to introduce Artificial Intelligence (AI) to reduce operational costs and to improve performance and user experience [1]. Its main features are the centralized global network-wide view, programmability, and separation of the data plane and control plane. Now SDN has become a promising method and gets great attention.

As the network size increases, the centralized controller faces to many challenges [2, 3]. Firstly, there will be a bottleneck in the real-time communication capability of the controller with the network scale expanding. Secondly, for large scale networks, the controller needs to be able to process millions of flows per second without sacrificing the service quality. Thirdly, this control plane usually encounters the risk of single point failure. This will disconnect the controller and the forwarding elements.

Many research works have been done to overcome those issues [4–6]. DevoFlow [4] and Software-Defined Counters (SDC) [5] reduced the overhead of the control plane by delegating some work to the forwarding elements. Maestro [6] makes efforts on designing and deploying high performance controllers to increase the performance of the control plane.

Published by Springer Nature Switzerland AG 2019
X. Tang et al. (Eds.): NPC 2019, LNCS 11783, pp. 333–337, 2019.
https://doi.org/10.1007/978-3-030-30709-7_29

In this paper, we attempt to present a novel architecture termed as the Distributed Intelligence Networking (DIN). DIN optimizes network intelligence based on distributed coordination of neighbor forwarding elements, and acts like the flocking motion such as bird flocking in nature [7]. This can achieve global network coordination and optimization with distributed controller and distributed neighbor network-wide view.

2 Flocking Motion Introduction

Flocking motion exists in the nature in the form of flocks of birds, schools of fish, and so on. The study of consensus problem in flocking motion offers an alternative way to design the intelligent, coordinated and complex systems.

As shown in Fig. 1, DIN has the same physically distributed architecture as SDN flat architecture. However, its logical view is quite different from the SDN flat architecture as described in Onix [8] and HyperFlow [9]. For our proposed DIN, it does not need to maintain the global network-wide view. Each controller in DIN only needs to perceive the state of the neighbor controller. The DIN controller makes decisions based on the coordination control protocol, and adjusts its resources to achieve global coordinated behavior to realize intelligent improvement of network management, performance optimization and service quality.

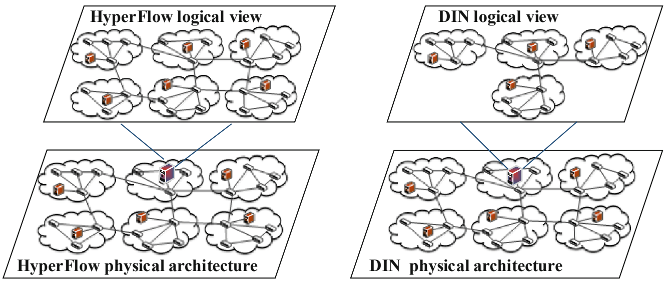


Fig. 1. Physical architecture and logical view in HyperFlow and DIN.

3 Architecture of DIN

There are two conditions to apply flocking motion to the Internet intelligence network. First, network individuals should have basic intelligent attributes. The network individuals can perceive the neighbor information, make decisions according to the control protocol, and make independent adjustments according to the decisions. Second, the network reaches convergence by coordinating among individuals based on the control protocol. This would show global intelligent behavior.

Figure 2 illustrates the architecture of the DIN forwarding node. The node architecture is divided into data-plane, control-plane and application-plane. The data-plane of DIN nodes, like the data-plane of SDN forwarding node, performs basic store and

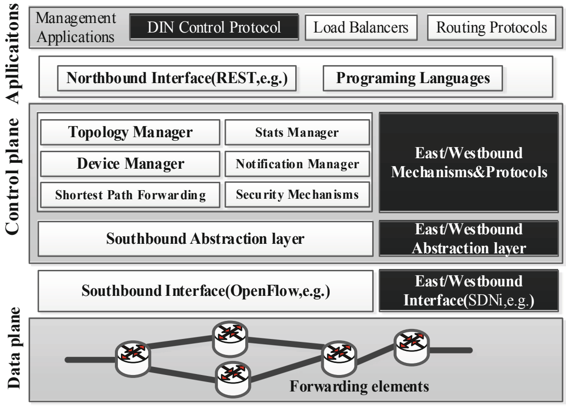


Fig. 2. DIN forwarding node architecture based on SDN

forwarding function. Besides this, it also has the data-plane adjustment function such as rate-limiting, traffic scheduling and so on.

The nodes of DIN can be implemented based on the existing SDN node structure. The DIN network works as follows: DIN node perceives the Coordination Situation (CoS) which means the current node situation, node resource and service request. By doing so, a neighbor network-wide view can be formed. Then it sends the CoS to the DIN control protocol. The DIN control protocol makes service decisions and issues adjustment commands to the data plane through NOS. The data plane adjusts resources for network services as ordered.

4 An Example for the DIN Protocol

A general Internet end-to-end service path is shown in Fig. 3, in which IP flows pass through forwarding nodes hop by hop and get a certain service.

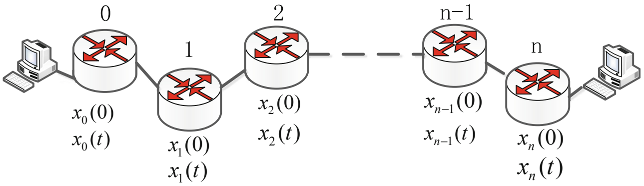


Fig. 3. A general end-to-end service path in network

A leader-followers multi-agent system is a particularly interesting topic in distributed multi-agent coordination theory, where the leader is an agent whose motion is followed by the other agent.

If we take the node labeled 0 in Fig. 3 as the leader, the other nodes indexed by $1, \ldots, n$ are the followers.

According to [10], the network can be controllable if it works under a coordination protocol with the dynamics of each agent

$$\dot{x}_i(t) = u_i(t), \quad i = 1, \ldots, n, \tag{1}$$

In protocol (1), the neighbor node service delay $x_j(t)$ and the communication delay T_{ji} serve as the node i perception information CoS, $u_i(t)$ is the decision rule, thus the node i conforming to DIN architecture can drive the node resource to adjustment and reach consensus on the service delay of the leader.

In this section, two example control protocols are designed based on DIN architecture, one is leader-followers delay guarantee protocol, and the other is leaderless delay guarantee protocol.

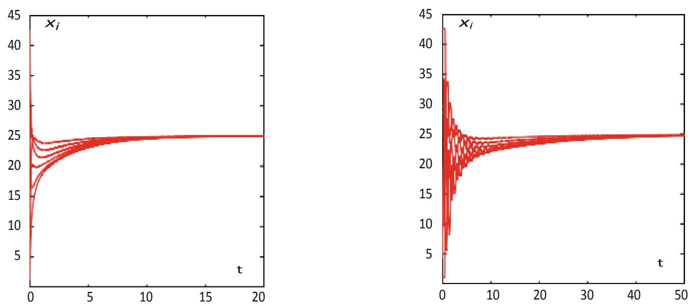
5 Simulation Results

As presented in Fig. 3, considering a leader-followers DIN network with control protocol (1), the topology is structured by

$$0 \rightarrow 1 \leftrightarrow 2 \leftrightarrow 3 \leftrightarrow 4 \leftrightarrow 5 \leftrightarrow 6$$

Node 0 works as leader with $x_0(0) = 25$ which is equal to R. It is assumed that the couplings gain is identical and equal to 5. All communication delays are set to 0.2. The initial states $x_i(0)(i = 1, 2, \ldots, 6)$ of the system are different, and each is chosen randomly.

Simulation results of (1) are plotted in Fig. 4. It is shown that under the two conditions without and with communication delay among nodes, all nodes along the path tend to reach a group consensus and remain stable at the initial service delay of the leader. That is to say, with the leader node providing the service level $x_0(0)$ required by the user, each following DIN node will automatically adjust under the protocol (1), and tend to converges to $x_0(0)$ of the leader.



(a) Without delays (b) With delays

Fig. 4. Simulation results for leader-followers delay guarantee protocol without and with communication delays.

We can also see that the convergence speed that Fig. 4(b) is obvious lower than the one of Fig. 4(a). This is mainly caused by communication delay among nodes.

Certainly, when the stable state of the group is broken at node i with x_i deviating the stable value $x_0(0)$, the protocol (1) will start a new round of coordination and tend to reach the group consensus again.

6 Conclusion

Inspired by the flocking motion in nature, this paper makes a preliminary exploration of the distributed coordination and control mechanism of Internet resources. We propose an intelligent network resource coordination and fitting scheme called DIN.

The characteristic and advantage of DIN is that it can keep distributed structure in physics and logic while introducing intelligent genes to the network. At present, for large-scale application, SDN moves from single controller to multi-controller, but the scalability and reliability problems caused by the global network-wide view and centralized control of SDN have not been solved well. The DIN proposed in this paper can be implemented completely based on the SDN node structure and specification, and can be deployed jointly with SDN, which can provide a new solution for improving the scalability and reliability of SDN.

Acknowledgments. This research was supported by the Research and Development Program in Key Areas of Guangdong Province under Grants 2018B010113001, and the National Natural Science Foundation of China under Grants 61502528.

References

1. Kim, H., Feamster, N.: Improving network management with software defined networking. IEEE Commun. Mag. **51**(2), 114–119 (2013)
2. Xie, J., Guo, D., Hu, Z., et al.: Control plane of software defined networks: a survey. Comput. Commun. Rev. **67**(1), 1–10 (2015)
3. Yeganeh, S., Tootoonchian, A., Ganjali, Y.: On scalability of software-defined networking. IEEE Commun. Mag. **51**(2), 136–141 (2013)
4. Curtis, A.R., Mogul, J.C., Tourrilhes, J., et al.: DevoFlow: scaling flow management for high performance networks. Comput. Commun. Rev. **41**(4), 254–265 (2011)
5. Mogul, J.C., Congdon, P.: Hey, you darned counters! Get off my asic!. In: Proceedings of the First Workshop on Hot Topics in Software Defined Networks, New York, USA, pp. 25–30 (2012)
6. Cai, Z., Cox, A.L., Ng, T.S.E.: Maestro: A System for Scalable OpenFlow Control. Rice University, Technical report (2011)
7. Vicsek, T., Zafeiris, A.: Collective motion. Phys. Rep. **517**(3–4), 71–140 (2012)
8. Koponen, T., Casado, M., et al.: Onix: a distributed control platform for large-scale production networks. In: Proceedings of the 9th USENIX Conference on Operating Systems Design and Implementation, Berkeley, USA, pp. 1–6 (2010)
9. Tootoonchian, A., Ganjali, Y.: HyperFlow: a distributed control plane for OpenFlow. In: Proceedings of the 2010 Internet Network Management Conference on Research on Enterprise Networking, USA (2010)
10. Rahmani, A., Ji, M., Mesbahi, M., et al.: Controllability of multi-agent systems from a graph-theoretic perspective. SIAM J. Control. Optim. **48**(1), 162–186 (2009)