Computer Networks 112 (2017) 24–35

Contents lists available at ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet

Density cluster based approach for controller placement problem in large-scale software defined networkings

Jianxin Liao[a,*], Haifeng Sun[a], Jingyu Wang[a], Qi Qi[a], Kai Li[a], Tonghong Li[b]

[a] State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing, 100876, China
[b] Department of Computer Science, Technical University of Madrid, Madrid, 28660, Spain

ARTICLE INFO

Article history:
Received 14 June 2016
Revised 23 September 2016
Accepted 28 October 2016
Available online 29 October 2016

Keywords:
SDN
Controller placement
Density based clustering
Resilience
Failure tolerance
Latency

ABSTRACT

Software Defined Networking (SDN) decouples control and data planes. The separation arises a problem known as the controller placement, i.e., how many and where controllers should be deployed. Currently, most works defined this problem as the multi-objective combinatorial optimization problem and used heuristic algorithms to search the optimal solution. However, these heuristic algorithms have the drawback of being easily trapped in local optimal solutions and consuming high time. In this paper, we propose an approach named as Density Based Controller Placement (DBCP), which uses a density-based switch clustering algorithm to split the network into several sub-networks. As switches are tightly connected within the same sub-network and less connected from the switches in other sub-networks, we deploy one controller in each sub-network. In DBCP, the size of each sub-network can be decided by the capacity of the controller deployed. Moreover, the optimal number of controllers is obtained according to the density-based clustering. We evaluate DBCP's performance on a set of 262 publicly available network topologies. The experimental results show that DBCP provides better performance than the state-of-the-art approaches in terms of time consumption, propagation latency, and fault tolerance.

1. Introduction

With the introduction of Software Defined Networking (SDN), the separation of control plane and data plane simplifies the networking management and improves its scalability [1]. The controller in control plane manages switches by providing them with rules that dictate their packet handling behavior. In a large-scale network, a good placement best utilizes existing network connectivity among the switches [2]. A fast response and reliable connection between the switch and the associated controller is a key point for SDN networks [3]. A single controller is hard to control all the switches in a large-scale SDN network, because the capability of the controller is limited and the propagation latency between the controller and switches is very large [3]. Currently, most researches aim to deploy multiple controllers at different locations to corporately control the whole data plane [2,4–6]. In this kind of architectures, the placement of multiple controllers becomes a critical problem.

As discussed in some related work, the controller placement is a complexity optimization problem [7,8], where the following factors should be taken into consideration whenever designing a placement strategy:

(1) **The latency of control signaling.** The switches receive the instructions on how to forward the new flows. Whenever the latency between controllers and switches reaches a threshold, the latency on the whole network will increase substantially. In this case, the controller processing latency is a non-negligible factor in the total round-trip latency [4].
(2) **The server capacity limitation.** Due to the constraints of the resources such as processors, memory, and access bandwidth, a commodity server only has the capacity to manage a limited number of switches. On the other hand, the overload of controllers may decrease the performance of SDN [5].
(3) **The required number of controllers.** In large-scale SDN networks, a large amount of switches in data plane construct a complex networks. It is difficult for administrators to figure out how many controllers should be deployed. Some work uses the traversal searching method to iteratively find the best performance number, which may lead to high time consumption [6].
(4) **Fault tolerance.** Unlike the traditional network architecture, the switches do not have control ability due to the split architecture of SDN. Each switch is assigned a controller. Therefore,

---

[*] Corresponding author.

E-mail addresses: jxlbupt@gmail.com (J. Liao), hfsun@bupt.edu.cn (H. Sun), wangjingyu@bupt.edu.cn (J. Wang), qiqi8266@bupt.edu.cn (Q. Qi), likai@ebupt.com (K. Li), tonghong@fi.upm.es (T. Li).

http://dx.doi.org/10.1016/j.comnet.2016.10.014
1389-1286/

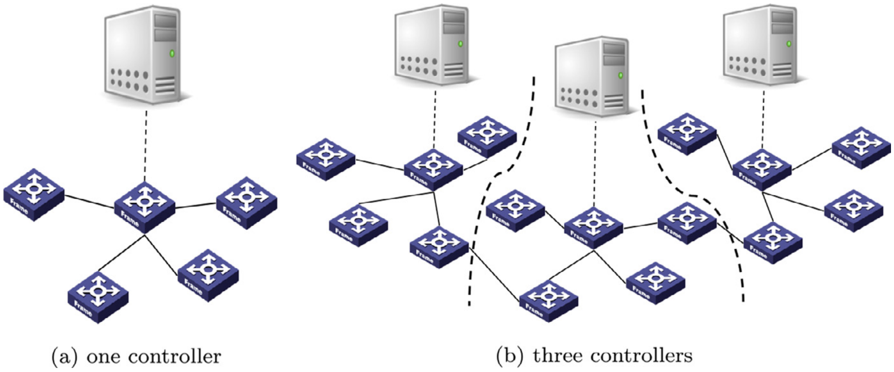
(a) one controller (b) three controllers

Fig. 1. Two examples of controller placement:(a) a star topology data plane and one controller. (b) a three star topologies constructed data plane and three controllers. The dotted curves represent the switches clustering result.

whenever a switch loses the connection to its controller, it will no longer receive any new routing instructions and thus drop all packets [2].

(5) **Inter-controller communication.** In multi-controller SDNs, each switch is controlled by a specific controller. If a controller wants to send messages to a switch controlled by another controller, the controllers need to communicate with each other [9]. Therefore, the inter-controller communication affects the performance in end-to-end communication between disparate switches controlled by different controllers.

To our best knowledge, there is no strategy to take into account all these factors for solving the controller placement problem. A well-known controller placement strategy, which is introduced in [3], motivated by minimizing the propagation latency between switches and controllers. Without considering the capacity limitation of the controller, this strategy is not always applicable. A good placement should minimize the propagation latency, whereas the load of each controller should not exceed its capacity. Yao et al. [5] defined a Capacitated Controller Placement Problem (CCPP) to consider the controller's capacity while minimizing the average propagation latency. Recently, some work, such as Pareto-based Optimal COntroller placement (POCO) [7] and Min-cut strategy [2], considered the reliability analysis of the networks. However, these methods cannot be applied to the CCPP problem.

Actually, the structure of the data plan is an important clue to find the optimal placement. As shown in Fig. 1a, a controller is deployed at the center switch in a star topology network, which is the optimal placement. In Fig. 1b, a data plane is constructed by three star topologies. In this network, the placement with three controllers is optimal in terms of latency and reliability.

For this purpose, we propose a new placement approach named as Density Based Controller Placement (DBCP) to solve the above problems. In this approach, we maintain a table of all the switch densities and the relevant information, which are newly defined in this article and will be discussed in detail at Section 3. Based on this table, the network is split into several sub-networks by suing a density-based clustering method [10] according to the network architecture. Then, the best placements of controllers can be selected by traveling all the candidate locations in each sub-network, and the SDN network is constructed by connecting all switches to their nearest controllers. Our proposed algorithm is a fast response and stable solution, which can be easily applied in real networks. Our critical contributions are as follows:

1) We use a fast density-based clustering method to cluster the data plane, where an optimal required number of controllers can be given. This method is faster for clustering, compared with the conventional iteration based clustering methods, such as k-means.
2) We propose a new strategy, which significantly improves the performance of control plane. Our experiment results show our approach is better than the state-of-the-art approaches.

The remainder of this paper is organized as follows: Section 2 surveys the related work. Section 3 defines the problem and proposes the design of DBCP. Section 4 presents some analysis regarding the recommended controller numbers and parameter. Section 5 provides the experiment results. Finally, discussions and conclusions are presented in Section 6.

2. Related work

Since Onix [11], Hyperflow [12], and Devoflow [13] were used as the distributed control architecture to solve the problem of scalability and reliability of SDN, more and more researchers have studied the controller placement problem. The controller placement problem (CPP) was first defined in [3]. The CPP problem mainly concerns on two questions: how many and where controllers should be deployed. It was proved as a NP-hard problem [3].

Heller et. al. [3] studied the best controller placement solutions that minimize the controller-to-switch propagation latency, which includes the average latency and the worst-case latency (or maximum latency). Sallahi et. al. [4] considered the cost of controllers, such as the cost of installing controllers, lining the controllers, and linking the controllers together. Both in [3] and [4], they all used a traversal method to search all the candidate solutions to find the optimal one. Traversal-based methods can provide the best performance solutions. However, the time consumption is extremely high in a large-scale network. As described in their papers, there are a lot of topologies that cannot be solved within 30 hours. Besides, Yao et. al. [5] considered that the load of the controller should not exceed its capacity, and defined a capacitated controller placement problem (CCPP). To solve the CCPP problem, they proposed an advanced capacity K-center algorithm [14] to search the best placement solutions, which travels different k values to find a least k to meet the capacity requirements. In [15], Yao et. al. found that the controller placement problem is a pre-planning problem, where the flow is varied. With the varied flow, some of the controllers may be overloaded. They proposed a method to place the controllers at hotspot where the switches carry the most flow. The switches with low flow can dynamically migrate from an overloaded controller to the other controller.

Some other works of CPP considered the reliability of networks. In [16], the authors defined a fault tolerant controller placement problem (FTCP) and found that the required controller number is positively correlated with the number of spokes (nodes with degree one) in a network. Based on these, they adopted a heuristic