MVA Course 2022/2023 Algorithms for network optimization and management: The Controller Placement Problem

Paul CHAUVIN
paulchauvin97@gmail.com

March 29, 2023

1 Introduction

Controller placement is a critical issue in network control as it plays a significant role in ensuring efficient network operation. The placement of controllers in a network directly affects the network's performance, including its ability to handle traffic, respond to failures, and provide Quality of Service (QoS) guarantees. In recent years, Software-Defined Networking (SDN) has emerged as a promising paradigm for network control, which separates the control plane from the data plane, allowing for dynamic network management and control. In SDN, controllers are responsible for managing the network and making decisions about how to forward traffic. Therefore, the placement of controllers is a crucial factor in determining the network's performance. Several studies have been conducted to investigate the controller placement problem [1][2]. These studies have shown that controller placement has a significant impact on network performance and that optimal controller placement can improve network efficiency and reduce deployment costs.

Optimizing controller placement is essential for achieving optimal network performance. When controllers are placed optimally, they can effectively manage network traffic and ensure that the network operates efficiently. This is particularly important in large-scale networks, where the volume of traffic and complexity of the network can lead to bottlenecks and performance issues. By optimizing controller placement, network administrators can ensure that the network operates at peak efficiency, reducing downtime and improving the overall user experience. Additionally, optimizing controller placement can help reduce network management costs by reducing the number of controllers needed to manage the network.

In this work, different clustering algorithms used to solve the controller placement problem in software-defined networks (SDN) are discussed. The controller placement problem involves selecting a prescribed number of objects as medians while respecting size constraints for the resulting clusters. Various clustering techniques such as p-median and capacitated p-median clustering, centered and capacitated-centered clustering, fuzzy clustering, and spectral clustering are explored. The advantages and limitations of each technique are discussed, and a comparison is made between fuzzy and spectral clustering. This work also discusses the use of Deep Reinforcement Learning (DRL) for controller placement in SDN, specifically the D4CPP and MACS policies.

2 Clustering for Controller Placement

Various types of algorithms have been proposed to solve the controller placement problem, including exact and heuristic algorithms. Exact algorithms guarantee optimal solutions, but they are computationally expensive and not suitable for large-scale networks. On the other hand, heuristic algorithms provide suboptimal solutions but are computationally efficient and can be applied to large-scale networks. Heuristic algorithms can be further classified into different categories, including random-based, greedy-based, and evolutionary-based algorithms. Random-based algorithms randomly generate solutions and evaluate them based on some criteria. Greedy-based algorithms make locally optimal decisions at each step of the algorithm to obtain a globally optimal solution. Evolutionary-based algorithms use evolutionary principles, such as mutation and selection, to generate and evaluate solutions.

Several studies have investigated the use of different types of algorithms to solve the controller placement problem. For example, [3] proposed a heuristic algorithm based on a genetic algorithm to solve the controller placement problem. [4] proposed a fast and scalable algorithm based on a greedy heuristic. [5] conducted a survey of different types of algorithms used to solve the controller placement problem in wireless software-defined networks. These studies have shown that different types of algorithms can be used to solve the controller placement problem, and the choice of algorithm depends on the specific requirements of the network. In what follows, we will focus on some of the clustering algorithms that demonstrated top results in several papers.

2.1 P-median and capacitated p-median clustering

One of the most widely studied clustering problems is the p-median problem, which involves selecting p objects as medians to minimize the total distance between all objects and their nearest medians. An extension of this problem is the capacitated p-median problem (CPMP), which is a more practical version of the p-median problem as it considers size constraints for clusters. The CPMP is known to be NP-hard, and finding a feasible solution to it is NP-complete. Many heuristic and exact solution approaches have been proposed for this problem [6], but the iterated reduction matheuristic algorithm (IRMA) [7] is considered the most efficient for instances involving up to 5,000 objects. However, the IRMA algorithm becomes computationally expensive for larger instances.

To tackle the computational complexity of the CPMP for larger instances, a new matheuristic algorithm was proposed in this paper [8] which algorithm is designed to focus on subproblems that have the potential to significantly improve the objective function value, utilize binary linear programming to maintain feasibility with respect to capacity constraints, and apply efficient data structures to avoid computing a large number of pairwise distances. The proposed matheuristic consists of two phases: a global optimization phase that involves all objects and a local optimization phase that involves only a subset of objects. The algorithm is tested on large-scale instances and demonstrates significant improvements in efficiency compared to previous benchmark approaches from the literature.

The paper [8] shows that its algorithm is specifically designed to handle large-scale instances of the capacitated p-median problem. The algorithm's decomposition strategies are optimized to focus on subproblems with the potential for significant improvement in the objective function value, while binary linear programming ensures the feasibility of the solution with respect to capacity constraints. Additionally, efficient data structures are employed to avoid computing large numbers of pairwise distances. The algorithm is comprised of a global optimization phase and a local optimization phase and has demonstrated significant improvements in computational efficiency for instances involving more than 5,000 objects.

2.2 Centered and capacitated-centered clustering

The capacitated-centered clustering problem (CCCP) [9] is a variant of the classical clustering problem that takes into account both capacity constraints and the cost of assigning clients to clusters. This problem involves partitioning clients into a fixed number of clusters, while minimizing the sum of the cost of assigning clients to clusters and the distance from each client to the center of its assigned cluster. The CCCP is particularly useful for networks with unevenly distributed traffic demands and controllers with different capacities. To solve large-scale instances of the CCCP, a matheuristic approach [8] has been proposed, which has been shown to outperform existing approaches [10][11] in terms of solution quality and computation time. The approach aims to minimize the maximum distance between any switch and its assigned controller, subject to capacity constraints. By using heuristics, the matheuristic approach achieves good results in terms of minimizing controller load and reducing network latency in large-scale networks.

2.3 Fuzzy clustering

Fuzzy Logic (FL) [12] based soft computing techniques provide a novel approach to making decisions regarding the architectural issues of the controller. The Controller Placement Problem (CPP) in SDN is an important management decision that influences all aspects related to the control plane. It involves finding the optimal number and placement of controllers in the network for effective switch-to-controller association. Poorly placed controllers result in long flow setup times and limit network availability and convergence time. The FL-based model proposed in this paper [12] provides a customized framework to solve CPP based on imprecise and incomplete data. The model considers application requirements, network conditions, and policy-based factors like controller resilience and availability.

The state of the art in fuzzy clustering for controller placement in SDN involves the use of different clustering techniques such as Fuzzy C-Means (FCM) [13], K-Means, and Hierarchical clustering. FCM-based techniques have been used to optimize controller placement for better network performance by considering network traffic patterns and network topology. The proposed FL-based model [12] provides a significant improvement over existing fuzzy clustering techniques for controller placement in SDN. By considering additional parameters like policy-based factors, controller resilience, and availability, which were not considered in previous techniques, the model presented in [12] is flexible and customizable and can be used in any type of SDN, irrespective of network size and management policy restrictions. The model also provides a framework for dynamic network management based on application requirements and network conditions. The preliminary experimental results showed promising performance of the FL-based model in resolving CPP.

2.4 Spectral clustering

Spectral clustering is a technique used for partitioning large networks into smaller ones, which is necessary when deploying software-defined networking (SDN) in real networks. This partitioning of a large network into multiple SDN domains improves privacy, scalability, incremental deployment, and security. The SDN partitioning algorithm uses a spectral clustering placement algorithm that efficiently and accurately partitions the network into small SDN domains. To address the SDN controller placement problem, the best controller placement should minimize propagation delays and improve reliability. The authors of [1] previously examined the impacts of controller placements on latency and worst-case latencies on real topologies, but they treated WAN as a whole rather than multiple SDN domains and ignored the reliability of each controller. While propagation latency is a significant design metric, reliability, and load-balancing design are also essential operational SDN parts. In [14] and [15], two solutions to exchange control and application information across multiple SDN domains were proposed, and in [16], the spectral clustering method is used to partition a WAN topology into several small SDN domains, and the SDN controller placement problem is defined in terms of latency, load balancing, and reliability.

The author of this paper [17] presents a Multi-Controllers Elastic Placement (MCEP) algorithm based on reliability and delay to improve the existing multi-controller deployment scheme's reliability and reduce delay. The MCEP algorithm uses spectral clustering to transform the multi-controller deployment problem into the row vector classification problem of the network topology matrix. The k-medoids algorithm based on simulated annealing is used to classify the row vectors to achieve flexible deployment of multiple controllers. Compared with the previous algorithms in the literature, the simulation results show that the MCEP algorithm improves the reliability of control paths by 17% on average while guaranteeing low delay.

In summary, the use of spectral clustering for controller placement in SDN is a novel approach that efficiently and accurately partitions large networks into multiple SDN domains, improving reliability, and load-balancing design. The proposed MCEP algorithm improved the existing multi-controller deployment scheme's reliability and reduces delay. Spectral clustering and MCEP are complementary approaches that provide essential operational components of SDN. The performance evaluations of both spectral clustering and MCEP algorithms have demonstrated their effectiveness in improving the reliability and delay of the SDN control networks. However, the authors indicate that future research can focus on further enhancing the performance of these algorithms and incorporating more metrics to improve the SDN controller placement problem's overall efficiency.

2.5 Comparison between Fuzzy and Spectral clustering

As written previously, when it comes to clustering algorithms, two popular options are fuzzy clustering and spectral clustering. One key advantage of fuzzy clustering is its flexibility. Unlike spectral clustering, fuzzy clustering allows for overlapping clusters, which means that data points can belong to multiple clusters with different degrees of membership. This flexibility can be especially useful when dealing with complex datasets where the boundaries between clusters are not well-defined.

In terms of the algorithm itself, fuzzy clustering is based on iterative algorithms that minimize a cost function. This means that the algorithm adjusts the cluster assignments of each data point during each iteration until the cost function is minimized. On the other hand, spectral clustering is based on linear algebra techniques that compute the eigenvectors of a graph Laplacian matrix. This approach can be very effective for datasets with clear geometric structures, but it can be more limited when it comes to more complex data.

When it comes to performance, fuzzy clustering is generally faster and more computationally efficient than spectral clustering, particularly for large datasets. This is because fuzzy clustering algorithms are often less complex and require fewer computations than spectral clustering algorithms. However, it's worth noting that spectral clustering has been shown to perform better than fuzzy clustering in some cases, particularly when the data has a clear geometric structure. In these cases, spectral clustering may be a better choice despite its higher computational cost.

Overall, the choice between fuzzy clustering and spectral clustering will depend on the specific needs of the application and the characteristics of the dataset. Fuzzy clustering may be a better option when dealing with complex datasets that require flexible cluster assignments and when computational efficiency is a concern. Spectral clustering may be a better option when dealing with datasets with clear geometric structures and when accuracy is the top priority.

3 Deep Reinforcement Learning for controller placement in SDN

To address the controller placement problem in SDN, researchers have turned to Deep Reinforcement Learning (DRL), which can explore the solution space and learn from the feedback generated during exploration. DRL algorithms, such as the Deep Q-Network (DQN) empowered Dynamic flow Data Driven approach for Controller Placement Problem (D4CPP) [18], can fully consider flow fluctuations, data latency, and load balance, leading to an optimized balance among these metrics. D4CPP integrates historical network data learning into the controller deployment and real-time switch-controller mapping decision, making it adaptable to the dynamic network environment with flow fluctuations. Simulations have shown that D4CPP outperforms traditional approaches by 13% in latency and 50% in load balance on average when the latency and load balance are equally weighted in SDN systems.

In distributed SDNs, multiple physical SDN controllers are implemented to balance centralized control, scalability, and reliability requirements. However, existing works typically only aim to eliminate anomalies arising from the inconsistencies in different controllers' network views, while neglecting performance aspects of controller synchronization designs with respect to SDN applications. To address this issue, researchers have formulated the controller synchronization problem as a Markov decision process (MDP) and applied reinforcement learning techniques combined with deep neural networks (DNNs) to train a smart, scalable, and fine-grained controller synchronization policy called the Multi-Armed Cooperative Synchronization (MACS) [19]. MACS uses DNNs to abstract latent patterns in the distributed SDN environment, leading to significant performance improvements over ONOS and greedy SDN controller synchronization heuristics.

In summary, Deep Reinforcement Learning and specifically, the D4CPP and MACS policies, have been proposed to address the challenges of controller placement and synchronization in SDN systems. D4CPP considers flow fluctuations, data latency, and load balance, leading to an optimized balance among these metrics, while MACS aims to maximize the performance enhancements brought by controller synchronizations. Both policies have shown significant performance improvements over traditional approaches, making them promising solutions for optimizing SDN systems.

4 Conclusion

In conclusion, clustering algorithms and machine learning techniques are critical for optimizing controller placement and network performance in SDN systems. Spectral clustering and fuzzy clustering are effective for optimizing metrics such as latency and load balance, while the D4CPP approach, which uses DRL, outperforms standard algorithms in terms of latency and load balance. The combination of clustering algorithms and machine learning techniques highlights the importance of data-driven approaches for optimizing network performance, and their continued development and application will be essential for ensuring optimal performance and scalability in increasingly complex SDN systems.

References

[1] Brandon Heller, Rob Sherwood, and Nick McKeown. The controller placement problem. In *Proceedings of the first workshop on Hot topics in software defined networks*. ACM, August 2012.

[2] Md. Tanvir Ishtaique ul Huque, Guillaume Jourjon, and Vincent Gramoli. Revisiting the controller placement problem. In *2015 IEEE 40th Conference on Local Computer Networks (LCN)*, pages 450–453, 2015.

[3] Abeer A. Z. Ibrahim, Fazirulhisyam Hashim, Aduwati Sali, Nor K. Noordin, and Saber M. E. Fadul. A modified genetic algorithm for controller placement problem in sdn distributed network. In *2021 26th IEEE Asia-Pacific Conference on Communications (APCC)*, pages 83–88, 2021.

[4] Rakesh Salam and Ansuman Bhattacharya. Efficient greedy heuristic approach for fault-tolerant distributed controller placement in scalable sdn architecture. *Cluster Computing*, 25, 08 2022.

[5] Si-Kee Yoon, ZIA Khalib, Naimah Yaakob, and Amiza Amir. Controller placement algorithms in software defined network - a review of trends and challenges. volume 140, page 01014, 01 2017.

[6] Jack Brimberg and Zvi Drezner. Solving multiple facilities location problems with separated clusters. *Operations Research Letters*, 47(5):386–390, September 2019.

[7] Fernando Stefanello, Olinto Araújo, and Felipe Martins Müller. Matheuristics for the capacitated p-median problem. *Int. Trans. Oper. Res.*, 22:149–167, 2015.

[8] Mario Gnägi and Philipp Baumann. A matheuristic for large-scale capacitated clustering. *Computers &amp Operations Research*, 132:105304, August 2021.

[9] Marcos Negreiros and Augusto Palhano. The capacitated centred clustering problem. *Computers Operations Research*, 33:1639–1663, 06 2006.

[10] Jianxin Liao, Haifeng Sun, Jingyu Wang, Qi Qi, Kai Li, and Tonghong Li. Density cluster based approach for controller placement problem in large-scale software defined networkings. *Computer Networks*, 112:24–35, January 2017.

[11] Hailan Kuang, Yiwen Qiu, Ruifang Li, and Xinhua Liu. A hierarchical k-means algorithm for controller placement in sdn-based wan architecture. In *2018 10th International Conference on Measuring Technology and Mechatronics Automation (ICMTMA)*, pages 263–267, 2018.

[12] Shaleeza Sohail and Aasia Khanum. Fuzzy approach to controller placement problem: Fuzzy controller (fc). In *2018 International Conference on Advances in Big Data, Computing and Data Communication Systems (icABCD)*, pages 1–5, 2018.

[13] Vidya Sagar Thalapala and Koppala Guravaiah. Fcmcp: Fuzzy c-means for controller placement in software defined networking. *Procedia Computer Science*, 201:109–116, 2022. The 13th International Conference on Ambient Systems, Networks and Technologies (ANT) / The 5th International Conference on Emerging Data and Industry 4.0 (EDI40).

[14] Hongtao Yin, Haiyong Xie, Tina Tsou (Ting ZOU), Diego R. Lopez, Pedro Andres Aranda, and Ron Sidi. SDNi: A Message Exchange Protocol for Software Defined Networks (SDNS) across Multiple Domains. Internet-Draft draft-yin-sdn-sdni-00, Internet Engineering Task Force, June 2012. Work in Progress.

[15] Pingping Lin, Jun Bi, and Yangyang Wang. East-west bridge for sdn network peering. In Jinshu Su, Baokang Zhao, Zhigang Sun, Xiaofeng Wang, Fei Wang, and Ke Xu, editors, *Frontiers in Internet Technologies*, pages 170–181, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

[16] Peng Xiao, Wenyu Qu, Heng Qi, Zhiyang Li, and Yujie Xu. The sdn controller placement problem for wan. In *2014 IEEE/CIC International Conference on Communications in China (ICCC)*, pages 220–224, 2014.

[17] Jie Lu, Zhang Zhen, and Tao Hu. Spectral clustering based approach for controller placement problem in software defined networking. *Journal of Physics: Conference Series*, 1087:042073, September 2018.

[18] Yiwen Wu, Sipei Zhou, Yunkai Wei, and Supeng Leng. Deep reinforcement learning for controller placement in software defined network. In *IEEE INFOCOM 2020 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pages 1254–1259, 2020.

[19] Ziyao Zhang, Liang Ma, Konstantinos Poularakis, Kin K. Leung, Jeremy Tucker, and Ananthram Swami. Macs: Deep reinforcement learning based sdn controller synchronization policy design, 2019.