UDC 004.738.5:004.7: 621.39 **doi: 10.26906/SUNZ.2025.4.186-189**

Stanislav Myhal

National University "Yuri Kondratyuk Poltava Polytechnic", Poltava, Ukraine

MATHEMATICAL MODEL OF A DATA FLOW MANAGEMENT SYSTEM IN A CLUSTER-BASED MULTICONTROLLER SDN

Abstract. The problem of load balancing among controllers and ensuring network state consistency is one of the key challenges in multicontroller Software-Defined Networks (SDN). To address this issue, it is essential to develop adequate mathematical models that accurately describe the processes of data flow management. The subject of this study is the data flow control system within a clustered multicontroller SDN environment. **The objective** of the research is to develop a mathematical model that accounts for the dynamic clustering of SDN controllers when receiving data from network switches, enabling balanced load distribution across controllers and facilitating rapid reallocation of links between controllers and switches**. The following results** have been obtained. A three-tier hierarchical clustering approach for control and data transmission/reception devices within the system has been proposed. The relationships between controllers and OpenFlow switches within the developed clustered architecture have been formalized. Computational analysis of the control element loads has been performed. **Conclusion.** The proposed mathematical model enables efficient load balancing among controllers and ensures dynamic redistribution of workloads in the event of controller failure.

Ключові слова: telecommunication network, communication network, controller, airborne network, OpenFlow switch, 5G standard, SDN.

Introduction

Modern computer networks are becoming increasingly complex due to the growing volume of data transmission, the diversity of services, and the demand for flexible resource management. One of the most effective approaches to organizing network infrastructure is Software-Defined Networking (SDN). The core concept of SDN lies in the separation of the control plane from the data forwarding plane, enabling centralized network management through software-based tools. This paradigm enhances the flexibility, scalability, and adaptability of network systems, while also simplifying the implementation of security policies and traffic optimization mechanisms.

However, as SDN networks expand in scale, there arises a need for distributed control systems, where control functions are no longer concentrated in a single node but are distributed among multiple controllers. Consequently, cluster-based multicontroller SDN architectures have gained significant popularity. In such architectures, multiple controllers operate collaboratively to ensure load balancing, fault tolerance, and reduced latency. Each controller is responsible for a specific segment of the network, yet must continuously interact with others to maintain network state consistency.

At the same time, data flow management in cluster-based multicontroller SDNs presents several challenges related to coordination among controllers, synchronization of routing tables, and maintenance of network stability. Additional difficulties arise from traffic heterogeneity, inter-controller communication delays, and the need for dynamic resource redistribution. Without an efficient flow management mechanism, these systems may experience performance degradation, excessive load concentration on individual controllers, or even instability, ultimately leading to network service deterioration.

To address these challenges, it is essential to develop adequate mathematical models that describe the processes of data flow management in cluster-based multicontroller SDNs. Such models make it possible to formalize controller interactions, determine equilibrium or steady states of the system, and evaluate the impact of network parameters on operational efficiency. Mathematical modeling thus provides a foundation for analytical studies, control algorithm optimization, network behavior prediction, and overall system stability improvement.

Therefore, the development of a mathematical model of data flow management in a cluster-based multicontroller SDN represents a crucial step toward enhancing the mechanisms of distributed control, improving network reliability, and ensuring the efficient performance of modern software-defined networks.

1. Review of Current Research

Recent scientific studies have devoted considerable attention to the development of Software-Defined Networking (SDN), which is recognized as a promising approach for building scalable and flexible network systems. The fundamental principles of the SDN concept and the OpenFlow architecture were established in early works, where the separation of the control plane from the data forwarding plane was first proposed, enabling centralized network management [1].

Subsequent research has focused on analyzing the architecture, advantages, and challenges of SDN. In particular, studies have examined issues of security, controller performance, scalability, and the efficiency of inter-controller communication [2]. It has been demonstrated that a purely centralized control model poses certain limitations, leading to the development of distributed and multicontroller architectures [3].

One of the major challenges in multicontroller SDNs is load balancing among controllers and network state synchronization. A number of studies have proposed synchronization mechanisms between controllers and assessed their impact on request processing delays and system stability [4]. Other works have focused on dynamic controller assignment algorithms that allocate controllers to network nodes based on distance and communication latency, thereby improving the efficiency of data flow distribution [5].

An important research direction in SDN development is controller clustering, which enhances fault tolerance and ensures network scalability. Studies in this area have proposed hierarchical clustering models and described mechanisms for inter-cluster coordination [6]. Analytical models have also been developed for evaluating controller performance and identifying critical load points [7].

A significant contribution has been made in the field of mathematical modeling of data flow management processes. The application of queueing theory has enabled the formalization of relationships among traffic volume, controller performance, and request processing delays. Other researchers have proposed dynamic interaction models of controllers represented by systems of differential equations, allowing for the analysis of transient and steady states of the network. Furthermore, adaptive mathematical models have been developed to predict network traffic behavior and optimize flow distribution [8].

Thus, the literature review demonstrates that the mathematical representation of data flow control processes in cluster-based multicontroller SDNs remains a highly relevant research topic. The development of such models is essential for further optimization of controller architectures, enhancement of system stability, and improvement of network resource management efficiency.

Research Objective: To develop a mathematical model that accounts for dynamic clustering of SDN controllers during data acquisition from network switches, enabling balanced load distribution across controllers and real-time reallocation of links between controllers and switches.

2. Hierarchical Clustering for Multicontroller SDN Networks

The proposed hierarchical clustering algorithm operates in a dynamic mode, adapting to changes in network conditions in real time. The input parameters for the algorithm are the network traffic characteristics and data flow requests. The set of control devices is divided into clusters of SDN controllers, with each cluster assigned a main SDN controller responsible for coordination. Fig. 1 illustrates the hierarchical structure of a clustered SDN network.

The SDN network consists of a centralized alpha-controller (Cα) and several clusters of regular SDN controllers, with an equal number of controllers in each cluster. Each cluster has a main node, referred to as a beta-controller (Cβ). The beta-controller is responsible for cluster configuration and load balancing among the controllers within that cluster.

In Fig. 1, the following designations are used: β – beta-controller; *c* –SDN controller, a member of the cluster; *OFS* – OpenFlow Switchboard.

The primary objective of the developed clustering algorithm is to balance the load among controllers within the set of control devices and to mitigate issues related to controller failures. This approach reduces the overall operational cost of the SDN network while enhancing its availability, reliability, and fault tolerance.

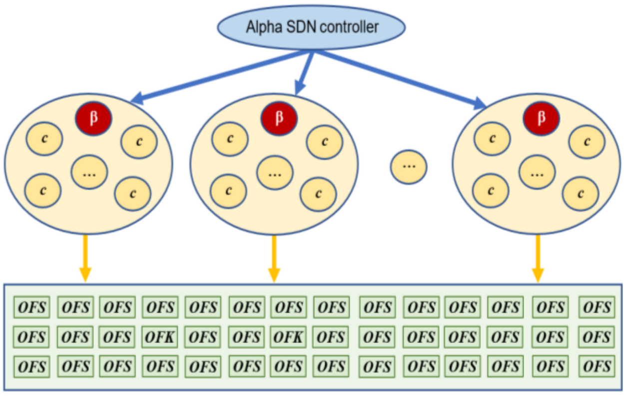

Fig. 1. General structure of a clustered multicontroller SDN network

The clustering process can be divided into three main phases: 1) controller configuration; 2) switch connection establishment; 3) formation of the steady state.

During the cluster formation phase, the alpha-controller initializes the SDN clusters by selecting beta-controllers and the member controllers for each cluster. The controller with the lowest expected load is selected as the beta-controller, assuming the role of coordinator for its respective cluster. All clusters are formed homogeneously, meaning that each contains the same number of SDN controllers. Once the clusters are established, the controller configuration phase concludes, and the switch connection phase begins.

Each cluster is then assigned a group of OpenFlow switches. During the dynamic allocation of switches among the member controllers, the system aims to achieve near-optimal delay and cost parameters.

At this stage, each SDN controller assumes a specific role – either as a regular cluster member or as a beta-controller – and establishes optimal connections with the assigned OpenFlow switches.

The process then proceeds to the steady-state phase. Each cluster member controller manages its connected OpenFlow switches and maintains the flow tables generated by those switches. Each beta-controller supervises both its assigned OpenFlow switches and the member controllers of the cluster, effectively serving as the head node of that cluster.

The beta-controller continuously monitors the traffic among the nodes within its cluster and reports to the SDN alpha-controller. When the beta-controller detects load imbalance among cluster members — for instance, when certain controllers are overloaded beyond a specified percentage of their maximum capacity, while others are underloaded — it initiates a load-balancing process among the cluster controllers through the following actions: 1) transferring the role of head node to a member controller with the lowest current load, which becomes the new beta-controller; 2) reassigning the OpenFlow switch connections among the cluster's SDN controllers. The reconnection process consists of the same three phases as the initial clustering procedure. However, unlike the overall clustering process, which is controlled by the alpha-controller, the reconnection process is managed locally by the beta-controller within its cluster. These dynamic processes are executed automatically whenever network imbalance is detected, continuing until the load among clusters is equalized. The alpha-controller collects reports from all beta-controllers and monitors inter-cluster traffic. If a significant imbalance is detected between clusters, it triggers a new round of the clustering procedure, forming updated clusters using the previously defined phases. This dynamic hierarchical approach ensures continuous load balancing among distributed SDN controllers throughout the entire network operation period, enabling optimal utilization of computational resources and maintaining high network performance and stability.

2. Development of a Mathematical Model for the Functioning of a Multicontroller SDN

The set of deployed SDN controllers that constitute the collection of control devices can be represented as a vector *C*, defined as follows:

$$C = \{C_1, C_2, \ldots, C_i, \ldots C_N\}, \quad (1)$$

where C_i denotes an SDN controller with index *i*, and *N* is the total number of deployed SDN controllers.

The controllers responsible for managing clustering are also included in the set *C*, that is:

$$C_\alpha \in C; \quad C_\beta \in C. \quad (2)$$

The total number of clusters created is denoted as *M*, and this value may vary as the system state changes. The set of beta-controllers constitutes the subset C_β, which is defined as follows:

$$C\beta = \{C_{\beta 1}, C_{\beta 2}, \ldots, C_{\beta j}, \ldots, C_{\beta M}\},$$
$$\operatorname{card} C\beta = M; \quad C\beta \subset C; \quad (3)$$
$$C_{\beta j} \in C \quad \forall j \in \overline{1, M}.$$

Each formed cluster has an associated set of member controllers, the length of which is *L*. The set of member controllers for each *j*-th cluster can be expressed as:

$$Cj = \{C_{j1}, C_{j2}, \ldots, C_{j\ell}, \ldots, C_{jL}\},$$
$$\operatorname{card} Cj = L \quad \forall j \in \overline{1, J}; \quad Cj \subset C; \quad (4)$$
$$C_{j\ell} \in C \quad \forall \ell \in \overline{1, L}.$$

Let the set of SDN control devices manage *K* OpenFlow switches deployed in the network, distributed among the controller clusters (as shown in Fig. 1).

Each OpenFlow switch is connected to a specific SDN controller, with the allocation determined by the controller placement algorithm. The set of deployed OpenFlow switches can be represented as:

$$S = \{S_1, S_2, \ldots, S_k, \ldots S_K\}. \quad (5)$$

Each ℓ - th controller of the *j*-th SDN cluster has a specific set of connected OpenFlow switches, which can be defined by the following set:

$$Sj\ell = \{S_{j\ell 1}, S_{j\ell 2}, \ldots, S_{j\ell r}, \ldots S_{j\ell R_{j\ell}}\},$$
$$\operatorname{card} Sj\ell = R_{j\ell}; \quad Sj\ell \subset S; \quad (6)$$
$$S_{j\ell r} \in S \quad \forall \ell \in 1, L; \quad \forall j \in 1, J,$$

where $S_{j\ell r}$ is the *r*-th OpenFlow switch currently connected to the ℓ-th controller of the *j*-th SDN controller cluster.

Connections between OpenFlow switches and SDN controllers are represented by a Boolean switching matrix *W*, where rows correspond to SDN controllers and columns correspond to OpenFlow switches. The matrix *W* therefore reflects the total number of switches connected to each SDN controller. If the row corresponding to a specific controller consists entirely of zeros, that controller acts as an alpha-controller, which communicates exclusively with beta-controllers. The beta-controllers, in turn, manage the cluster controllers and have the lowest number of ones in the matrix *W* among all elements of their respective clusters.

Next, consider an example fragment of an SDN network that includes five SDN controllers and six OpenFlow switches, distributed across two clusters:

$$C = \{C_1, C_2, C_3, C_4, C_5\}; \; C_\alpha = C_1; \; C_{\beta 1} = C_2;$$
$$C_{\beta 2} = C_4; \quad C_{11} = C_3 \in C1; \quad C_{21} = C_5 \in C2;$$
$$\operatorname{card} C1 = \operatorname{card} C2 = 2; \quad S = \{S_1, S_2, S_3, S_4, S_5, S_6\}.$$

For this fragment of the SDN network, the Boolean switching matrix *T* may be represented as follows:

$$W = \begin{array}{c} \\ C_1 \\ C_2 \\ C_3 \\ C_4 \\ C_5 \end{array} \begin{array}{c} \begin{array}{cccccc} S_1 & S_2 & S_3 & S_4 & S_5 & S_6 \end{array} \\ \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix} \end{array}. \quad (7)$$

One of the ways to evaluate the performance of a controller is by measuring its response time, which is mainly affected by queueing delay. The controllers can be modeled using a multi-server *M*/*M*/*s* queueing model, where each controller is assumed to have *s* cores. The transmitted packets arrive at the controller with a defined rate corresponding to a Poisson process, forming a single queue on the selected controller. The average response time T_i of controller Ci is the sum of the waiting time in the queue and the processing time, and it can be calculated using the Erlang formula as a function of the data packet arrival rate λ_i and the service rate μ:

$$T_i(\lambda) = P(s, \lambda_i/\mu) / (s \cdot \mu - \lambda_i) + 1/\mu, \quad (8)$$

where $P(s, \lambda_i/\mu)$ is the probability that all servers in the system are occupied and any incoming packet will be placed in the corresponding queue. This probability can be calculated as follows:

$$P(s, \lambda_i/\mu) = \frac{\left((s \cdot \rho)^s / s!\right) \cdot (1-\rho)^{-1}}{\sum_{k=0}^{s-1} \left((s \cdot \rho)^k / k!\right) + \left((s \cdot \rho)^s / s!\right) \cdot \frac{1}{1-\rho}} =$$
$$= 1 \Big/ \left(1 + \frac{1}{1-\rho} \cdot \frac{s!}{(s \cdot \rho)^s} \cdot \sum_{k=0}^{s-1} \left((s \cdot \rho)^k / k!\right)\right), \quad (9)$$

where ρ is the server utilization factor, which indicates the stability of the system and is calculated as follows:

$$\rho = \lambda_i / (s \cdot \mu). \tag{10}$$

The system has a stable distribution only if the value of ρ is less than one. When the number of incoming requests in the queue exceeds the number of server cores, the controller operates at maximum throughput. The arrival rate of requests to a controller can be calculated as the sum of the average arrival rates of requests from all switches connected to that controller, i.e.

$$\lambda_i = \sum_{k_s} \lambda_{k_s} \tag{11}$$

where the summation is performed over all switches currently connected to the considered controller C_i. The average load on controller C_i can be calculated as the average number of requests that are queued for processing and those being processed. Using (8), the average load on controller C_i can be calculated as follows:

$$L_i(\lambda_i) = s \cdot \rho + (\rho/(1-\rho)) \cdot P(s, \lambda_i/\mu). \tag{12}$$

Thus, the developed mathematical model, represented by (1–6) and (8–12), allows for balancing the load among controllers and dynamically redistributing traffic in case of failure of any controller that belongs to the set *C*.

Conclusions

The paper presents a mathematical model for the operation of a clustered multicontroller SDN. The proposed approach involves performing three-level hierarchical clustering of the system's control and data transmission devices. The relationships between controllers and OpenFlow switches in the implementation of this clustered structure have been formalized. Calculations of the load distribution on the control elements have been carried out.

The use of the proposed mathematical model makes it possible to balance the load among controllers and quickly redistribute it in the event of a controller failure. The direction of further research is the development of an algorithm for distributing OpenFlow switches among SDN network controllers.

REFERENCE

1. McKeown, N. et al. (2008), "OpenFlow: Enabling Innovation in Campus Networks", *ACM SIGCOMM Computer Comm. Review*, vol. 38, is. 2, pp. 69–74, doi: https://doi.org/10.1145/1355734.1355746
2. Kreutz, D. et al. (2015), "Software-Defined Networking: A Comprehensive Survey", *Proceedings of the IEEE*, vol. 13, is. 1, pp. 14–76, doi: https://doi.org/10.1109/JPROC.2014.2371999
3. *Tootoonchian, A. and Ganjali, Y. (2010), "*HyperFlow: A Distributed Control Plane for OpenFlow," *INM/WREN*, available at: https://www.usenix.org/legacy/events/inmwren10/tech/full_papers/Tootoonchian.pdf
4. Yeganeh, S. H., Tootoonchian, A. and Ganjali, Y. (2013), "On Scalability of Software-Defined Networking", *IEEE Commun. Magazine*, available at: https://www.cs.toronto.edu/~soheil/papers/sdnscalability-ieeemag.pdf
5. Hu, Y., Wendong, W., Gong, X.; Que; X. and Shiduan C. (2012), "Reliability-Aware Controller Placement for Software-Defined Networks" *IEEE Communications Letters*, Gent, available at: https://ieeexplore.ieee.org/document/6573050/authors
6. Suarez-Varela, J. and Barlet-Ros, P. (2017), "Towards a NetFlow Implementation for OpenFlow Software-Defined Networks", *Proc. of the 29th Int. Teletraffic Congress Itc 2017*, 1, pp. 187–195, 8064355, doi: https://doi.org/10.23919/ITC.2017.8064355
7. Shekhawat, V.S., Kulshrestha, R., Yadav, P., Singh, A. and Firdous, F. (2025), "Modeling and performance evaluation of OpenFlow switches using a MAP/PH/1/n queueing model", *Computer Networks*, 266, doi: https://doi.org/10.1016/j.comnet.2025.111338
8. Zhang, J., Huang, X., Li, J., Sun, Q. and Lu, J. (2022), "A Dynamic Flow Table Management Method Based on Real-time Traffic Monitoring", IEEE International Conference on High Performance Switching and Routing Hpsr, 2022-June, pp. 212–217, doi: https://doi.org/10.1109/HPSR54439.2022.9831366

Received (Надійшла) 03.09.2025
Accepted for publication (Прийнята до друку) 05.11.2025

ВІДОМОСТІ ПРО АВТОРІВ/ ABOUT THE AUTHORS

Мигаль Станіслав Вікторович – аспірант кафедри автоматики, електроніки та телекомунікацій, Національний університет «Полтавська політехніка імені Юрія Кондратюка», Полтава, Україна;
Stanislav Myhal – PhD student of the Department of Automation, Electronics and Telecommunications, National University "Yuri Kondratyuk Poltava Polytechnic", Poltava, Ukraine;
e-mail: stas.migal1998@gmail.com; ORCID Author ID: https://orcid.org/0009-0007-4675-7358.

Математична модель системи управління потоками даних кластерної мультиконтролерної SDN

С. В. Мигаль

Анотація. Проблема балансування навантаження між контролерами та узгодження станів мережі є однією з ключових у мультиконтролерних SDN. Для подолання цієї проблеми необхідно розробляти адекватні математичні моделі, які описують процеси управління потоками даних. **Предметом** дослідження є система управління потоками даних в середовищі кластерної мультиконтролерної SDN. **Метою** дослідження є розробка математичної моделі, яка враховує динамічну кластеризацію контролерів SDN при отриманні даних з комутаторів мережі, та дозволяє збалансувати навантаження на контролери та проводити оперативний перерозподіл зв'язків між контролерами та комутаторами.. **Отримані наступні результати**. Розглянутий підхід до проведення трирівневої ієрархічної кластеризації пристроїв управління і прийому/передачі даних системи. Формалізовані зв'язки між контролерами та OpenFlow-комутаторами при реалізації даної кластерної структури. Проведені розрахунки навантаження на елементи управління. **Висновок**. Використання запропонованої математичної моделі дозволяє збалансувати навантаження між контролерами та оперативно перерозподілити навантаження при відмові якогось із контролерів.

Ключові слова: телекомунікаційна мережа, мережа зв'язку, котролер, літаюча мережа, OpenFlow-комутатор, стандарт 5G, стандарт 5G, SDN.