Computer Systems
Science and Engineering

Tech Science Press

DOI: 10.32604/csse.2024.058314

REVIEW

Navigating the Complexities of Controller Placement in SD-WANs: A Multi-Objective Perspective on Current Trends and Future Challenges

Abdulrahman M. Abdulghani[1,*], **Azizol Abdullah**[1], **A. R. Rahiman**[1], **Nor Asilah Wati Abdul Hamid**[1,2], **Bilal Omar Akram**[3,4] and **Hafsa Raissouli**[1]

[1]Department of Communication Technology and Network, Faculty of Computer Science and Information Technology, Universiti Putra Malaysia (UPM), Serdang, 43400, Malaysia

[2]Institute for Mathematical Research, Universiti Putra Malaysia (UPM), Serdang, 43400, Malaysia

[3]Department of Computer and Communication Systems Engineering, Faculty of Engineering, Universiti Putra Malaysia (UPM), Serdang, 43400, Malaysia

[4]Wireless and Photonics Networks Research Centre of Excellence (WiPNET), Faculty of Engineering, Universiti Putra Malaysia (UPM), Serdang, 43400, Malaysia

*Corresponding Author: Abdulrahman M. Abdulghani. Email: abdulrahman.m.abdulghani@gmail.com

Received: 10 September 2024 Accepted: 25 November 2024 Published: 03 January 2025

ABSTRACT

This review article provides a comprehensive analysis of the latest advancements and persistent challenges in Software-Defined Wide Area Networks (SD-WANs), with a particular emphasis on the multi-objective Controller Placement Problem (CPP). As SD-WAN technology continues to gain prominence for its capacity to offer flexible and efficient network management, the task of 36optimally placing controllers—responsible for orchestrating and managing network traffic—remains a critical yet complex challenge. This review delves into recent innovations in multi-objective controller placement strategies, including clustering techniques, heuristic-based approaches, and the integration of machine learning and deep learning models. Each methodology is critically evaluated in terms of its ability to minimize network latency, enhance fault tolerance, and improve overall network performance. Furthermore, this paper discusses the inherent limitations and challenges associated with these techniques, providing a critical evaluation of their current utility and outlining potential avenues for future research. By offering a thorough overview of state-of-the-art approaches to multi-objective controller placement in SD-WANs, this review aims to inform ongoing advancements and highlight emerging research opportunities in this evolving field.

KEYWORDS

SDN; SD-WAN; multi-objectives; controller placement problem (CPP); clustering algorithm; heuristic algorithm; fault tolerance

1 Introduction

The challenge of controller deployment in Software-Defined Wide Area Networks (SD-WAN) involves determining the optimal placement of controllers within the network to ensure efficient and reliable connectivity for end users. The primary objectives in SD-WAN controller placement focus on maximizing network performance and reliability while minimizing operational costs and system complexity. Additionally, one of the key aspects to consider when reviewing the controller placement problem in SD-WAN is the network deployment design. The physical layout of the network, including the location of branch offices and data centers, affects the optimal placement of controllers [1–3]. Controllers installed in data centers can improve network performance by lowering latency and increasing throughput. Another important aspect to consider is traffic patterns. Understanding traffic flows through a network can help determine where controllers should be placed to reduce latency and improve performance [2]. A study by [3] proposed a capacity-aware controller placement algorithm that takes into account both the current and future traffic patterns in the network. Scalability is another important factor to consider when studying the placement of a multi-controller hitch in SD-WAN. As the network grows, the number and placement of controllers must be reevaluated to ensure that they can serve more sites and handle more traffic. A cost-effective controller placement algorithm proposed in [4] considers the cost of deploying and maintaining controllers. Quality of service (QoS) is an important aspect of controller placement in SD-WAN. The controllers should be placed in such a way that they can provide good QoS for the end users. Nonetheless, Reference [5] proposed a scalable controller placement algorithm that considers the network's scalability requirements. When making placement decisions, cost is an important consideration. It is important to keep controller deployment and upkeep costs as low as possible. As described in [5], they proposed a QoS-aware controller placement algorithm that considers the QoS requirements of the network. Cloud integration is another important aspect of controller placement in SD-WAN. Controllers should be placed in a way that they are easily integrated with Cloud infrastructure. In addition, Reference [6] suggested a cloud-aware controller allocation technique that considers the network's needs for integrating with the cloud. In conclusion, controller placement in SD-WAN is a complex problem that involves balancing multiple objectives including network performance, security, scalability, cost, QoS, and cloud integration. The abovementioned have proposed various algorithms and approaches to solve this problem, but further research is needed to address the trade-offs between these objectives and to develop more sophisticated and adaptive controller placement algorithms.

1.1 Software-Defined Wide Area Network (SD-WAN)

SD-WAN is a modern approach to managing Wide Area Networks (WANs) that allows organizations to use software to control and optimize network traffic flow. Fig. 1 shows that the SD-WAN architecture is divided into three planes: control, data, and orchestrations, each of which performs a specific network management function [1]. The data plane is in charge of sending traffic between the various sites, the control plane is in charge of administering the network's protocols and layouts, and the orchestration plane is in charge of keeping an eye on and troubleshooting the network. Furthermore, the SD-WAN architecture consists of several layers, including the control, infrastructure, and application layers, each of which serves a specific purpose in providing network services. Controller placement is an important aspect of SD-WAN, and the multi-controller placement framework seeks to determine the best location for controllers in SD-WAN clusters. To maintain Quality of Experience (QoE), performance metrics such as link load, end-to-end delay, throughput, and latency must also be considered [2]. However, some challenges must be addressed, such as current controller placement issues and future research directions.

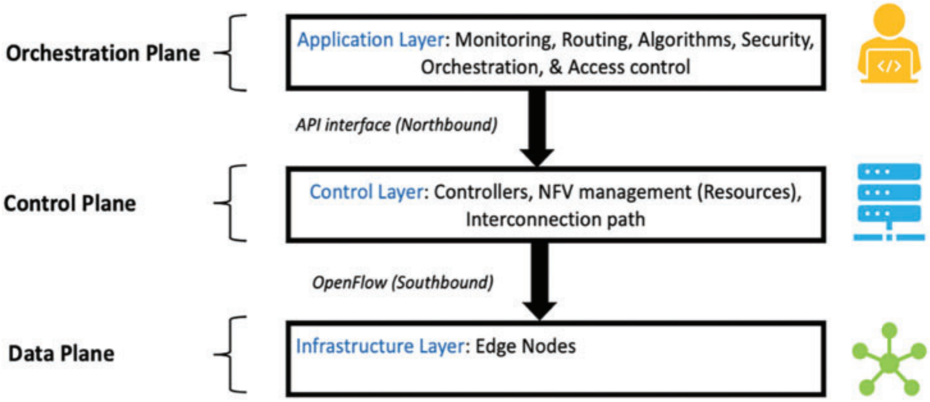

Figure 1: SD-WAN architecture, planes, and layers

Nevertheless, this type of network is rapidly gaining popularity among organizations of all sizes. Fitting to a current market research statement, the worldwide SD-WAN market is expected to reach $13.7 billion by 2027, growing at a Compound Annual Growth Rate (CAGR) of more than 30% during the forecast period (2018–2027) [7] SD-WAN technology is being adopted by many organizations to improve the quality of service and increase the flexibility of their networks. SD-WAN is a networking technology that enables organizations to improve the performance, security, and reliability of their wide area networks. It uses software to abstract the underlying hardware and create a virtualized network that can be easily managed and controlled. By dynamically altering traffic flows following application needs and network circumstances, this enables enterprises to enhance the efficiency of their network infrastructure [8]. One of SD-WAN's main advantages is its ability to reduce wide-area network connectivity costs by using less expensive internet connections while maintaining network security and dependability. SD-WAN allows businesses to improve network performance by dynamically modifying traffic flows in response to application needs and network circumstances [9].

1.2 Controller Placement in SD-WAN

Controller placement refers to the process of determining where controllers should be deployed in the SD-WAN architecture. Controllers are responsible for controlling the network's standards and settings to maintain its dependability and performance [10]. In particular, throughput and latency can be significantly impacted by the positioning of controllers on the network. Several techniques, like the clustering approach [2] and meta-heuristic algorithms [4], may be used to discover the best position for controllers based on elements like network density and traffic patterns to attain appropriate controller placement. Additionally, predictive techniques can also be used to anticipate and prepare for potential controller failures. Proper controller placement can help to ensure that the SD-WAN network performs efficiently and effectively and can also help to minimize the risk of network downtime. Controller placement in SD-WAN is a critical task that involves determining the optimal location of controllers in a network to provide efficient and effective network management. Fig. 2 depicts the three main controller placement methods in SD-WAN: centralized, distributed, and hybrid.

- *Centralized controller placement:* This is a method in which all controllers are in a single central location. This method eases the management and monitoring of the entire network by establishing a single point of control. On the other hand, it may be susceptible to network failures and may not be capable of accommodating high traffic volumes. For example, a game-theoretic approach to centralized controller placement in SD-WAN was proposed in a study conducted by [11].

- *Distributed controller placement:* This is a method for distributing controllers across the network. This method has greater fault tolerance and can handle high traffic loads, but it is more difficult to manage and monitor [12]. For example, a study by [4] proposed a multi-objective optimization technique for locating distributed controllers in SD-WAN.
- *Hybrid controller placement:* This approach integrates the strengths of centralized and distributed methods by strategically positioning controllers in both central and dispersed locations. Ideal for large networks with complex traffic patterns, it offers a balanced solution that optimizes network management. For instance, Reference [13] introduced a machine learning-based technique for hybrid controller placement in SD-WAN, demonstrating the method's adaptability to modern network architectures.

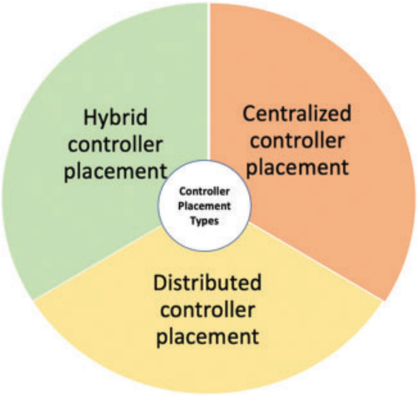

Figure 2: SD-WAN controller placement types

The literature review reveals a prevailing trend toward distributed controller placement strategies in SD-WANs, with the majority of studies favoring this approach across a variety of network topologies. Distributed placement is shown to offer greater scalability and fault tolerance, particularly in larger or more complex network environments. While centralized and hierarchical methods are still applied in specific cases, distributed and dynamic strategies have become increasingly dominant as networks grow in size and traffic complexity.

1.3 SD-WAN Multi-Objective Controller Placement and Key Performance Indicators Evaluation

The placement of controllers in the network to achieve different objectives is a critical component of multi-objective controller placement in SD-WAN. These goals may include decreasing network expenses, raising network dependability, and enhancing network throughput and latency [13]. Innovative techniques like genetic algorithms, particle swarm optimization, and ant colony optimization may be used to balance the trade-offs between many objectives and to determine the best position for controllers depending on numerous objectives [14]. Multi-objective controller placement can be challenging due to the network's dynamic nature and the need to consider multiple objectives at once. However, by utilizing these advanced algorithms, it is possible to strike a better balance between various goals, such as lowering network latency while also lowering network costs [15]. Additionally, multi-objective controller placement can help to improve the overall performance of the network by increasing network reliability and minimizing the risk of network downtime [16]. Moreover, the use of multi-objective controller placement can also help to prepare the network for future requirements, as the network traffic and the network's requirements change over time, it's important to be able to adapt the network to these changes by adjusting the controller placement accordingly. According to Fig. 3, the placement of multi-objective controllers can guarantee that the network is consistently in the optimal condition to satisfy both present and future needs.

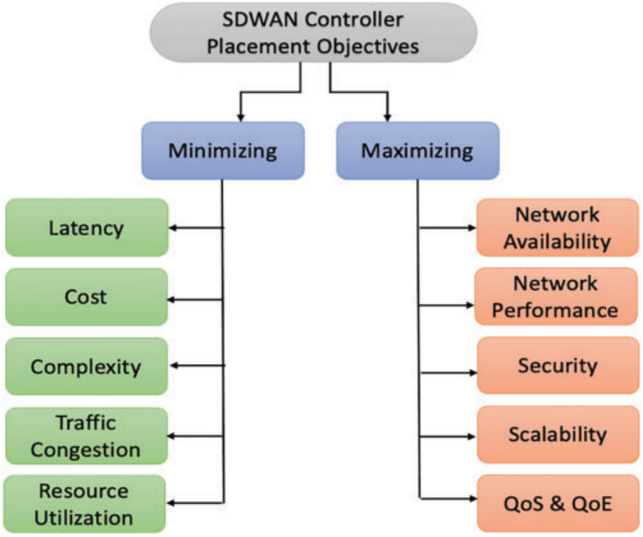

Figure 3: Controller placement objectives for SD-WAN

Overall, the deployment of numerous controllers in SD-WAN networks is essential for optimizing network management across multiple objectives, including increased fault tolerance, reduced operating costs, enhanced performance, and scalability. To achieve these goals effectively, sophisticated algorithms—such as clustering methods, heuristic-based approaches, and machine learning techniques—are employed. These methods provide a balanced approach to competing objectives by ensuring that critical key performance indicators (KPIs) are met.

1.3.1 Importance of Reducing Latency in SD-WAN

Reducing latency in SD-WAN is crucial for ensuring high-quality network performance and meeting the needs of applications that require low-latency communication. Latency, the efficacy of real-time applications like VoIP, teleconferencing, and internet-based gaming can be significantly impacted by packet latency, which is the amount of time it takes a packet to make the journey from its point of origin to its recipient [17]. High latency can cause delays in communication, jitter, and packet loss, resulting in poor quality of service for end users [18]. In addition to affecting application performance, high latency can also impact the user experience. For example, excessive latency in a video conferencing application can result in insufficient video quality and hinder participants' comprehension of the conversation. High latency can result in delays in player actions, which can result in a less responsive and less enjoyable gaming experience in the context of online gaming [14]. The optimization of network resource utilization and the realization of cost savings are both contingent upon the reduction of latency in SD-WAN. Network congestion can lead to wasted bandwidth and increased costs as a consequence of high latency. By reducing latency, organizations can more efficiently use their network resources and achieve cost savings [18]. Reducing latency in SD-WAN is also important for ensuring the reliability and security of the network. High latency can make it difficult for network devices to communicate with each other, which can lead to service disruptions and outages. By reducing latency, organizations can improve the reliability of their networks and minimize the risk of service disruptions [3]. Moreover, reducing latency can also improve the security of SD-WAN by reducing the time it takes for security protocols to be executed. High latency can make it difficult for security protocols to keep up with fast-moving network traffic, which can leave the network vulnerable to attacks [19]. Organizations can improve network security by reducing latency, which ensures that security protocols are executed quickly and effectively. To summarize, minimizing latency in SD-WAN is important for ensuring excellent network performance, meeting the demands of applications with low latency requirements, and making the most efficient use of network resources.

To maintain low latency inside a Software-Defined Wide Area Network (SD-WAN), strategic controller placement is crucial, as seen in Fig. 4. Across numerous nodes, network traffic in SD-WAN is orchestrated and managed by controllers. Ensuring effective communication and lowering delay depend on the proper location of these controllers.

- *Node-Controller Connections:* These connections show the lines of direct communication that exist between each controller on a network and its nodes (switches). The latency of the network can be greatly impacted by the nodes' proximity to controllers. Nodes that are closer to their controllers usually have reduced latency, which improves network performance as a whole.
- *Inter-Controller Connections:* These connections allow controllers in various domains to communicate with one another. Effective inter-controller communication is necessary to keep the SD-WAN synchronized and coordinated, which lowers the risk of latency spikes and guarantees smooth network operations.
- *Multi-Domain Configuration:* The figure above presents a multi-domain configuration in which three controllers (Controller 1, Controller 2, and Controller 3) oversee various network segments. This configuration ensures that controllers are not overloaded and can handle the network load, improving fault tolerance, and optimizing latency.

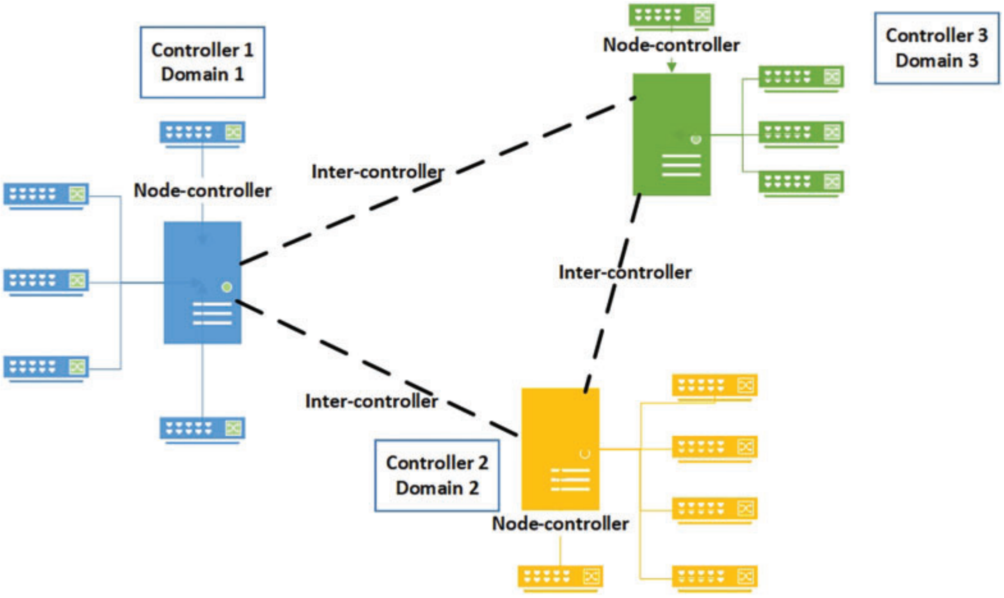

Figure 4: SD-WAN controller placement and latency management

The network can attain a balanced trade-off between dependability, scalability, and performance by optimizing both node-controller and inter-controller connections. This emphasizes how crucial controller location is to building high-performance, low-latency SD-WAN networks. Moreover, the intricacy of the assignment demands the application of advanced algorithms to determine the best placement tactics, guaranteeing the network's ability to effectively accomplish its operational objectives [20].

1.3.2 Cost and Resource Utilization Minimization

Minimizing the costs associated with deploying controllers in SD-WAN environments is a key objective in optimizing network performance and scalability. Several studies have focused on reducing the initial setup costs, operational costs, and resource allocation costs of controller placement, balancing these with other performance metrics like latency and load balancing. Naseri et al. [21] proposed a binary linear programming model that focuses on reducing setup costs by optimizing both the number and type of controllers deployed, while also maintaining low latency between switches and controllers. The model determines the optimal controller placement and types based on their processing capacities and costs, allowing for a balance between cost efficiency and network performance. Fu et al. [22] presented a unique framework for Software-Defined Virtual Private Networks (SD-VPN) for SD-WANs, combining VPN technology with SDN architecture to optimize cost-effectiveness and resource use. The research automates VPN node placement and service orchestration, lowering deployment costs and operating complexity. A combined placement algorithm for VPN controllers and gateways ensures low latency transmission and high-efficiency controllability. Techniques for WAN service loading and offloading maximize network resource use. The study addresses conventional constraints like platform reliance and lack of scalability in VPN-SDN integrations, making SD-VPN a viable option for various cloud, edge, and IoT network applications. Similarly, Reference [23] emphasized the importance of addressing both the Controller Placement Problem (CPP) and Controller Scheduling Problem (CSP) in a single framework. They introduced a Gradient-Descent-based scheduling algorithm combined with a Clustering-based Genetic Algorithm to optimize controller placement while minimizing operational costs. Their approach leverages dynamic traffic allocation to reduce controller workload imbalances, effectively lowering the response time and improving controller utilization without incurring high costs.

To sum it up, minimizing costs while placing SDN controllers is still a crucial goal for maximizing network deployment and operating effectiveness. It is feasible to strike a compromise between lowering setup costs and preserving excellent network performance, including low latency and effective load balancing, by utilizing models like binary linear programming and sophisticated scheduling algorithms. The cost-effectiveness and scalability of SDN installations in wide-area networks are further improved by the combination of heuristic and multi-objective optimization approaches.

1.3.3 Network Complexity Minimization

As networks grow in size and functionality, as is the case with SD-WAN, minimizing the complexity of the control plane becomes increasingly important for maintaining efficient management and scalability [23]. In SD-WANs, where multiple controllers need to communicate and manage a vast array of network devices, inter-controller communication, and control message routing can significantly increase network complexity if not properly optimized [24]. Optimizing controller location to reduce the number of connections between them while maintaining their ability to effectively handle traffic needs is one method of lowering complexity [25]. The network control structure can be simplified by reducing the number of controllers or grouping them in key areas to decrease overall communication overhead. Implementing intelligent traffic distribution methods is another vital tactic. Controllers can perform more effectively and avoid overload by dynamically allocating control messages according to network traffic conditions while maintaining low communication latency. As a result, there is less need for intricate, inflexible scheduling mechanisms, and the network can more readily adjust to shifting traffic patterns [26]. Additionally, simplifying the network architecture through hierarchical or clustered controller arrangements can further reduce complexity [13]. These structures allow for localized decision-making within smaller network segments, reducing the overall load on the central control system and making the network easier to manage as it scales.

In conclusion, network complexity minimization in SD-WANs can be achieved through optimized controller placement, dynamic traffic distribution, and the adoption of simpler, hierarchical control architectures. These approaches help ensure that the network remains scalable, manageable, and efficient, even as it grows in size and complexity.

1.3.4 Traffic Congestion Minimization

In large-scale SD-WAN environments, minimizing traffic congestion is crucial for maintaining optimal network performance and ensuring low latency [3]. Congestion can occur when network controllers become overloaded with requests, leading to delays in processing control messages and increasing response times across the network [19]. Effective strategies to minimize congestion involve intelligent traffic distribution, controller load balancing, and dynamic adjustment mechanisms [27]. One approach to reducing congestion is to balance the load across multiple controllers. By distributing traffic more evenly, no single controller becomes a bottleneck, which helps prevent delays in processing control packets [24,28]. Dynamic load balancing algorithms can adjust the traffic distribution based on real-time network conditions, ensuring that controllers operate efficiently and are not overwhelmed by sudden traffic spikes [27–29]. Moreover, minimizing the distance between switches and controllers can significantly reduce the likelihood of congestion. Placing controllers strategically closer to high-traffic areas ensures that control messages travel shorter distances, reducing both the likelihood of congestion and the overall response time.

Another important method for minimizing congestion is the use of traffic prediction and prioritization mechanisms [30,31]. By predicting potential traffic bottlenecks and prioritizing critical control messages, the network can ensure that high-priority tasks are completed without delay, even during times of high traffic [17]. This proactive approach helps to maintain smooth network operations and prevent the buildup of traffic at any single point. In essence, traffic congestion minimization in SD-WANs requires a combination of load balancing, strategic controller placement, and proactive traffic management techniques. By implementing these strategies effectively, networks can maintain high performance even under heavy loads, ensuring smooth and reliable operations.

1.3.5 Maximizing Network Efficiency and QoS/QoE

In the context of SD-WANs, maximizing availability, performance, security, scalability, as well as Quality of Service (QoS) and Quality of Experience (QoE), is important for ensuring a robust, efficient, and user-satisfactory network. These factors collectively determine the overall reliability and effectiveness of the network.

- *Availability:* Is a key metric in ensuring that the network is operational at all times, particularly in mission-critical environments. High availability can be achieved by employing redundant controllers and failover mechanisms, which allow the network to continue functioning smoothly even in the event of controller or network failures. Ensuring distributed control and backup strategies also contributes to minimizing downtime and maintaining seamless operations [32].

- *Performance:* Closely linked to how efficiently the network processes control and data packets. This can be enhanced by optimizing controller placement, using high-performance controllers, and implementing dynamic traffic balancing mechanisms that adjust to varying traffic loads. Fine-tuning these components ensures minimal latency, high throughput, and fast response times [28].
- *Security:* Security is paramount in SD-WANs, especially as they manage sensitive control plane communications across potentially large and geographically dispersed networks. Implementing strong encryption protocols, access controls, and continuous monitoring of control messages ensures that the control plane is protected from malicious attacks. Security measures must be integrated into the SDN architecture to ensure that both data integrity and confidentiality are maintained [33].
- *Scalability:* The crucial aspect to accommodate the growing needs of modern networks. As networks expand, SD-WAN architectures must be designed to scale without degrading performance or overwhelming controllers. This can be achieved through modular and hierarchical designs that allow for the incremental addition of controllers and network components without disrupting existing operations [22].
- *Quality of Service (QoS):* QoS refers to the network's ability to prioritize different types of traffic and ensure that critical applications receive the necessary bandwidth and low latency. Implementing QoS policies enables the network to dynamically allocate resources based on the needs of various services, ensuring that high-priority traffic—such as real-time applications—is treated with priority [24].
- *Quality of Experience (QoE):* On the other hand, reflects the end user's perception of the network's performance. Maximizing QoE requires not only ensuring high performance and availability but also providing consistency in service delivery. Reducing latency, jitter, and packet loss, particularly in real-time applications like video conferencing and VoIP, is crucial to maintaining high QoE. Proactive monitoring and adjusting the network in real-time based on user feedback are key to ensuring a positive user experience [34].

Ultimately, maximizing availability, performance, security, scalability, QoS, and QoE requires a well-coordinated approach that balances these factors effectively. By deploying strategies that ensure high availability, optimize performance, strengthen security, and provide scalable solutions, SD-WANs can deliver both technical excellence and user satisfaction across diverse and growing network environments.

2 State-of-the-Art Approaches

This section presents an inclusive summary of the previous studies that have been directed in the area of controller placement in SD-WAN. These studies were divided into three main categories: clustering methods for controller placement, heuristic algorithms for controller placement, and fault tolerance approaches for controller over-capacity situations. The following discussion presents a detailed analysis of each of these categories, highlighting the key contributions of each study and how they relate to the proposed research.

2.1 Clustering Methods

Clustering approaches, which combine network devices into sub-domains, have been proposed as a strategy to reduce latency in SD-WAN and the number of communication hops required to reach a target [4]. Clustering occurs when items are grouped so that they are more similar to one another than to objects in other clusters. Clustering is an effective approach in a variety of fields, including data mining, computational science, and networking. Clustering can be used in the SD-WAN framework to group network devices to reduce the number of communication hops required to reach an endpoint, thereby lowering latency [14]. Some of the clustering techniques suggested for use in SD-WAN include K-means clustering, hierarchical clustering, and density clustering. The choice of approach will be based on the network's specific requirements. Each of these methods has advantages and disadvantages of their own.

2.1.1 K-Means Clustering

K-means clustering is one of the most widely used clustering techniques. It is a straightforward, effective strategy that is simple to use. K-means clustering's fundamental premise is to group a set of items into k clusters, where k is an adjustable value [35]. The mean of the items in each cluster is then used to represent the objects in that cluster.

2.1.2 Hierarchical Clustering

Is another well-Known clustering technique. It is a technique that organizes the data into a hierarchy, with each level of the hierarchy denoting a different level of granularity. At the top of the hierarchy, all of the items are in a single cluster, and the cluster sizes decrease as we move down the hierarchy [4]. In some studies, network devices were grouped using hierarchical clustering to reduce the number of communication hops required to get somewhere [35]. Furthermore, by combining network devices to decrease the number of communications hops necessary to reach an endpoint, hierarchical clustering is a technique for lowering latency in SD-WAN. It is a technique that organizes the data into a hierarchy, with each level of the hierarchy denoting various degrees of resolution [36]. At the top of the hierarchy, all of the objects are in a single cluster, and the cluster sizes decrease as we move down the hierarchy.

2.1.3 Density-Based Clustering

Is a technique for separating dispersed objects and grouping those that are densely packed together. When dealing with noise and outliers, as well as finding clusters of any shape, density-based clustering is extremely useful. In various studies, network devices were grouped using density-based clustering to reduce the number of communication hops required to get somewhere [37]. Density-based clustering, however, is a popular technique used in SD-WAN to group comparable data points. It is predicated on the notion that a cluster is an area with a high point density that is divided from other clusters by regions with a lower point density. Discovering clusters of any shape is one of the key benefits of density-based clustering. As a result, it is an excellent fit for SD-WAN solutions where identifying specific network traffic patterns is difficult.

In several studies, the total number of transmission hops required to reach a destination has been reduced by grouping network devices together using K-means clustering [36]. However, K-means clustering, which groups network devices to minimize the number of communications hops necessary to reach a destination, is a well-liked technique for minimizing latency in SD-WAN. To split a set of objects into k clusters, where k is a user-specified value, is the fundamental notion behind K-means clustering. The mean of the items in each cluster is then used to represent the objects in that cluster. In various research, K-means clustering has been used to group network devices to minimize the number of communication hops needed to reach a destination [38]. K-means clustering has the benefit of being an easy-to-implement, straightforward, and efficient approach. However, it also has several drawbacks, including sensitivity to the original cluster centers and data scale sensitivity. However, a lot of research has been done on the use of K-means clustering in SD-WAN, concentrating on enhancing its performance and removing its drawbacks. In [39], the authors of this study suggested placing K self-adaptive SDN controllers for WAN. Their strategy is based on employing a spectral clustering placement method to divide a big network into multiple discrete SDN regions. The authors provide measures for spectrum clustering placement that aim to increase controller reliability and decrease WAN latency. They recommend creating a self-adaptive spectral clustering approach based on the matrix perturbation theory by automatically determining the number of SDN domains using the structure of the eigenvectors. To demonstrate the concepts and methods utilizing the Internet2 OS3E architecture, a test framework with Beacon and bench is utilized. The results of the tests the authors perform under various settings and metrics demonstrate that the self-adaptive placement is successful in resolving the SDN controller positioning problem and accurately determining the number of SDN domains. They do admit that figuring out where to locate SDN controllers generally is still a work in progress and that their strategy is only a first step toward SDN domain division. In further work, they intend to broaden their study to include additional network latencies. Yet, Reference [40] described a controller placement technique that divides the network into partitions and deploys controllers in each one using the K-means approach and collaborative game theory-based initialization. They also suggested two load-aware cooperative K-means techniques to handle partition mismatch concerns. The effectiveness of various techniques was evaluated on Internet2 OS3E and Internet Topology Zoo networks. The findings revealed that our cooperative K-means strategy yielded almost optimum solutions and beat the traditional K-means algorithm for controller placement. The tests also showed that the cooperative K-means strategy's initial variation created balanced partitions when there were fewer divisions, but the second form consistently produced balanced partitions.

According to [35], the controller placement issue in SDN-based WAN systems is a major challenge. They proposed a novel K-means clustering-based method for dividing a large network into several single-controller domains in their research. The primary goal is to distribute the load evenly among all controllers while minimizing the maximum delay between the controller and its associated switches. They evaluate the performance of our method using the Internet2 OS3E topology. The experimental results show that the proposed algorithm produces an acceptable propagation delay, though it is still limited in its ability to account for future reliability. However, their research sheds light on the possibility of employing clustering-based methods to address the issue of controller placement in SDN-based WANs and offers helpful suggestions for further study in this area. In addition, Reference [2] suggested an Agglomerative Hierarchical Clustering (AHC) approach combined with a cluster validity score to address the switch deployment issue in SDN-based networks. The suggested approach efficiently divides the network into several subnetworks and places the SDN-enabled switches in the most advantageous locations, removing the irregularities seen in previous placement methods. Simulation results showed that the AHC outperformed the Optimized K-means algorithm in terms of throughput and latency, demonstrating the utility of the proposed technique. The study of project planning and techno-economic analysis would serve as a future research focus and serve as the foundation for the first steps in the migration to an entirely SDN-enabled South African National Research Network (SANREN) infrastructure.

In [41], the researchers provide an algorithm for SD-WAN's hierarchy architecture's multi-controller deployment. The suggested technique takes into account many variables in the hierarchical control architecture, including request latency, load balancing, and dependability. An updated Louvain method is suggested for network partitioning to enhance latency and load balancing between various divisions. The algorithm also considers availability and dependability while deploying domain controllers and root controllers for SD-WAN in the hierarchical control plane. The suggested technique additionally takes path dependability, degree, and node flow request rates into account. It also balances the number of switches and the size of the flows in each partition in addition to load balancing. The simulation results show that the suggested technique decreases the typical request delay, optimizes load balancing across various partitions, and increases control plane dependability. The suggested algorithm is also reasonably efficient in terms of execution. Their work, however, does not account for deployment costs or control plane latency, both of which should be considered in future research. In [38], the controller placement issue in SD-WANs was addressed by the Simulated Annealing Centroid-based K-Means (SACKM) algorithm, which aims to maximize reliability while adhering to propagation latency limitations. Numerical results show the algorithm's efficacy in terms of computing complexity and network efficiency. It handles the challenges of network partition and controller placement. The method is also appropriate for usage online. The authors also state that future work will concentrate on where to install load-balancing controllers and how to use the algorithm in 5G networks and dynamic circumstances. The location of satellite gateways and SDN controllers in combined spacecraft-terrestrial networks is another area where the authors indicate they are interested in expanding their research.

Moreover, in [42], the authors suggested innovative strategies to lessen the delay in SDN-based networks between controllers and the switches they are connected to. They look into and develop the total delay between switches and controllers in a qualitative manner. To do this, they suggest using a clustering-based network partition algorithm (CNPA) to divide the network into smaller subnetworks and reduce end-to-end latency. To reduce the queuing delay brought on by excessive packet requests from switches, the authors suggest deploying several controllers in each subnetwork because the density of switches in each subnetwork may vary owing to geographic variations. Extensive simulations using two genuine topologies from the Internet Topology Zoo are used to assess the effectiveness of the suggested approach. According to the simulation findings, the CNPA algorithm can successfully reduce the maximum end-to-end latency between controllers.

In [35], the authors described a modified Density Peak (DP) method for multi-controller placement in an SDN-based WAN architecture. This algorithm is assessed and contrasted with well-known ones such as hierarchical K-means, modified-Affinity Propagation (AP), and basic DP. On six distinct SD-WAN topologies from the Internet Topology Zoo, the performance of the Modified-DP method is evaluated using several metrics, including Average Case Latency (ACL), Worst Case Latency (WCL), Inter-Controller Latency (ICL), and Coefficient imbalance Factor (CIF) values. The Criteria Ranking and Inter-Criteria Correlation (CRITIC) approach is used to calculate the weights for each performance parameter, which are then applied to a normalized weighted additive utility function to calculate the total utility value. The evaluation's findings show that Modified-DP is the best method for controlling inequality placement since it produces the lowest utility value across all network topologies. Additionally, this result is supported by the computer controller utilization and fairness measures.

Regarding machine learning-assisted techniques [27], a Stochastic Computational Graph Model with Ensemble Learning (SCGMEL) solution was put out. This solution especially tackles the issues of computational complexity, scalability, and intelligence in controller placement inside SD-WAN. With the use of computational graph models, XGBoost for predictive intelligence, and stochastic gradient descent, their approach optimizes controller placement and achieves improved performance in terms of fault tolerance, scalability, and latency reduction across SD-WAN topologies.

Clustering approaches have been proposed as a method of integrating a controller into SD-WAN to reduce the number of communication hops required to reach a destination. K-means clustering, hierarchical clustering, and density-based clustering are just a few clustering techniques proposed for use in SD-WAN. The network's specific requirements will determine which approach to take. Each of these approaches has advantages and disadvantages of its own. To summarize, density-based clustering is an excellent choice for SD-WAN because it can identify arbitrary-shaped clusters, which are common in network traffic patterns. K-means is a simple and effective computational technique, but it may not always be.

2.2 Heuristic Methods

Recent studies have proposed various controller placement strategies in SD-WANs. According to [43], the controller placement issue in SDN-based networks is critical in addressing the multitude of issues that this sophisticated architecture faces. However, their multi-objective dilemma presents challenges for decision-makers due to the existence of numerous conflicting objectives. Their study addressed the issue of controller location in terms of important metrics including load balancing, controller-to-controller latency, and latency between nodes and controllers. To make a wise choice, one must carefully weigh the trade-offs between these opposing criteria. They used a variety of methods, which have been shown effective at locating a variety of Pareto optimum front solutions for these goals. In their suggested study, the benefits of employing this heuristic strategy over exhaustive search techniques are also explored.

In [36], a controllers' placement strategy was proposed for solving the problem of optimal controller placement in small to medium network instances. The proposed strategy is based on the Non-dominated Sorting Genetic Algorithm III (NSGA III) and can provide optimal solutions for small network instances while providing good solutions in a small number of generations for medium network instances. The proposed strategy can also be easily extended to a wide range of placement problems such as virtual machine placement in data centers, network virtual functions placement, or chains of virtual network functions. Furthermore, the number of objectives can be easily increased, and the optional crossover step can be considered for algorithm improvement and generation reduction. However, this requires a definition and deep analysis of the crossover strategy, which will be considered in future work. However, the limited resource controller deployment problem in SDN-based networks was addressed by a solution approach proposed in [28]. Near-optimal solutions are what the suggested solution method aims to deliver. With the use of several parameter settings and actual WAN topologies, the effectiveness of the suggested approach was examined. This study could help network operators build and manage SDN-based networks that meet a variety of service-level agreements. Otherwise, they fail to investigate the dynamic approaches, which adjust controller-switch allocations depending on the time-varying traffic load of switches, and to expand the suggested method to accommodate controller site or link failures. The suggested approach can also easily handle a variety of objective functions, such as reducing the anticipated control path loss. However, Reference [32] provided two strategies addressing the (CPP) in SDN-based networks that are resource-restricted. These algorithms reduce the search space by locating maximum cliques, resulting in effective and affordable solutions. The efficacy of these proposed solutions was evaluated using a variety of parameter settings and actual WAN topologies. Network operators may build and manage their SDN-based networks to adhere to various service-level agreements with the help of such an analysis. Additionally, it is simple to modify the suggested methods to manage node or link failures or to improve other target functions, including reducing the anticipated control path loss. However, they did not look into the dynamic controller placement, which alters controller-switch assignments based on the switches' fluctuating traffic loads over time.

Using a Nash equilibrium placement procedure, Reference [44] proposed a game theory-based SDN controller placement for SD-WAN that divides bigger networks into more manageable SDN domains. This method seeks to minimize WAN latency and increase controller dependability. The results of tests using the NS-2 emulator show that the technique is successful in resolving SDN controller location difficulties and exhibits the capability to automatically calculate the number of SDN subdomains in a network. The model is assessed using measures for gaming. The authors of [45] discussed the problem of choosing the best location for controllers in an SDN-based WAN design. They suggest two population-based meta-heuristic algorithms called controller placement Particle Swarm Optimization (PSO) and controller placement Fire Fly Algorithm (FFA) to address this significant issue. The major objective is to optimize a set of measures like latency and durability to provide dependable communication between nodes and controllers. The algorithms seek to optimize the performance of the network by determining the optimal number and placement of controllers. The authors also provide a heuristic technique for allocating nodes to controllers. The simulation results show that, in terms of lowering the average latency during single-link failures, the suggested technique performs better than the alternatives. Additionally, the findings demonstrate that while both methods yield respectable outcomes, the controller placement FFA technique delivers the best outcomes with the least amount of computational effort. These are not the only objectives that the research may address; they can also be expanded to include other objective functions such as load balance and energy awareness.

In [28], the paper addressed the controller placement problem (CPP) for wide-area networks (WANs) within Software-Defined Networking (SDN). The focus is on minimizing propagation latency and ensuring load balancing among controllers. This problem involves optimizing the placement of controllers based on network topology, propagation latency, controller availability, capacity, flow request rates, and failure tolerance. The study proposes a robust co-evolutionary algorithm that finds the best controller placement to minimize worst-case latency and maintain balanced loads across controllers, even under varying network conditions and failures. The proposed method's performance is validated using real-world WAN topologies, demonstrating significant improvements in reducing switch-to-controller latency, balancing controller loads, and maintaining network resilience under different scenarios.

Additionally, the particular needs and limitations of the SD-WAN network influence the choice of the controller placement strategy. A graph-based strategy for controller placement in SD-WAN was presented by [27]. This technique models the network using graph theory to determine where the best location for controllers should be by reducing the distance between controllers and endpoints. Reference [46] proposed a method for deploying controllers in SD-WANs using swarm intelligence. This approach takes advantage of the idea of swarm intelligence, in which many agents cooperate to identify the best location for the controller. A mix of ant colony optimization and heuristics approaches was also suggested by [47] to tackle the SD-WAN controller placement problem. To locate the ideal controller location, this approach combines Ant Colony Optimization (ACO) with heuristics like greedy algorithms.

In conclusion, recent studies have explored various strategies to address the controller placement problem (CPP) in SD-WAN networks, recognizing the importance of optimizing placement to enhance network performance and fault tolerance. The key challenge remains balancing conflicting objectives such as minimizing latency, ensuring load balancing, and maintaining network resilience. Approaches such as NSGA-III have demonstrated their efficacy in solving multi-objective placement issues, while other methods like Response Time optimization and game theory-based solutions offer promising improvements in controller reliability and performance under dynamic network conditions. Several algorithms, including swarm intelligence, ant colony optimization, and hybrid heuristic approaches, have also shown potential in finding optimal controller placements across diverse topologies. However, challenges persist, particularly in dynamic environments where fluctuating traffic loads and potential node or link failures need to be accounted for. Moving forward, future research should prioritize the development of dynamic controller placement strategies that can adapt to varying network conditions in real time. Additionally, incorporating user priority flows, enhancing load prediction capabilities, and addressing energy efficiency and cost-effectiveness will further optimize controller placement in SD-WAN. Finally, expanding the applicability of these solutions to broader use cases, such as virtual machine placement and network function virtualization, could significantly benefit emerging SD-WAN architectures.

2.3 Machine Learning and Deep Learning Methods

Due to their success in improving particular network metrics like latency and controller load, traditional clustering and heuristic-based approaches for SD-WAN controller placement have been used extensively. By positioning controllers in the best possible positions with network nodes, for instance, have been used to reduce Switch-to-Controller (S2C) and Controller-to-Controller (C2C) latency. These techniques work well in static environments where traffic patterns are predictable, but they are ineffective in dynamic, large-scale networks where traffic patterns change quickly [48]. The incapacity of these conventional methods to dynamically adjust to shifting network conditions is one of their main drawbacks. Static placement solutions result in unbalanced controller loads and worse delays as network traffic increases or changes, particularly during periods of high traffic. Clustering intends to focus on single-objective optimization, primarily reducing latency or balancing load, without considering multi-dimensional objectives such as energy efficiency or fault tolerance. These methods lack predictive or forecast future traffic patterns, leading to suboptimal controller placement when network conditions change unpredictably [49].

Heuristic methods such as Particle Swarm Optimization (PSO) and Genetic Algorithm (GA) further attempt to reduce network latency and improve reliability by balancing controller loads and ensuring failover mechanisms [25]. However, these approaches often suffer from scalability issues when applied to larger networks with varying traffic demands. Static approaches, while computationally simpler, are nonets where real-time adjustments are necessary to prevent network congestion. Additionally, most heuristic models do not account for future network states, limiting their ability to provide long-term optimization for growing and dynamic networks.

Deep Reinforcement Learning (DRL) approaches further enhance the dynamic capabilities of SD-WANs. Temporal Deep Q-Learning (tDQN) has been specifically designed to manage controller load balancing under fluctuating network traffic conditions [50]. This model continuously learns from network performance and adjusts switch-controller mappings to optimize key performance indicators, including latency, throughput, and Packet Delivery Ratio (PDR). Through its reward-punishment mechanism, tDQN makes real-time decisions that mitigate traffic burst effects by dynamically rerouting traffic to prevent controller overload. This adaptability is essential in modern SD-WANs, where IoT proliferation leads to highly volatile traffic volumes and patterns. An advanced approach involves Multi-Agent Reinforcement Learning (MARL), which distributes control across multiple agents in optimized SD-WAN environments [51]. In this framework, agents make independent decisions while communicating to ensure coordinated, system-wide optimization. MARL frameworks excel in large, geographically dispersed networks where centralized control could create bottlenecks and increase latency. By enabling agents to operate independently yet collaboratively, MARL significantly improves fault tolerance and scalability. It optimizes controller placement by learning from multiple agents' experiences, balancing network load in real time, and addressing the limitations of static, centralized methods [50–52].

Despite the significant advancements offered by Machine Learning (ML), Deep Learning (DL) and Reinforcement Learning (RL) techniques, several challenges persist. Computational complexity and high resource consumption remain critical issues when applying these methods to real-time, large-scale networks [53,54]. Training RL models demands substantial processing power, which can introduce delays in decision-making if network computing resources are insufficient to support rapid learning and adjustment processes. Moreover, while RL-based approaches excel at dynamically adapting to current network states, their effectiveness heavily depends on the quality and quantity of historical training data. The model's predictive accuracy may deteriorate if traffic patterns deviate significantly from those in the training dataset, potentially compromising network performance [53].

In contrast to these limitations, Machine Learning (ML) techniques, Reference [55] proposed Boosting Neural Networks (NNBoost), have shown significant promise in predicting network performance metrics such as Round-Trip Time (RTT), Node to Controller traffic, and Controller to Controller traffic before deployment. These predictive models help optimize controller placement by minimizing errors such as Root Mean Square Error (RMSE) and Mean Absolute Error (MAE), outperforming traditional methods. By leveraging historical data, NNBoost can anticipate traffic congestion and ensure that control locations reduce network latency, balancing both throughput and packet loss. This predictive capability offers a critical advantage over static approaches, which are reactive rather than proactive [48].

Fortunately, these cutting-edge methods are essential for the upcoming SD-WAN management generation due to their advantages, which include multi-objective optimization, predictive analytics, and real-time adaptability. Even in settings with uncertain and quickly shifting traffic patterns, contemporary SD-WAN designs can optimize controller placement, enhance network performance, and guarantee scalability via the use of both machine learning and reinforcement learning approaches. By providing more thorough solutions that take into account both current and future network circumstances, these approaches get beyond the drawbacks of clustering and heuristic approaches. In conclusion, more advanced methods are required due to the limits of traditional clustering and heuristic techniques in terms of scalability, flexibility, and predictive power, even if they offer fundamental answers for SD-WAN controller placement. Robust, dynamic solutions are provided by machine learning and reinforcement learning models, especially when used with methods like Deep Q-learning and Multi-Agent Reinforcement Learning. These advanced methods not only enhance network performance but also future-proof SD-WAN architecture against emerging challenges posed by traffic volatility, large-scale Internet of Things (IoT) integration, and evolving network demands.

2.4 Controller Capacity Management (Fault Tolerance)

Fault tolerance is closely related to the capacitated controller issue, which affects the controller placement procedure in SD-WAN. As control and administration of networks are concentrated in the control plane, defects and failures become increasingly serious. The term "capacitated controller issue" describes the challenge of making sure that the controllers have an adequate number of controllers deployed and sufficient processing power to manage the network traffic. Overwhelming a controller might result in higher latency and subpar network performance [10]. Multiple controllers can help to reduce the capacitated controller problem by providing additional capacity to manage traffic in the event of a controller failure. This reduces the risk of increased delay by ensuring that the network can continue to operate even if a controller fails [54]. When using redundant controllers, keep in mind that reliability and latency may have to be compromised. Latency may increase, for example, if the backup controllers are too far from the primary controllers.

Yet, the control plane is where network administration and control are centralized in the SDNWAN paradigm and each switch in this model has a corresponding controller [23]. Losses in responses and processing at the controller can occur when a controller fails, as can the link that connects the controller and switch. When the link between the switch and the controller is detached, this harms the controlled switches and restricts the operation of the control plane. As a result, packets are discarded, and no new routing directions are sent to the switch [41]. Given these considerations, it is essential to locate the ideal controller location that reduces the required number of devices while maintaining dependability. The fault domains are divided into three different categories [55]. First, there is the forwarding fault domain, which includes problems with the data forwarding plane and elements like switches and cables. The second fault domain is the interface fault domain, which covers issues with both the northbound and southbound interfaces. These interfaces serve as the logical and actual communication channels between the controller(s) and the forwarding devices. The control platform, which is in charge of operations like network state aggregation, communication with northbound apps, and updating the flow tables of the forwarding devices, also falls within the controller fault domain.

Nevertheless, in Reference [29], an effort has been made to enhance the performance of control messages in an SD-WAN environment by introducing the Response Time optimization during the Switch Migration (RTSM) method. This method aims to minimize the dependence of local switches on WAN communications, thereby reducing the influence on end-user Quality of Service (QoS). A device selection technique has been proposed, which is based on selecting the devices that produce the least traffic. Subsequently, another set of OpenFlow (OF) devices that generate the highest load on the controller are identified. The response time of control messages is optimized by using Karush-Kuhn-Tucker (KKT) conditions. The selected devices then migrate to the new controller domain with the least response time. The results obtained through this method demonstrate improved QoS parameters and faster convergence time compared to other existing algorithms. However, the proposed method does not consider user priority flows, which will be a focus of future research, as well as load prediction and traffic priority-based switch selections.

SD-WAN's controller capacity management is a vital component in ensuring the network's performance and dependability. The demand for efficient controller capacity management will increase as more businesses use SD-WAN for their wide-area network [7]. This includes monitoring and controlling the number of devices connected to the controller and managing the bandwidth and resources allocated to each device. Additionally, it is important to have a plan in place for scaling the controller as the number of devices on the network grows. A key advantage of SD-WAN is the ability to optimize network traffic in real-time, which can improve application performance and reduce costs. However, effective controller capacity management is essential to guarantee that the network can handle the increasing quantity of connected plans and maintain optimal performance.

In summary, effective controller capacity management is required to ensure fault tolerance and optimal performance in SD-WAN networks. The capacitated controller issue presents a significant challenge, as controllers must be deployed in sufficient numbers and possess adequate processing power to handle increasing network traffic. Failure to address this issue can result in heightened latency and diminished network performance. Redundant and distributed controllers provide additional capacity and fault tolerance, reducing the risk of network outages in the event of a controller failure. However, the trade-off between latency and reliability, especially when backup controllers are located far from primary controllers, must be carefully managed. To mitigate these challenges, it is recommended to employ advanced algorithms and strategies, and device selection techniques that optimize controller response time and reduce reliance on wide-area communications. Future research should focus on incorporating user priority flows and load prediction into these strategies to further enhance Quality of Service (QoS) and controller performance. Additionally, businesses should develop scalable controller management plans to accommodate the rapid growth of SD-WAN adoption, ensuring that bandwidth, resources, and the number of connected devices can be managed effectively. By optimizing controller placement and capacity management, SD-WAN networks can achieve higher reliability, lower latency, and better fault tolerance in an increasingly complex networking landscape.

3 Analysis and Discussion

3.1 An Overview of the Number of Controllers Used in the Literature

It has been demonstrated that a single controller is insufficient for the handling and management of large networks, and it also serves as a single point of failure, which reduces the network's availability [56]. Ever since the trend began, the deployment of numerous controllers has taken place. The determination of the number of controllers to deploy in a specific network, however, is not a straightforward endeavor [57]. Various factors are considered when determining the number of controllers. First, the size and complexity of the network are important considerations. Larger networks require more controllers to distribute management and improve scalability. Second, if failure is a concern, controller redundancy will be considered to increase resilience against controller failure. Furthermore, a greater number of controllers results in a higher cost. As a result, budget constraints determine the practical number of controllers for any given organization. The literature has covered several network architectures and identified the suggested number of controllers for that architecture. Table 1 recapitulates the literature findings concerning the number of controllers for each network architecture studied. In overall, the number of controllers suggested ranges between 2 and 4 for relatively small network architectures and 10 to 20 for larger architectures.

Table 1: Conclude the number of controllers for each network architecture in literature

Authors	Year	Network topology	No. of controllers
[21]	2023	Colt Forthnet GtsCe GtsHungary HiberniaGlobal KentmanApr Interroute Ion Niif Renater Syringa TataNld Tecove VtlWavenet Tinet Chinanet	3

(Continued)

Table 1 (continued)

Authors	Year	Network topology	No. of controllers
[26]	2021	ATT	5
		Belnet	3
[39]	2020	Polska	2–3
		Spain	2–5
		Cost266	2–3
[44]	2021	AT&T	4
[47]	2022	CloudSim Generated	7
[56]	2024	Internet2 OS3E	3
[58]	2023	Internet2 OS3E	4

In comparing the findings across the literature, it is evident that different studies recommend varying numbers of controllers for similar network topologies, based on distinct optimization goals, algorithms, and network characteristics. For instance, both References [56] and [58] utilized the Internet2 OS3E architecture but arrived at different conclusions regarding the number of controllers needed. Sigh et al. used CCA and PSO algorithms, optimizing solely for latency, and recommended 3 controllers for this architecture. In contrast, Reference [58] employed the Sequential Game Theory Controller Placement (SGTCP) algorithm, which considered not only latency but also load balancing and cost, and thus recommended 4 controllers. This variation underscores how the choice of optimization algorithm and the specific goals of the optimization (e.g., reducing latency *vs*. balancing load and cost) directly affect the number of controllers deployed.

Similarly, the literature on the Savvis network shows a consistent recommendation of 3 controllers, regardless of the algorithm or optimization objective. Both References [56] and [59] suggested 3 controllers for this network topology, though the optimization methods and goals differed between the two studies. This consistency suggests that certain network architectures may have more predictable requirements for controller placement, especially when network complexity and traffic load are relatively stable.

Furthermore, Reference [60] conducted a comprehensive analysis of the TataNld and Planetlab v2 architectures, recommending 10 to 15 controllers due to the increased complexity and traffic volume in these networks. The study emphasized that in larger, more geographically dispersed networks, more controllers are required to handle traffic efficiently, reduce Controller-to-Controller (C2C) latency, and ensure adequate redundancy. Similarly, Reference [61] found that networks such as DFN, Colt, and Cogent require a higher number of controllers (up to 20) to ensure fault tolerance and maintain performance during peak traffic periods, particularly when high fault tolerance and redundancy are prioritized.

When examining the CloudSim generated topology, Reference [45] suggested deploying 7 controllers, a relatively high number for a generated network of this size. This recommendation was made based on a focus on load balancing and cost efficiency, highlighting how simulation environments with synthetic topologies can sometimes require more controllers to achieve balanced load distribution across the network. On the other hand, Reference [62] recommended 3 controllers for Mininet-generated topologies, a number more in line with medium-sized real-world network deployments.

Moreover, Reference [63] focused on randomly generated tree topologies (T1, T3, and T5) and concluded that 5 controllers were optimal for these topologies. Although these findings might seem similar across different tree structures, the controller variation count across other network architectures (e.g., Internet2, TataNld) demonstrates that topology structure, network size, and optimization goals have significant influence on the recommended controller count.

Overall, the literature demonstrates that the number of controllers required for a given network topology varies significantly depending on several factors, including the network architecture, traffic volume, optimization goals, and the specific algorithm used for controller placement. For smaller networks such as Internet2 OS3E, the number of controllers suggested ranges between 3 and 4 depending on whether the primary objective is reducing latency or balancing cost and load. In larger, more complex networks like Planetlab v2 or TataNld, 10 to 15 controllers are often recommended to ensure performance scalability and fault tolerance, with considerations for geographical distribution and synchronization overhead. Studies on larger networks, such as DFN and Colt, suggest up to 20 controllers to maintain high availability and ensure resilience against failures.

3.2 Exploring the Impact of Controller Placement on Latency and QoS

The number of controllers in a network, as well as their placement, have a wide range of consequences for the network. The impact studied in each research work is regarded as the optimization goal. Table 2 summarizes the impact studied in each paper from the literature reviewed.

The consequences are as follows:

- *Latency:* This measurement is considered one of the most important and is highly considered in the literature as an optimization goal. Latency is classified into several types including controller-to-controller latency, switch-to-controller latency, overlay network latency, packet processing latency, and path setup latency. Most of the state-of-the-art focus on three types of latency that are controller-to-controller latency, switch-to-controller latency, and average network latency. In the covered literature, only References [23] and [61] did not consider latency as an impact study. The rest of the papers studied between one to three types of latency [64].
- *Cost:* the cost of controller placement and the number of controllers chosen are very important in CPP. This factor limits, in many scenarios, the number of controllers that can be deployed in a given network. While there are potential benefits to increasing the number of controllers, such as improved redundancy and lower latency, it is important to consider the associated cost implications [40]. Different strategies can be employed to achieve the lowest cost. One such strategy is to minimize the number of controllers [62]. To accomplish this, it is worth identifying suitable locations for each controller. Another approach entails diversifying the types of controllers utilized, with selection based on factors like throughput and cost [56].
- *Load Imbalance:* there is a strong relationship between CPP and load imbalance. In terms of the number of controllers, if the number is too few relative to the network size, this can cause load imbalance due to the reason some controllers might be overwhelmed with the network traffic while other controllers are underutilized [20], and [58]. On the other hand, too many controllers can introduce inefficiencies because of the communication overhead. In terms of the controllers' placement, the placement can be optimized in such a way that the controllers get even loads.
- *Response Times:* The time and throughput are also frequently studied in the context of controller placement. Response time refers to how quickly a controller can process requests and issue commands, while throughput measures the network's ability to handle large volumes of data. Both metrics are influenced by the placement of controllers and the traffic distribution

across the network. Studies such as [62] and [65] emphasize that optimizing controller placement can significantly improve response time and throughput, leading to better overall Quality of Service (QoS). By reducing response times and improving throughput, networks are better equipped to handle high traffic volumes without compromising performance, especially in mission-critical environments.

Table 2: Summarize the Impact of controller placement

Authors	Year	Impact studied
[21]	2023	• Latency – Average switch-controller latency – Average controller-controller latency • Setup cost
[26]	2021	Average computation time
[39]	2020	• Availability • Cost
[44]	2024	• QoS • Programmability percentage
[47]	2022	• Latency – Average switch-controller latency – Propagation latency • Average RTT • Time Session (TS) matrix • Delay • Reliability • Throughput
[56]	2024	• Latency – Average switch-controller latency • Average network latency
[58]	2023	• Latency – Average switch-controller latency – Average network latency • Load imbalance • Cost
[59]	2022	• Latency – Average switch-controller latency – Average network latency • Load imbalance
[60]	2021	• Cost • Average latency

(Continued)

Table 2 (continued)

Authors	Year	Impact studied
[61]	2021	• Response time • Energy • Throughput • Average latency
[62]	2020	• Latency – Average switch-controller latency – Average controller-controller latency • Delay • Cost

Several studies in the literature have explored how these optimization goals—latency, cost, load imbalance, response time, and throughput—interact with each other in the context of controller placement. For instance, Reference [45] focused on reducing propagation latency and improving throughput while also considering reliability and delay as additional performance indicators. Similarly, Reference [65] addressed the interplay between latency, packet loss, and average delay, offering insights into how controller placement affects multiple network performance metrics simultaneously. The literature highlights that the number of controllers deployed in a network and their placement can have far-reaching implications for the overall efficiency and performance of the network. Most studies emphasize the importance of minimizing latency, balancing traffic loads, and controlling costs, while also improving response times and throughput [34,47]. The optimal number of controllers depends on various factors, including the network's size, topology, and traffic patterns. A key challenge is achieving an appropriate balance between reducing latency and cost, ensuring redundancy, and preventing load imbalance, all while maintaining high throughput and minimizing response times [51].

Ultimately, controller placement has a wide range of impacts on network performance, and these impacts are extensively studied in the literature. The focus of each study varies based on the specific optimization goals, which range from latency and cost to load balancing and throughput. By understanding the trade-offs associated with different optimization goals, network designers can make informed decisions about the number and placement of controllers, ultimately improving the efficiency, scalability, and resilience of their networks.

3.3 Dynamic Controller Placement and Cloud-Awareness

Dynamic controller placement refers to the process of dynamically allocating or reallocating controllers within an SDN or SD-WAN network based on changing network conditions. This approach ensures that controller resources are optimally utilized, reducing latency and enhancing overall performance. Several studies in the literature have explored dynamic controller reallocation for various purposes, addressing different optimization goals such as failure recovery, traffic management, and energy efficiency. One of the primary motivations for dynamic controller placement is recovery from failures. As discussed in [54,57,61], networks with multiple controllers have built-in redundancy that allows for failover mechanisms. When a controller fails, the system can dynamically activate a redundant controller and redirect traffic flows to maintain network continuity. This ability to adapt in real-time is crucial for enhancing the fault tolerance of the network, ensuring that services remain uninterrupted during failures. In these cases, the controllers are reallocated based on network health and status monitoring, ensuring that the network continues operating at peak efficiency even in the event of component failures. Another significant driver for dynamic controller reallocation is dynamic flow management, where controllers are allocated based on current traffic patterns and loads. As seen in [60], the ability to dynamically assign controllers depending on the traffic demand minimizes latency by ensuring that controllers are placed optimally relative to the traffic flows. This approach reduces congestion and enhances throughput, particularly in environments with high traffic variability [66]. By continually adjusting the placement of controllers, network operators can balance the traffic load across multiple controllers, ensuring that no single controller is overloaded while others are underutilized. Additionally, energy efficiency is a growing concern in dynamic controller placement, particularly in large-scale networks. As addressed by [45], energy-aware approaches to dynamic controller placement optimize power consumption by reallocating controllers based on the current network demand. During periods of low traffic, certain controllers can be deactivated to conserve energy, while during peak times, additional controllers can be activated to handle the increased traffic load. This method not only reduces operational costs but also aligns with global trends toward more sustainable network operations.

On other hand, cloud-awareness introduces a new layer of optimization to dynamic controller placement, particularly in cloud-integrated SDN and SD-WAN architectures. With the proliferation of cloud services and applications, networks need to be more adaptive and responsive to the demands of cloud-based workloads [67]. Cloud-aware architectures are designed to dynamically adjust controller placement based on the location of cloud resources and user traffic patterns, optimizing both latency and performance in cloud environments [68,69]. Cloud-aware dynamic controller placement is particularly beneficial in hybrid cloud environments, where traffic is distributed across both on-premise and cloud-based infrastructures [64]. By dynamically placing controllers closer to cloud data centers or leveraging cloud-native controllers, propagation delay can be minimized, significantly reducing end-to-end (E2E) latency. This is especially important for latency-sensitive applications such as video conferencing, online gaming, and real-time data analytics, where every millisecond of delay impacts user experience.

Moreover, cloud-awareness enhances the efficiency of resource allocation in dynamic controller placement [41]. As traffic demand fluctuates between cloud and on-premise infrastructures, cloud-aware networks can dynamically allocate bandwidth and controller resources to prioritize cloud workloads, ensuring that critical applications maintain high performance. For example, in times of high cloud traffic, controllers can be dynamically reallocated to manage ingress and egress points to optimize traffic routing and enforce network policies [66,70]. This ensures that the network remains scalable and resilient, even as the volume of cloud-based traffic grows. In cloud environments, controller reallocation can also respond to the changing locations of cloud services, where workloads may migrate between different data centers for reasons such as load balancing, fault tolerance, or regulatory compliance. Cloud-aware dynamic controller placement ensures that controllers are positioned optimally relative to these data centers, minimizing latency and improving Quality of Service (QoS) for cloud users. Furthermore, cloud-native controllers can be scaled dynamically to meet traffic demands in public and hybrid cloud environments, providing greater flexibility and reducing the overhead associated with maintaining a fixed number of controllers.

In summary, dynamic controller placement is a key strategy for optimizing network performance and resilience, with motivations ranging from failure recovery and traffic management to energy efficiency. The advent of cloud awareness further enhances this approach by optimizing controller placement based on the unique demands of cloud-based workloads, reducing latency, improving resource allocation, and ensuring that cloud services are delivered with high performance and reliability. As networks continue to evolve toward more cloud-centric architectures, cloud-aware dynamic controller placement will become increasingly important in ensuring seamless, scalable, and efficient network operations.

3.4 Review the Network Size

The network topologies discussed in the literature vary significantly in terms of size, which is primarily measured by the number of nodes. The largest networks studied contain between 139 and 197 nodes, as observed in the research conducted by [32] and [56], while the smallest networks analyzed include between 9 and 15 nodes. Table 3 summarizes the network topologies, their sizes, and classifications, providing a comprehensive overview of the networks used in SDN controller placement research. Network topologies can be broadly categorized into backbone, customer, and transit segments, each playing a distinct role in network connectivity and data routing. Backbone networks serve as the core infrastructure that interconnects other networks, facilitating high-capacity data transport across long distances. Customer networks, on the other hand, are typically smaller and are used by organizations for accessing services and internal communication. Transit networks are intermediaries, responsible for routing data traffic between different networks. These classifications directly influence decisions regarding SDN controller placement. In backbone networks, where scalability and high data throughput are critical, strategic controller placement at major aggregation points is essential for ensuring efficient management and control. For customer networks, the placement of controllers varies depending on the scale and services required, with controllers often deployed either at the network edge or centrally, depending on the specific needs of the network. In transit networks, controller placement is typically concentrated at ingress and egress points to optimize traffic management and enforce network policies.

Table 3: Summarizes the network topologies used in the literature along with their sizes and types

Authors	Year	Network topology	Number of nodes	Network classification
[56]	2024	Internet2 OS3E	32 nodes	Transit
[58]	2023	Internet2 OS3E	32 nodes	Transit
[59]	2022	Ernet	16 nodes	Backbone, Customer
		Savvis	19 nodes	Backbone, Customer
[60]	2021	Internet2	34 nodes	–
		Planetlab v2	41 nodes	Backbone
		TataNld	145 nodes	Backbone, Customer, Transit
[61]	2021	Mininet-generated	20 nodes	Testbed
[62]	2020	Aarnet	19 nodes	Backbone
		DFN	58 nodes	Backbone, Testbed
		Colt	153 nodes	Backbone, Customer, Transit
		Cogent	197 nodes	Backbone, Customer, Transit
[64]	2020	Random generated T1, T3,	40 nodes	Testbed
		and T5 tree topology	63 nodes	Testbed

(Continued)

Table 3 (continued)

Authors	Year	Network topology	Number of nodes	Network classification
			34 nodes	Testbed
[65]	2021	SDN	20 nodes	–
[66]	2021	SouthAfrica	9 nodes	Backbone

Controller placement decisions are therefore influenced by several factors, including network scale, traffic patterns, and control requirements. For instance, larger backbone networks require more controllers to maintain centralized control and improve fault tolerance, while transit networks benefit from distributed controller placement to efficiently manage cross-network traffic and ensure policy enforcement. As a result, network topology and classification play a critical role in determining the optimal controller placement strategy.

The classification of networks into the backbone, customer, and transit segments has profound implications for controller placement. In backbone networks, strategic placement at core aggregation points is necessary to manage high data throughput efficiently. For customer networks, which are typically smaller, the decision of whether to place controllers at the edge or centrally depends on the specific services and scale of the network. Transit networks require controllers positioned at ingress and egress points to optimize traffic routing and enforce policies. These classifications, along with the size and complexity of the network, dictate how controllers should be deployed to ensure optimal network performance, reduced latency, and effective traffic management. By considering these factors, SDN controller placement can be tailored to meet the specific demands of different network types, ensuring scalability, efficiency, and resilience.

3.5 Insights and Emerging Trends in Multi-Objective CPP for SD-WANs

The CPP in SDN is a fundamental scope of optimizing network performance, particularly in wide-area networks (WANs). The analysis presented in Table 4 provides a comprehensive overview of research studies that tackle multi-objective CPP, shedding light on the various methods, network topologies, and the criteria each study addresses. This table highlights the diversity in approaches and priorities among researchers, revealing areas of strength as well as gaps in the current body of work.

The key observations are as follows:

- *Focus on Controller Failure and Load Balancing:* A key finding from the table is that many studies emphasize controllers' failure recovery and load balancing. These two factors are pivotal in ensuring network resilience and efficient performance. The majority of research has explored methods for dynamically activating redundant controllers in response to failures, thus ensuring uninterrupted network operations. Similarly, load balancing—which distributes network traffic evenly among controllers to prevent overloading—has been addressed in several studies, particularly those focusing on real-world deployments and simulations. This indicates that maintaining reliability and performance is a primary concern for researchers.
- *Dynamic Placement:* Is well-studied but not universal dynamic placement of controllers, which involves reallocating controllers based on real-time traffic patterns or failures, is another heavily researched area. Many studies, such as those utilizing game theory approaches, PSO-based methods, and machine learning models like Q-learning and Deep Reinforcement Learning

(DRL), include dynamic placement to improve overall network adaptability and efficiency. However, a significant portion of research still does not fully address dynamic placement, which suggests that some studies may prioritize static or pre-optimized placement models, potentially limiting adaptability in real-time network environments. Addressing this in future research could further enhance network flexibility, particularly in dynamic and heterogeneous WAN environments.

- *Scalability:* An underexplored area scalability, the ability to maintain performance as the network size or traffic grows, is less frequently addressed compared to other factors. Only a handful of studies have made scalability a core consideration, which is surprising given the growing complexity of WANs and the increasing need to manage larger and more distributed networks. The importance of scalability in global networks is undeniable, and future studies should aim to incorporate scalable solutions that allow SDN architectures to evolve and expand without compromising performance.
- *Limited Focus on Energy Efficiency and Cost:* Despite the potential benefits of improving energy efficiency and reducing deployment costs, these factors remain underexplored in CPP research. Only a few studies, such as those employing Deep Reinforcement Learning (DRL) and Hybrid HSA-PSO, have considered energy efficiency as a critical parameter. Similarly, cost optimization has been addressed in limited scenarios, often in conjunction with other objectives like load balancing. As energy consumption and cost efficiency become growing concerns in modern network architectures, particularly with the rise of green networking and sustainable IT, incorporating these parameters more extensively in CPP research could significantly impact future network designs.
- *Network Partitioning and Security:* rarely addressed Network partitioning, which segments the network to improve fault tolerance and manageability, is addressed in only a few studies. This suggests that while network segmentation could offer more robust solutions for large-scale networks, it is not widely prioritized. Partitioning, particularly in geographically distributed WANs, could help enhance network performance by localizing control and reducing inter-controller communication latency. Moreover, security is almost absent from the focus of these studies, a concerning gap given the increasing threats in SDN architectures. Future research could benefit from integrating security concerns into controller placement strategies, especially with the rise of cyberattacks targeting controller vulnerabilities.
- *Diverse Use of Topologies:* The topologies employed in the reviewed studies vary significantly, ranging from Internet2 OS3E to simulated topologies like those generated in CloudSimSDN or Mininet. This variation in topology use reflects the flexibility of SDN, allowing researchers to explore controller placement across different network structures, from simple testbed simulations to complex, real-world WAN environments. However, this also introduces a challenge for comparing results across studies, as different topologies may lead to varied outcomes based on network size, traffic patterns, and structural complexity. Standardizing the use of certain topologies, such as those from Topology Zoo, could improve the comparability of research findings in future studies.

Table 4: Overview of various multi-objectives research studies on the controller placement problem

Authors	Year	Method used	Network topology	Network partitioning	Controllers' failure	Scalability	Load balancing	Dynamic placement	Cost	Energy	Switch assignment
[21]	2023	Binary linear programming	40+ topologies	No	No	No	Yes	No	Yes	No	Yes
[23]	2021	Clustering-based Genetic Algorithm with Cooperative Clusters	Graph-based CloudSim simulation	No	Yes	Yes	Yes	Yes	Yes	No	Yes
[26]	2021	Matchmaker	ATT Belnet	No	Yes	No	No	Yes	Yes	No	Yes
[39]	2020	Steiner tree and heuristic approach	Polska Spain Cost266	No	Yes	No	No	No	Yes	No	No
[44]	2024	QoS-aware per-flow remapping	AT&T	No	Yes	No	No	Yes	Yes	No	No
[47]	2022	Multi-controller SDN	CloudSim Generated	No	No	Yes	No	No	Yes	No	Yes
[49]	2024	Deep RNN	Synthetic SD-WAN topology	No	No	Yes	Yes	No	Yes	No	Yes
[51]	2024	Temporal Deep Q-Learning	Generic SDN topology	No	Yes	Yes	Yes	No	No	Yes	Yes
[52]	2024	Q-Learning	Distributed SDN topology	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes
[53]	2024	Deep Reinforcement Learning (DRL)	Multi-Access Edge Computing	Yes	No	Yes	Yes	Yes	Yes	Yes	Yes
[55]	2024	NNBoost	Real-world and synthetic network topologies	No	No	Yes	Yes	No	Yes	No	Yes
[56]	2024	CCA and PSO	Internet2 OS3E	No	Yes	No	No	Yes	No	No	Yes
[58]	2023	SGTCP game theory approach	Internet2 OS3E	Yes	No	No	No	No	Yes	No	Yes
[59]	2022	Naked Mole-Rat	Ernet Savvis	No	No	No	Yes	No	Yes	No	Yes
[60]	2021	Quantum particle swarm optimizer	Internet2 Planetlab v2 NataNld	Yes	No	No	No	No	No	No	Yes
[61]	2021	Controller placement model	Mininet-generated	No	Yes	No	No	Yes	No	Yes	Yes
[62]	2020	Garter snake optimization capacitated controller placement problem	Aarnet DFN Colt Cogent	Yes	No	Yes	No	No	Yes	No	No
[64]	2020	Proactive recovery framework for SDN control plane	Random generated T1, T3, and T5 tree topology	No	Yes	No	Yes	Yes	Yes	No	No
[65]	2021	Game theory nash equilibrium	Internet2 OS3E	Yes	No	No	No	No	No	No	Yes
[66]	2021	Node weight deployment policy	South Africa	No	No	No	Yes	Yes	Yes	No	Yes

4 Research Opportunities

The review and analysis conducted in this paper have resulted in the identification of several potential research directions. This includes, but is not limited to:

- *Multi-Objective Algorithms:* The majority of current CPP schemes are aimed at achieving specific goals. Current trends point to a shift toward multi-objective optimization, which can address multiple competing goals at once. Techniques such as evolutionary algorithms and Pareto optimization are being used to investigate the trade-offs between different goals, such as minimizing latency while increasing security. Future research should create comprehensive frameworks that can accommodate multiple objectives and provide decision-makers with insights into the best trade-offs for their specific requirements [3,5].
- *Practical Implementation and Testing:* Moving from simulation studies to practical implementation and testing in real-world scenarios is crucial. Trends include the use of network emulators and testbeds to evaluate the performance of CPP algorithms under realistic conditions. This approach can provide useful information about the scalability, robustness, and reliability of various solutions. Crowdsourcing and participatory sensing should also be investigated as methods of gathering real-world data, allowing for more accurate and representative evaluations of CPP algorithms [20].
- *Time-Sensitive Networks (TSN) in SDN/SD-WAN:* TSN is gaining traction as a critical technology for applications demanding deterministic communication, particularly in fields like industrial automation and automotive systems. By implementing TSN standards, networks can achieve the low latency and high reliability essential for real-time data transmission [30]. This presents a significant opportunity to integrate TSN with software-defined networking (SDN) and software-defined wide area networks (SD-WAN), further enhancing network capabilities. However, this integration poses unique challenges that necessitate further research and development. A key area of focus is the development of efficient CPP algorithms tailored for deterministic data communication across SDN/SD-WAN architectures. This encompasses exploring mechanisms to seamlessly integrate TSN standards with existing SDN controllers, optimizing the placement of TSN-enabled devices within the network, and ensuring the dynamic adaptation of network policies to meet the stringent requirements of time-sensitive applications. Such advancements will pave the way for leveraging the combined potential of TSN and SDN/SD-WAN technologies in supporting the next generation of real-time applications [31].
- *Machine Learning Approaches:* While traditional optimization algorithms have been extensively studied, there is growing interest in ML-based approaches. Trends indicate a shift towards using advanced machine learning techniques like federated learning, which allows models to be trained across decentralized devices without sharing raw data, ensuring privacy and security. Additionally, reinforcement learning is being applied to adaptively manage network resources in real-time, learning optimal strategies from interactions with the network environment. Future research should focus on developing robust ML models that can generalize across diverse network scenarios and handle the dynamic nature of SD-WANs [49].
- *Enhanced CPP Algorithms:* The development of more efficient and effective CPP algorithms is crucial. Recent trends focus on leveraging advanced optimization techniques such as metaheuristics and hybrid algorithms that combine the strengths of multiple approaches. Techniques like deep reinforcement learning (DRL) and quantum computing are also being explored to tackle the complexity of CPP in large-scale SD-WAN environments. Future research on CPP algorithms can significantly improve their efficiency and accuracy by leveraging the computational power of these cutting-edge technologies [50–53].

- *Cost and Energy Efficiency:* There is a growing emphasis on sustainability and cost-effectiveness in network design. Current trends involve the integration of green computing principles into CPP algorithms, aiming to reduce energy consumption and operational costs. This includes the use of renewable energy sources, energy-efficient hardware, and adaptive algorithms that can dynamically adjust the network configuration based on real-time energy availability and cost considerations. Research should also look into economic models that quantify the trade-offs between performance and cost, thereby providing a comprehensive decision-making framework [71].
- *CPP for Diverse Paradigms:* Current research often overlooks other paradigms such as SD-WSN. With the increasing adoption of IoT and edge computing, there is a need to develop CPP algorithms tailored for these environments [63]. Trends include the use of hierarchical and distributed controller architectures to manage the unique characteristics of SD-WSN and SD-IoT networks. This involves addressing challenges such as limited computational resources, intermittent connectivity, and the need for low-latency communication. Research should also consider the integration of SDN with emerging technologies like 5G and beyond, which can provide enhanced connectivity and performance for these paradigms [72].
- *Dynamic and Adaptive Algorithms:* Most existing CPP algorithms focus on static solutions. However, the trend is moving towards developing dynamic and adaptive algorithms that can respond to real-time changes in network conditions. This includes the use of AI-driven approaches that can predict traffic patterns and proactively adjust controller placements. The concept of self-organizing networks (SON) is also gaining popularity, in which the network optimizes its configuration autonomously based on predefined objectives. Research should look into the possibility of using digital twins to dynamically simulate and optimize network performance, providing a real-time feedback loop for continuous improvement [73].
- *Security Considerations:* Integrating security into CPP algorithms is increasingly critical as cyber threats become more sophisticated. Current trends involve the use of blockchain technology to enhance the security and transparency of SDN/SD-WAN operations. Blockchain can provide a decentralized and tamper-proof ledger for logging network events, ensuring the integrity of controller placement decisions. Additionally, research is focusing on developing security-aware CPP algorithms that incorporate threat intelligence and real-time risk assessments, enabling the network to dynamically adapt to emerging threats and vulnerabilities [74].
- *Resource and Request Scheduling:* Effective CPP algorithms should consider resource and request scheduling to ensure controllers can handle network traffic efficiently. Recent trends include the use of AI and ML to optimize resource allocation and predict traffic demand. Techniques like network slicing, which allows the creation of virtual networks tailored to specific applications, are also being explored [75]. This approach can help ensure that resources are allocated based on the unique requirements of each application, improving overall network efficiency and performance. Future research should look into integrating SDN with edge computing to enable more localized and efficient resource management [76].
- *Expanded Performance Metrics:* Future research should include additional performance metrics such as energy consumption, network scalability, and robustness, beyond the commonly considered metrics like throughput, delay, and jitter. Current trends emphasize the need for holistic performance evaluation frameworks that consider a wide range of factors. This includes the use of multi-objective optimization techniques that can balance competing goals, such as maximizing performance while minimizing energy consumption. Research should also explore the development of standardized benchmarks and testing environments to facilitate the comparison of different CPP algorithms [77].

- *Real-World Implementation:* While many CPP strategies are designed for large-scale SD-WAN deployments, they should also be adapted for specific network types, including satellite, optical networks, SD-WSN, SD-IoT [78], and SD-Unmanned Arial Vehicles [79]. Recent trends highlight the importance of field trials and pilot projects to validate the performance of CPP algorithms in real-world scenarios. This includes collaboration with industry partners to deploy and test solutions in operational networks. Research should also focus on developing modular and flexible CPP frameworks that can be easily adapted to different network environments, enabling seamless integration and deployment.

In brief, controller placement and optimization are a critical challenge faced in the realm of the SD-WAN. There is a need for more research to be conducted in this area, particularly in the areas of developing more efficient and cost-aware algorithms, exploring the application of CPP in other paradigms such as SD-WSN, and exploring the potential of ML-based approaches. Additionally, more research should be conducted on multi-objective CPP algorithms and practical implementation and testing algorithms in real-world scenarios. These research directions can help to advance the field of CPP and pave the way for effective and practical deployment of CPP in SD-WAN networks.

5 Conclusion

The optimal placement of controllers in SD-WAN remains critical for ensuring high network performance, reliability, and scalability. While significant progress has been made through various multi-objective optimization strategies, from clustering methods to machine learning techniques, key challenges persist in balancing cost efficiency, energy consumption, and adaptation to dynamic network conditions. Future research must prioritize developing sophisticated algorithms capable of dynamically optimizing controller placement using real-time data and predictive analytics. The integration of DRL and federated learning shows particular promise in enabling self-learning networks that optimize performance without centralized oversight. Additionally, as cyber threats evolve, developing security-aware CPP algorithms with integrated threat intelligence becomes imperative. The scope must also expand to address emerging paradigms like SDN and the IoT, which require unique solutions for low-power operation and massive device connectivity. Furthermore, with the growing emphasis on green networking and sustainable IT, integrating energy-aware algorithms with cost-reduction strategies will be crucial. The field stands at a pivotal point, driven by advanced optimization techniques, machine learning, and real-time data analytics. By addressing these challenges, researchers can develop more resilient, efficient, and scalable SD-WAN solutions that are secure, adaptable, and sustainable for future applications.

Acknowledgement: The authors acknowledge the contribution and support of the Faculty of Computer Science and Information Technology at Universiti Putra Malaysia (UPM).

Funding Statement: The authors received no specific funding for this study.

Author Contributions: The authors confirm contribution to the paper as follows: study conception and design: Abdulrahman M. Abdulghani, Azizol Abdullah; data collection: Abdulrahman M. Abdulghani, A. R. Rahiman, Bilal Omar Akram; analysis and interpretation of results: Abdulrahman M. Abdulghani, Azizol Abdullah, Nor Asilah Wati Abdul Hamid, Bilal Omar Akram, Hafsa Raissouli; draft manuscript preparation: Abdulrahman M. Abdulghani, Hafsa Raissouli. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: All data generated or analyzed during this study are included in this published article.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

[1] A. Jalili and M. Keshtgari, "A new reliable controller placement model for software-defined WANs," *J. AI Data Min.*, vol. 8, no. 2, pp. 269–277, 2020.

[2] B. Isong, R. R. Molose, A. M. Abu-Mahfouz, and N. Dladlu, "Comprehensive review of SDN controller placement strategies," *IEEE Access*, vol. 8, pp. 170070–170092, 2020. doi: 10.1109/ACCESS.2020.3023974.

[3] A. Jalili, V. Ahmadi, M. Keshtgari, and M. Kazemi, "Controller placement in software-defined WAN using multi objective genetic algorithm," in *2015 2nd Int. Conf. Knowl.-Based Eng. Innov. (KBEI)*, IEEE, 2015, pp. 656–662.

[4] A. Chakraborty, S. Misra, and J. Maiti, "Mobility-aware controller orchestration in multi-tier service-oriented architecture for IoT," *IEEE Trans. Veh. Technol.*, vol. 71, no. 2, pp. 1820–1831, 2021. doi: 10.1109/TVT.2021.3126803.

[5] A. Abdi Seyedkolaei and S. A. Hosseini Seno, "Multi-criteria decision-making for controller placement in software-defined wide-area networks," *J. Supercomput.*, vol. 77, no. 11, pp. 13447–13473, 2021. doi: 10.1007/s11227-021-03815-3.

[6] K. Basu, A. Hamdullah, and F. Ball, "Architecture of a cloud-based fault-tolerant control platform for improving the QoS of social multimedia applications on SD-WAN," in *2020 13th Int. Conf. Commun. (COMM)*, IEEE, 2020, pp. 495–500.

[7] Marketsandmarkets, "Marketsandmarkets," 2017. Accessed: Nov. 24, 2024. [Online]. Available: Marketsandmarkets.com:https://www.marketsandmarkets.com/Market-Reports/software-defined-wan-market-53110642.html?gclid=EAIaIQobChMI6JibqvbZ_AIVTQ9yCh2GuwZ-EAAYASAAEgIqw_D_BwE

[8] C. Ujan, M. M. Mohamad, and A. Kasim, "A review of the role of latency in multi-controller placement in software-defined-wide area networks," in *Advances on Intelligent Informatics and Computing*, Cham: Springer, 2021, pp. 435–445.

[9] G. Wang, Y. Zhao, J. Huang, and Y. Wu, "An effective approach to controller placement in software defined wide area networks," *IEEE Trans. Netw. Serv. Manage.*, vol. 15, no. 1, pp. 344–355, 2017. doi: 10.1109/TNSM.2017.2785660.

[10] H. Kuang, Y. Qiu, R. Li, and X. Liu, "A hierarchical K-means algorithm for controller placement in SDN-based WAN architecture," in *2018 10th Int. Conf. Meas. Technol. Mechatron. Autom. (ICMTMA)*, IEEE, 2018, pp. 263–267.

[11] K. Bavani, P. Deepalakshmi, and E. Kalaimannan, "Comprehensive survey of implementing multiple controllers in a Software-Defined Network (SDN)," *Softw.-Defin. Netw. Framew.*, vol. 21, pp. 155–179, 2024.

[12] A. H. Abdi *et al.*, "Security control and data planes of SDN: A comprehensive review of traditional, AI and MTD approaches to security solutions," *IEEE Access*, vol. 12, pp. 69941–69980, 2024. doi: 10.1109/ACCESS.2024.3393548.

[13] J. M. Sanner, Y. Hadjadj-Aoul, M. Ouzzif, and G. Rubino, "An evolutionary controllers' placement algorithm for reliable SDN networks," in *2017 13th Int. Conf. Netw. Serv. Manage. (CNSM)*, IEEE, 2017, pp. 1–6.

[14] J. Liao, H. Sun, J. Wang, Q. Qi, K. Li and T. Li, "Density cluster based approach for controller placement problem in large-scale software defined networkings," *Comput. Netw.*, vol. 112, pp. 24–35, 2017. doi: 10.1016/j.comnet.2016.10.014.

[15] B. P. Killi, E. A. Reddy, and S. V. Rao, "Cooperative game theory based network partitioning for controller placement in SDN," in *2018 10th Int. Conf. Commun. Syst. Netw. (COMSNETS)*, IEEE, 2018, pp. 105–112.

[16] K. Golani, K. Goswami, K. Bhatt, and Y. Park, "Fault tolerant traffic engineering in software-defined WAN," in *2018 IEEE Symp. Comput. Commun. (ISCC)*, IEEE, 2018, pp. 01205–01210.

[17] K. Yang, D. Guo, B. Zhang, and B. Zhao, "Multi-controller placement for load balancing in SDWAN," *IEEE Access*, vol. 7, pp. 167278–167289, 2019. doi: 10.1109/ACCESS.2019.2953723.

[18] A. Sharma, V. Balasubramanian, and J. Kamruzzaman, "A novel dynamic software-defined networking approach to neutralize traffic burst," *Computers*, vol. 12, no. 7, 2023, Art. no. 131. doi: 10.3390/computers12070131.

[19] L. Mamushiane, J. Mwangama, and A. A. Lysko, "Controller placement optimization for Software Defined Wide Area Networks (SDWAN)," *ITU J. Future Evol. Technol.*, vol. 2, no. 1, pp. 45–66, 2021. doi: 10.52953/PUIU5171.

[20] M. Tanha, D. Sajjadi, R. Ruby, and J. Pan, "Capacity-aware and delay-guaranteed resilient controller placement for software-defined WANs," *IEEE Trans. Netw. Serv. Manage.*, vol. 15, no. 3, pp. 991–1005, 2018. doi: 10.1109/TNSM.2018.2829661.

[21] A. Naseri, M. Ahmadi, and L. PourKarimi, "Placement of SDN controllers based on network setup cost and latency of control packets," *Comput. Commun.*, vol. 208, pp. 15–28, 2023. doi: 10.1016/j.comcom.2023.05.015.

[22] C. Fu, B. Wang, H. Liu, and W. Wang, "Software-defined virtual private network for SD-WAN," *Electronics*, vol. 13, 2024, Art. no. 2674. doi: 10.3390/electronics13132674.

[23] V. Huang *et al.*, "A scalable approach to SDN control plane management: High utilization comes with low latency," *IEEE Trans. Netw. Serv. Manage.*, vol. 17, no. 2, pp. 682–696, 2020. doi: 10.1109/TNSM.2020.2973222.

[24] S. Ghosh, M. Iqbal, and T. Dagiuklas, "A centralized hybrid routing model for multicontroller SD-WANs," *Trans. Emerg. Telecommun. Technol.*, vol. 32, no. 6, 2021, Art. no. e4252. doi: 10.1002/ett.4252.

[25] V. Varsha and C. N. Sminesh, "QoS aware multi mapping technology in SD-WAN," in *Intelligent Data Communication Technologies and Internet of Things*. Singapore: Springer, 2020, vol. 57, pp. 421–433.

[26] S. Dou, G. Miao, Z. Guo, C. Yao, W. Wu and Y. Xia, "Matchmaker: Maintaining network programmability for software-defined WANs under multiple controller failures," *Comput. Netw.*, vol. 192, 2021, Art. no. 108045. doi: 10.1016/j.comnet.2021.108045.

[27] K. S. Sahoo, D. Puthal, M. S. Obaidat, A. Sarkar, S. K. Mishra and B. Sahoo, "On the placement of controllers in software-defined-WAN using meta-heuristic approach," *J. Syst. Softw.*, vol. 145, pp. 180–194, 2018. doi: 10.1016/j.jss.2018.05.032.

[28] W. Hassan, T. S. Chou, L. Pagliari, J. Pickard, and O. Tamer, "Is public cloud computing adoption strategically the way to go for all the enterprises?" in *2019 IEEE 5th Intl. Conf. Big Data Secur. Cloud (Big Data Secur.), IEEE Intl. Conf. High Perform. Smart Comput., (HPSC) IEEE Intl. Conf. Intell. Data Secur. (IDS)*, IEEE, 2019, pp. 310–320.

[29] O. Adekoya and A. Aneiba, "A stochastic computational graph with ensemble learning model for solving controller placement problem in software-defined wide area networks," *J. Netw. Comput. Appl.*, vol. 225, 2024, Art. no. 103869. doi: 10.1016/j.jnca.2024.103869.

[30] B. O. Akram, N. K. Noordin, F. Hashim, M. F. Rasid, M. I. Salman and A. M. Abdulghani, "Joint scheduling and routing optimization for deterministic hybrid traffic in time-sensitive networks using constraint programming," *IEEE Access*, vol. 11, pp. 142764–142779, 2023. doi: 10.1109/ACCESS.2023.3343409.

[31] B. O. Akram, N. K. Noordin, F. Hashim, M. F. Rasid, M. I. Salman and A. M. Abdulghani, "Enhancing reliability of time-triggered traffic in joint scheduling and routing optimization within time-sensitive networks," *IEEE Access*, vol. 12, pp. 78379–78396, 2024. doi: 10.1109/ACCESS.2024.3408923.

[32] K. S. Sahoo, M. Tiwary, B. Sahoo, B. K. Mishra, S. RamaSubbaReddy and A. K. Luhach, "RTSM: Response time optimisation during switch migration in software-defined wide area network," *IET Wirel. Sens. Syst.*, vol. 10, no. 3, pp. 105–111, 2020. doi: 10.1049/iet-wss.2019.0125.

[33] G. D'Angelo and F. Palmieri, "A co-evolutionary genetic algorithm for robust and balanced controller placement in software-defined networks," *J. Netw. Comput. Appl.*, vol. 212, 2023, Art. no. 103583. doi: 10.1016/j.jnca.2023.103583.

[34] C. N. Sminesh, E. G. Kanaga, and A. Roy, "Optimal multi-controller placement strategy in SD-WAN using modified density peak clustering," *IET Commun.*, vol. 13, no. 20, pp. 3509–3518, 2019. doi: 10.1049/iet-com.2019.0124.

[35] V. J. Abergos and F. Medjek, "A risk assessment analysis to enhance the security of OT WAN with SD-WAN," *J. Cybersecur. Priv.*, vol. 4, no. 4, pp. 910–937, 2024. doi: 10.3390/jcp4040042.

[36] I. Ellawindy and S. S. Heydari, "Crowdsourcing framework for QoE-aware SD-WAN," *Future Internet*, vol. 13, no. 8, 2021, Art. no. 209. doi: 10.3390/fi13080209.

[37] O. Adekoya and A. Aneiba, "An adapted nondominated sorting genetic algorithm III (NSGA-III) with repair-based operator for solving controller placement problem in software-defined wide area networks," *IEEE Open J. Commun. Soc.*, vol. 3, pp. 888–901, 2022. doi: 10.1109/OJCOMS.2022.3172551.

[38] K. Yang, B. Zhang, D. Guo, M. Lin, and T. de Cola, "Partitioned controller placement in SDWANs for reliability maximization with latency constraints," in *2019 IEEE Globecom Workshops (GC Wkshps)*, IEEE, Dec. 9, 2019, pp. 1–6.

[39] D. Santos, J. P. Vidal, T. Gomes, and L. Martins, "A heuristic method for controller placement and enhanced availability between SDN controllers," in *2020 11th Int. Conf. Netw. Future (NoF)*, IEEE, 2020, pp. 82–90.

[40] N. Cai, Y. Han, Y. Ben, W. An, and Z. Xu, "An effective load balanced controller placement approach in software-defined WANs," in *MILCOM 2019-2019 IEEE Mil. Commun. Conf. (MILCOM)*, IEEE, 2019, pp. 361–366.

[41] T. Shozi, S. Dlamini, P. Mudali, and H. Kobo, "SDN-enabled switch placement in multi-domain hybrid SD-WAN," in *2019 IEEE AFRICON*, IEEE, 2019, pp. 1–7.

[42] P. Xiao, Z. Y. Li, S. Guo, H. Qi, W. Y. Qu and H. S. Yu, "AK self-adaptive SDN controller placement for wide area networks," *Front. Inf. Technol. Electron Eng.*, vol. 17, no. 7, pp. 620–633, 2016. doi: 10.1631/FITEE.1500350.

[43] H. Mojez, A. M. Bidgoli, and H. H. Javadi, "Star capacity-aware latency-based next controller placement problem with considering single controller failure in software-defined wide-area networks," *J. Supercomput.*, vol. 78, no. 11, pp. 13205–13244, 2022. doi: 10.1007/s11227-022-04360-3.

[44] S. Dou, L. Qi, and Z. Guo, "Mitigating the impact of controller failures on QoS robustness for software-defined wide area networks," *Comput. Netw.*, vol. 238, 2024, Art. no. 110096. doi: 10.1016/j.comnet.2023.110096.

[45] X. L. Hou, M. Q. Wu, B. Lv, and Y. F. Liu, "Multi-controller deployment algorithm in hierarchical architecture for SDWAN," *IEEE Access*, vol. 7, pp. 65839–65851, 2019. doi: 10.1109/ACCESS.2019.2917027.

[46] S. Badotra and S. N. Panda, "A survey on software defined wide area network," *Int. J. Appl. Sci. Eng.*, vol. 17, no. 1, pp. 59–73, 2020.

[47] N. S. Radam, S. T. Al-Janabi, and K. S. Jasim, "Multi-controllers placement optimization in SDN by the hybrid HSA-PSO algorithm," *Computer*, vol. 11, no. 7, 2022, Art. no. 111. doi: 10.3390/computers11070111.

[48] S. Yi, H. Wang, X. Yao, C. Gao, and L. Zhai, "Maximizing network utilization for SDN based on WiseAnt colony optimization," in *2016 IEEE 18th Int. Conf. High Perform. Comput. Commun.; IEEE 14th Int. Conf. Smart City; IEEE 2nd Int. Conf. Data Sci. Syst. (HPCC/SmartCity/DSS)*, IEEE, 2016, pp. 959–966.

[49] Z. Nazemi Absardi and R. Javidan, "A predictive SD-WAN traffic management method for IoT networks in multi-datacenters using deep RNN," *IET Commun.*, vol. 18, pp. 1151–1165, 2024. doi: 10.1049/cmu2.12810.

[50] Z. Yang, Y. Cui, B. Li, Y. Liu, and Y. Xu, "Software-defined wide area network (SD-WAN): Architecture, advances and opportunities," in *2019 28th Int. Conf. Comput. Commun. Netw. (ICCCN)*, IEEE, 2019, pp. 1–9.

[51] A. Sharma, V. Balasubramanian, and J. Kamruzzaman, “A temporal deep Q learning for optimal load balancing in software-defined networks,” *Sensors*, vol. 24, no. 4, 2024, Art. no. 1216. doi: 10.3390/s24041216.

[52] A. Kumari, A. Roy, and A. S. Sairam, “Optimizing SDN controller load balancing using online reinforcement learning,” *IEEE Access*, vol. 12, pp. 131591–131604, 2024. doi: 10.1109/ACCESS.2024.3459952.

[53] C. Li *et al.*, “Deep reinforcement learning based controller placement and optimal edge selection in SDN-based multi-access edge computing environments,” *J. Parallel Distrib. Comput.*, vol. 193, 2024, Art. no. 104948. doi: 10.1016/j.jpdc.2024.104948.

[54] D. K. Dake, J. D. Gadze, G. S. Klogo, and H. Nunoo-Mensah, “Multi-agent reinforcement learning framework in SDN-IoT for transient load detection and prevention,” *Technologies*, vol. 9, no. 3, 2021, Art. no. 44. doi: 10.3390/technologies9030044.

[55] W. Jiang, H. Han, M. He, and W. Gu, “ML-based pre-deployment SDN performance prediction with neural network boosting regression,” *Expert Syst. Appl.*, vol. 241, 2024, Art. no. 122774. doi: 10.1016/j.eswa.2023.122774.

[56] G. D. Singh *et al.*, “A novel framework for capacitated SDN controller placement: Balancing latency and reliability with PSO algorithm,” *Alex Eng. J.*, vol. 87, pp. 77–92, 2024. doi: 10.1016/j.aej.2023.12.018.

[57] M. D. Tache, O. Păscuţoiu, and E. Borcoci, “Optimization algorithms in SDN: Routing, load balancing, and delay optimization,” *Appl. Sci.*, vol. 14, 2024, Art. no. 5967. doi: 10.3390/app14145967.

[58] B. R. Killi, R. Tripathi, and V. R. Chintapalli, “Sequential game theory based multi criterion network partitioning for controller placement in software defined wide area networks,” *Comput. Commun.*, vol. 212, pp. 298–312, 2023. doi: 10.1016/j.comcom.2023.10.007.

[59] A. B. Sapkota, B. B. R. Dawadi, and C. S. R. Joshi, “Multi-controller placement optimization using naked mole-rat algorithm over software-defined networking environment,” *J. Comput. Netw. Commun.*, vol. 2022, no. 1, 2022, Art. no. 3145276. doi: 10.1155/2022/3145276.

[60] Q. Zhang *et al.*, “A new quantum particle swarm optimization algorithm for controller placement problem in software-defined networking,” *Comput. Electr. Eng.*, vol. 95, 2021, Art. no. 107456. doi: 10.1016/j.compeleceng.2021.107456.

[61] B. Balakiruthiga and P. Deepalakshmi, “A distributed energy aware controller placement model for software-defined data centre network,” *Iran J. Sci. Technol. Trans. Electr. Eng.*, vol. 45, no. 4, pp. 1083–1101, 2021. doi: 10.1007/s40998-021-00425-w.

[62] S. Torkamani-Azar and M. Jahanshahi, “A new GSO based method for SDN controller placement,” *Comput. Commun.*, vol. 163, pp. 91–108, 2020. doi: 10.1016/j.comcom.2020.09.004.

[63] L. M. Zorello, S. Troia, S. Giannotti, R. Alvizu, S. Bregni and G. Maier, “On the network slicing for enterprise services with hybrid SDN,” in *2020 IEEE Latin-Am. Conf. Commun. (LATINCOM)*, IEEE, 2020, pp. 1–6.

[64] S. Güner, G. Gür, and F. Alagöz, “Proactive controller assignment schemes in SDN for fast recovery,” in *2020 Int. Conf. Inform. Netw. (ICOIN)*, IEEE, 2020, pp. 136–141.

[65] W. R. Varuna and I. R. Vadivel, “Handling controller placement problem using gaming theory in software defined wide area network (SD-WAN),” *Int. J. Adv. Res. Eng. Technol.*, vol. 12, no. 1, pp. 1005–1017, 2021. doi: 10.34218/IJARET.12.1.2021.092.

[66] Y. Dong, B. Li, B. Wu, and M. Feng, “Multi-controller deployment strategies based on node weight and request flow in distributed software defined networks,” in *2021 IEEE 24th Int. Conf. Comput. Support. Coop. Work Design (CSCWD)*, IEEE, 2021, pp. 446–451.

[67] C. Xu, C. Xu, B. Li, S. Li, and T. Li, “Load-aware dynamic controller placement based on deep reinforcement learning in SDN-enabled mobile cloud-edge computing networks,” *Comput. Netw.*, vol. 234, 2023, Art. no. 109900. doi: 10.1016/j.comnet.2023.109900.

[68] S. Troia, L. M. Zorello, and G. Maier, “SD-WAN: How the control of the network can be shifted from core to edge,” in *2021 Int. Conf. Opt. Netw. Design Model. (ONDM)*, IEEE, 2021, pp. 1–3.

[69] A. M. Abdulghani, “Hybrid task scheduling algorithm for makespan optimisation in cloud computing: A performance evaluation,” *J. Artif. Intell.*, vol. 6, no. 1, pp. 241–259, 2024. doi: 10.32604/jai.2024.056259.

[70] J. Wang, M. Bewong, and L. Zheng, "SD-WAN: Hybrid edge cloud network between multi-site SDDC," *Comput. Netw.*, vol. 250, 2024, Art. no. 110509. doi: 10.1016/j.comnet.2024.110509.

[71] Z. Guo, L. Qi, S. Dou, J. Weng, X. Fu and Y. Xia, "Maintaining control resiliency for traffic engineering in SD-WANs," *IEEE/ACM Trans. Netw.*, vol. 32, no. 4, pp. 3485–3498, 2024. doi: 10.1109/TNET.2024.3393841.

[72] M. George and M. K. Roberts, "Design of routing protocols for heterogeneous WSN based on multi-agent reinforcement learning," in *2024 7th Int. Conf. Devices, Circ. Syst. (ICDCS)*, IEEE, 2024, pp. 72–76.

[73] H. Babbar and S. Rani, "PUAL-DBSCP: Personalized ubiquitous adaptive learning for density-based splitting controller placement in software-defined networks," *Comput. Hum. Behav.*, vol. 154, 2024, Art. no. 108135. doi: 10.1016/j.chb.2024.108135.

[74] C. Fan, X. Zhang, Y. Zhao, Y. He, and Y. Yang, "Dynamic relay node selection and routing for cloud-native software defined WANs," *Comput. Netw.*, vol. 241, 2024, Art. no. 110219. doi: 10.1016/j.comnet.2024.110219.

[75] K. Park, S. Sung, H. Kim, and J. I. Jung, "Technology trends and challenges in SDN and service assurance for end-to-end network slicing," *Comput. Netw.*, vol. 234, 2023, Art. no. 109908. doi: 10.1016/j.comnet.2023.109908.

[76] F. Altheide, S. Buttgereit, and M. Rossberg, "Increasing resilience of SD-WAN by distributing the control plane [Extended Version]," *IEEE Trans. Netw. Serv. Manage.*, vol. 21, no. 3, pp. 2569–2581, 2024. doi: 10.1109/TNSM.2024.3386962.

[77] A. Abderrahmane, H. Drid, and A. Behaz, "A survey of controller placement problem in SDN-IoT network," *Int. J. Netw. Distrib. Comput.*, vol. 12, pp. 170–184, 2024. doi: 10.1007/s44227-024-00035-y.

[78] A. M. Abdulghani, M. M. Abdulghani, W. L. Walters, and K. H. Abed, "Cyber-physical system based data mining and processing toward autonomous agricultural systems," in *2022 Int. Conf. Comput. Sci. Comput. Intell. (CSCI)*, IEEE, 2022, pp. 719–723.

[79] A. M. Abdulghani, M. M. Abdulghani, W. L. Walters, and K. H. Abed, "Improving intrusion detection in UAV communication using an LSTM-SMOTE classification method," *J. Cyber Secur.*, vol. 4, no. 4, pp. 287–298, 2022. doi: 10.32604/jcs.2023.042486.