International Journal of Computer Networks and Applications (IJCNA)
DOI: 10.22247/ijcna/2025/32 Volume 12, Issue 4, July – August (2025)

REVIEW ARTICLE

Resource Allocation Load Balancing Secure Schemes in Software Defined Networks: An Analytical Review

Lokesh Pawar
Amity School of Engineering and Technology, Amity University Punjab, Mohali, India.
lokesh.pawar@gmail.com

Rohit Bajaj
Computer Science and Engineering, Chandigarh University, SAS Nagar, Punjab, India.
rohit.rick@gmail.com

Darpan Anand
Department of Computer Science and Engineering, Chitkara University Institute of Engineering and Technology, Chitkara University, Rajpura, Punjab, India.
darpan.anand.agra@gmail.com

Received: 08 May 2025 / Revised: 25 July 2025 / Accepted: 09 August 2025 / Published: 30 August 2025

Abstract – In this day of rapidly advancing technology, small hardware and software designs have become almost natural forces. The Software Defined Networks (SDN) introduced the idea of soft implementations, which it implements as three independent tiers of layers that are referred to as "Data Plane," "Control Plane," and "Application Plane." The load balancing techniques and energy-saving techniques are the main attractions for high performance network improvement. Load Balancing of the network can be achieved at number of stages, and similarly energy efficiency. Managing the allocation of resources, congestion management and using single or multi controller environments the load of the whole network can be shaded. To achieve efficient networking a combination of basic analysis and Quality of Service (QoS) parallel processing will give a comprehensive solution to the readers. The analysis focuses upon i) preventing inefficient resource use, and ii) balancing the load. iii) to secure the network iv) Tool and Technique Selection. This article reviews SDN and Software Defined Edge Controller architectures, emphasizing their roles in addressing network congestion, inefficient resource allocation and energy consumption. It explores load balancing techniques, security issues and SDN's programmability through tools like OpenFlow. The outcome is a novel modular SDEC framework that enables efficient edge computing, dynamic control and reduced carbon emissions. The study demonstrates how SDN, combined with intelligent controllers and local OTH services, offers a scalable low- power and eco-friendly alternative to traditional infrastructures. It is a novel work and provides rich knowledge about effective tools and techniques in SDN for load balancing, energy efficiency and security.

Index Terms – SDN Technology, Software Defined Edge Controller (SDEC), Energy Efficiency, Load Balancing, Security, Review, Analysis, Carbon.

1. INTRODUCTION

With the advent of 5G networks the user demand has exponentially increased for the services from the server. The increasing number of android, window and Mac users is growing day by day. The traditional network infrastructure is not capable enough to provide high processing, ubiquitous access and dynamic management, due to which it immersed itself into congestion, and service request failures. To avoid such type of failures the integration of SDN, Network Function Virtualization (NFV) with edge computing leverages AI-driven orchestration to enhance resource management [1] which intends helps in load balancing. Literature-based evaluation on hybrid cloud-edge-fog designs, cryptographic integrations and threat detection strategies using Machine Learning (ML) techniques reveal that hybrid architectures improve workload distribution [2], however scalability and dataset limitations hinder real-world deployment. At the same time, to mitigate congestion and service request failures several infrastructures-based solutions, architectural and algorithmic solutions were proposed and implemented for better results [3]. Software Defined Networks and Edge computing are newly touted ways for solving the challenges that occurred in the network for giving services to the end users. Software Defined Edge Controller (SDEC) works on the concept of defining the

REVIEW ARTICLE

fundamental definition of the terminal devices and other resources along with virtualizations for the same resources. Figure 1 shows that SDEC has decoupled itself into Software-Defined Edge Devices (SDED), Software-Defined Edge Storage (SDEStr), Software-Defined Edge Computing Resources (SDECR) and Software Defined Edge Service (SDES) where all of these decoupled divisions work in-line with each other in order to cultivate the network for better yield [4]. The network response time is improved using decoupling of the layers for providing support to distinguished services. Figure 1 also shows that the actual internal processing takes place in edge-based service provisioning. Taking SDEC in consideration a few challenges or issues can be mitigated in Edge ecosystems. Load Balancing, energy efficiency, resource allocation and carbon emission are a few challenges which can be completely resolved or partially optimized, depending upon the severity of the problem.

The traditional network suffers from dynamic and automatic management, carbon emission and reconfiguration, also the present network conditions urge for optimization of the network performance such as network device performance and quality assurance for user experiences. The current network deals with the local information and has no exposure to global information of the network. The SDEC supports internet ecology in less carbon emission by providing the services/solutions on the house (OTH). OTH service is a solution provided to the users without the use of internet and solution is locally available. It then reduces carbon emissions by physical infrastructure. In the field of networks congestion caused due to imbalanced traffic, load management and response time from server is major bottlenecks. To give a balanced service to the whole network without any kind of lag in response time and load management a potential approach SDN must be utilized. SDN became a prominent technology for serving network problems; however, there are many tools which can be utilized for implementing the SDN fundamental network, therefore, to identify which terminology is useful to reach an optimized solution. In Figure 2 The impact of edge computing in SDN along with controllers is discussed and represented in the present article.

The Software Defined Network with the help of its unified approach which includes Load Balancers, Routers, Switches, Firewall and Network Address Translation (NAT), which are automatically managed and controlled through programming. Data Plane collects the data and sends it to the Edge Cloudlets as depicted in Figure 2. The data reaches to the edge cloudlets with the help of Base Station. A seamless connectivity is provided to base station and edge cloudlet. The edge cloudlet can be connected to Software Defined Edge Services, where Software Defined Edge Controller manages Software Defined Edge Devices, Software Defined Edge Storage and Software Defined Edge Computing Resources. The next plane encountered is the Control Plane: It is set responsible for Control Synchronization, Control Decision and Resource Management [5].

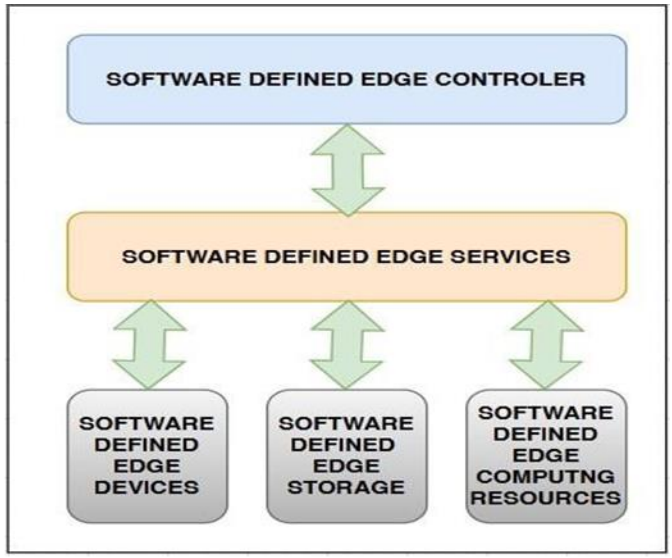

Figure 1 Component View of SDEC

Control Plane is also connected with the Edge Servers and Cloud Servers for managing the services of resource management and service orchestration.

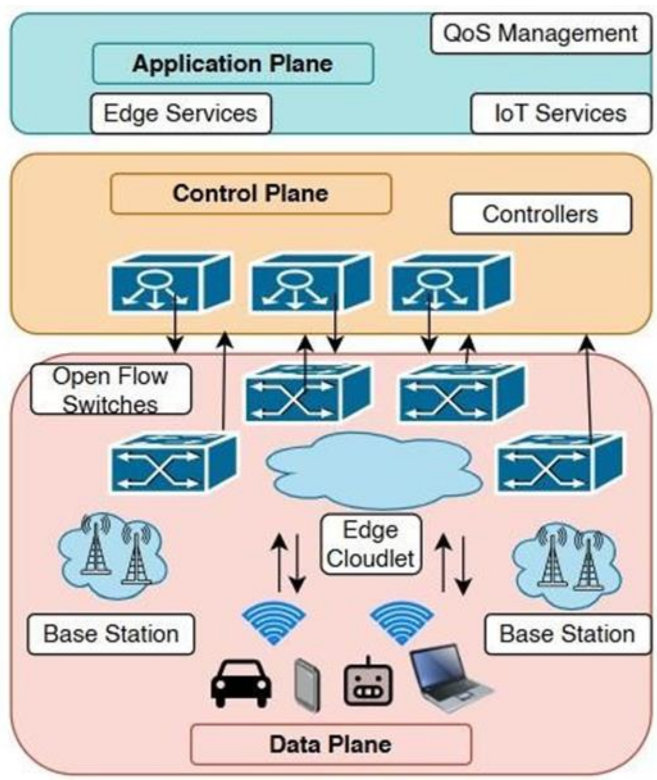

Figure 2 Architecture of SDN

The control plane can be stated as the mind of the whole Software Defined Networks. It is also responsible for defining the routing tables and setting packet handling policies [6]. These controllers are programmable and with soft interventions they can tout good and efficient results [7].

SDN does not require any separate infrastructure for optimizing the network, it can be easily achieved by programming and with its help capacity awareness can be shared with the controller [8]. The network in this scenario is not rigid but placement of the controller plays a vital role in the management of the network [9-11].

The advantages of using SDN rather than traditional routing strategy:

- SDN architecture removes the routing protocols at the hardware level thus it provides an improved network management and diagnosis [12].
- Controller and intelligent load balancer with the help of Open Flow protocol, makes data path control decisions [13].
- Running time can be saved using SDN- Load Balancer.
- SDN can be used for low power IoT devices.[14]
- SDN reduces the CO_2 emissions caused by traditional networking devices.

With rising user demands and 5G deployments, traditional networks face challenges like congestion, energy inefficiency and poor dynamic management. This article is motivated by the need to overcome these limitations using SDN and SDEC. It introduces a modular SDEC framework-comprising SDED, SDEStr, SDECR and SDES for efficient load balancing, reduced carbon emissions and intelligent resource management. The work contributes by integrating edge computing with SDN, detailing control and data plane interactions, highlighting SDN's programmability benefits and presenting key tools and strategies for performance optimization in next generation sustainable networks.

The flow of the article: Section 1 discusses the introduction and applications of the SDN and SDEC. Section 2 represents the load balancing and routing in the field of SDN, further in the current section 2 it represents resource allocation management, delivers information for load balancing in SDEC and discusses the security in defined software networks. The next section 3 discusses the materials and methods. Followed by section 3 the next section 4 discusses the results and section 5 explains conclusion of the article.

2. LOAD-BALANCING AND ROUTING

In recent years, high service provisioning (HSP) and user demand for more efficient and secure services has grown in theworld. With the advent of this many challenges evolved up in Software Defined Edge Computing environment. The user's service requests are increasing day by day with an exponential speed and posing extra burden on the existing cloud and edge infrastructure. The service requests can be handled by a Software Defined Single controller or with the help of Software Defined Multi Controller. The approach of using Single Controller or Multi Controller or distributed controller [15] brings in a lot of challenges which can be traced out as: efficiency, reliability, energy consumption, resource allocation and security. The following literature is reviewed in the field of Software Defined Edge Computing

Resource allocation is a vital part of the network to enhance the performance of the network, not only allocation matters placement also plays an important role [16]. A few articles are considered for review, and they are as follows:

An algorithm for efficient resource slicing which allows first respondents to share their activities was proposed [17]. It focuses on Cellular Networks 3GPP to 5G and B5G. The authors have built a system model which uses thinning function when BSs are not operational due to the impact of catastrophic events. The author has used thinned Poisson Point Process (PPP) for building Voronoi tessellation, where only those BS's (Base Stations) are represented which are working. A Probability Mass Function was utilized to identify the User Equipment (UE) in the Voronoi Cell. Power path loss of a BS-to-UE link was identified in the article.

The probability of success for delivery of a packet was analyzed using signal-to-interference-plus-noise ratio. The SINR expression displays that all the left-over BS's compose a new scenario where the area is made larger to cover all the UE's in the disaster struck city/state/country. The transmission power keeping it at the same rate as SINR can decrease along with quality improvement and latency in the transmission. The probability of transmission's success about a packet can be extracted with the help of SINR equation. If the $SINR_k^{\tau}$ is greater than the threshold value, then the packet is successfully delivered. Using joint optimization algorithm and model users can send messages with a low latency in emergency services. The proposed solution is for disaster situation where in few Base-Stations (BS) are exposed and are no more operational. The parameters on which the results are analysed are time, frequency, reliability, and low latency. Energy efficiency was left behind due to resource slicing and reliability management for first responder. The tool used for experimental setup is NS-3.

Energy efficiency can be taken into consideration and using terminology and algorithms we can devise a system for Wireless Sensor Networks using SDN where energy efficient data transfer can be made available to the UE [18].

Resource Allocation schemes in high performance networks play an important role in network performance. The authors

REVIEW ARTICLE

considered resource allocation in grid computing, multi-core systems, edge computing and fog computing. Also, the authors have compared all the resource allocation schemes on common parameters [19].

A new framework which dynamically balances the resources and allocates them in such a manner which itself increases the accuracy was discussed in [20]. The proposed algorithm is not using a priori approach where task types of parameters and associates' characteristics are considered. The algorithm dynamically searches for sufficient resources which can provide accuracy of desired level. Based on comparison with straw man approach DREAM loosely decoupled the allocation of resources which is further connected with divulgence. The DREAM project follows allocation settlement based on Task Accuracy and Target Accuracy. The straw man approaches can be explicitly said as i) resources can be allocated by using convex optimization and ii) to perform the optimization singularly independent, where in admission control is dependent on Traffic Change at the time of Resource Allocation. CO is convex optimization; OSI stands for Optimization Singularly Independent. Resource allocator based on switches uses the final accuracy to rearrange the resources from high priority task to low priority task. DREAM makes the decisions based on epochs and optimizes the growth and loss of TCAM resources [21]. In the experimental setup the authors have used JAVA and Floodlight controller. The code in java is nearly 20000 lines. A Novel approach of resource allocation where the authors have utilized the advanced version of K-Mean++ algorithm for creating the clusters of the network. The highly required/demanded data is kept on the higher quality clusters where the number of hubs and delays is very low. The authors have increased the number of groups/clusters in the network to increase memory. The proposed scheme has used the memory capacity and balancing of the load in the network with memorization capability. K-means++ algorithm is used for forming cluster of similarity which is measured on delay and hubs. With the increased number of clusters and a significant decrease in the servers the load balancing is reduced hence the quality is degraded in normal conditions. The authors have used two clustering methods k-means++ and spectral clustering for evaluating the proposed method. The average load on the server was calculated. The time complexity of the k-means++ algorithm and spectral clustering was evaluated specific theorems. The authors suggested that in SDN context each challenge is opportunistic and can be considered as a topic of the research [22].

Congestion is a major bottleneck of the network. An analysis on the concept of load balancing using transmission delay between controllers and software defined networks (SDN) switches when network is experiencing heavy traffic was discussed in [23]. The authors proposed an algorithm for load-balancing which is based on Link State prediction. Adaptive Load-Balancing Architecture based on Link-State Prediction (ALBLP) generated a prediction model to identify the state ofthe link and to identify it uses long short-term memory neural network. To achieve optimization, model the authors worked upon the bandwidth of the network. If there is an existence of the link between i and j then the result will be a positive real number and if there is no link establishment then the result will be (-1). If the link exists between nodes i and j then the result will be a positive real number and if there is no link establishment then the result will be (-1). To identify the link utilization ratio of the bandwidth which is used to the bandwidth of the link has been taken. The tool used for experimental setup was Mininet and the algorithms compared with the proposed algorithm are OSPF, RMBQL. The controller used for implementation is Open V-Switch and Ryu. The authors have used GEANT topology at the time of implementation where 23 switches 37 links, 3 different bandwidths: 10 Gbps, 2-5 Gbps and 155 Mbps were used [24].

An algorithm for minimizing the cloud delay and using distributed services rather than using centralized managed services was found to be more promising rather than using different infrastructures. There were two single objective algorithms which are hybridized as joint optimization. By using deep programmability in Software Defined Network the author was able to reconfigure the hosting and routing services. There were two major objective functions which were involved in the present research i) Optimized delay problem ii) Optimized service host problem. The communication amongst vehicles and the deployed RSU's (Roadside Units) was adopted using {IEEE802.11p} wireless communication standard. Poisson distribution in combination with the history of traffic was used for modelling. To address the fundamental objective functions of the present research MILP (Mixed Integer Linear Programming) was used in combination with certain Cloud resource manager and capacity planner tools such as VMware Capacity planner. Ultimately load balancing was achieved using MILP, it supports the network when a single RSU is overburdened it sends the part of the request to the other RSU's with less demands and a load balanced network was achieved. Limitation imposed on the maximum number of hosts on a single RSU allowed better performance. The optimization problem related to delay was converted to (MCF) minimum multi-commodity flow problem. As per the statistics related to the minimization of the service hosts, the problem was scheduled into the NP-hard section, and it was converted to (MWVC) minimum weighted vertex coverage problem to reach an optimized solution [25].

Hampering the stability and robustness of the network congestion is a key factor. The author has given an algorithm for congestion control on IP network and named it as Software Defined Congestion Control Algorithm (SDCC). The algorithm devised works upon TCP and UDP characteristics. There are several other algorithms which have already been

International Journal of Computer Networks and Applications (IJCNA)
DOI: 10.22247/ijcna/2025/32 Volume 12, Issue 4, July – August (2025)

REVIEW ARTICLE

proposed and working on congestion control named Veno, Cubic, Westwood and Hybla, but all these algorithms work upon packet loos judgement. The SDCC algorithm works upon the data flow condition in real network and the author focused upon TCP and UDP data flow. The proposed algorithm first finds out an estimate for congested links and to identify this it uses Link Layer discovery protocol (LLDP) and uses bandwidth as a calculating factor. In the second step the algorithm identifies and reroutes the links and the fundamental base of the algorithm is link and node's distance. Using the link and node's distance it finds the new route for sending the packets. In the third and final step the authors have chosen TCP data flows for rerouting because UDP data flows always demand short transfer time and high efficiency. The authors have also proposed a new method for judging node congestion using the algorithm the author achieved optimized utilization of the rerouted links [26].

Due to semi-automatic management of the network systems there may be challenging performance environments for the 5G systems because of heavy and heterogeneous traffic. The author proposed a novel framework based on cutting-edge technology, like software defined networks and network function virtualizations. The proposed framework is project named as "SELFNET". There are many large-scale projects in this area and none of the projects achieved full integrated and coordinated Software Defined Networks- Network Function Virtualization except "SELFNET". The SELFNET project is scalable and has service integration with cutting-edge technologies like SDN, CC, SON and NFV. The name of the project suggests that it is self-dependent and will organize self-optimization, self- protection and self-healing. The SELFNET six layers architecture topology introduced artificial intelligence, and automatic organizing of the network which decreased the Capital Expenditure (CAPEX) and Operational Expenditure (OPEX) [27].

Semong T. et al. surveyed certain intelligent load balancing strategies. Authors have studied and summarized the importance of providing intelligent load balancing services using the software defined networks (SDN) and Network Function Virtualizations (NFV). The authors have also discussed the tools used by several researchers for implementing and testing strategies and algorithms. Mininet was identified as the best emulator for SDN devices [28]. SDN gives an edge to the researchers and provides them with the flexibility to develop and design their own load balanced and optimized resource allocation strategies. To reach the optimized or intelligent routing decisions network topology is used for achieving optimized results. Load balancing in SDN can be performed on controller and server using the communication path and finding the best device in the whole network web and gear up the performance. The division of network load into several service locations can lead to minimum congestion in the networks [29]. The load balancing in the case of multiple controllers is discussed in [30] and an approach is proposed for managing the adverse network conditions. The scenario was replicated at the time of splitting up of the planes of SDN [31]. Traditional networks uses LSR, OSPF and IS-IS to reach the metrics and network topology details [32]. Due to the approximate synchronization between the routers the transient network traffic loops are formed as and when the metric is updated in LSR or OSPF [33]. The authors in [30] worked on the relative load on several different networking instances where on the contrary others have worked on round-robin [34]. The intelligent load balancing is also discussed at top level of academic achievement in the field of SDN, where genetic algorithms are used for server load optimization. An algorithm in [35] was proposed and it has used 3 form factors Selection, Crossover and Mutation, for the selection of crossover and mutation form factor roulette wheel was used which has given better results as compared to random and round-robin algorithms. Single controllers cannot manage the heavy traffic due to restricted capability. The authors suggested that if multiple controllers are implemented on the control plane it may become easier to achieve high network quality performance and a robust communication system. With these authors proposed a scalable load balancing algorithm which is based on migrating switches [36]. MDCLB, an algorithm on ISSDN-IoT was proposed in the article and on comparing with DLBNB (Dynamic load balancing based on Nash Bargaining), ESMLB (Efficient Switch Migration Load balancing algorithm) and EASM (Efficiency Aware Load Balancing). The problem of scalability and reliability were also addressed by parallel load balancing. Delay in the network causes many issues related to throughput, frequency, reliability and energy which leads to poor QoS. The delay is dependent on the queuing delay. Further, for the implementation of proposed scheme using linear topology was done with the help of Ryu Controller and Mininet, communication between controllers and switches was performed with the help of OpenFlow1.3.

An efficient software defined network (SDN) load balancing scheme based on variance analysis for massive mobile users' authors analyzed and proposed a probabilistic method for managing the load on the network based on variance analysis. A Single SDN Controller for monitoring the data traffic and Open Flow Switching technology is used for implementing the single controller environment. The method proposed in the article provides the services using open flow switching technology for dynamically managing the traffic load. There are two vital strategies i) Probability based selection ii) Variance Analysis. The usage of SDN empowers the services by offering higher flexibility and rapid routing of traffic flow. The F-Test was used to detect the variations within the group and between the group. The results are generated based on dispersion amongst the unbiased sample and the population mean. MS_b and MS_w means between group difference and

REVIEW ARTICLE

within group difference respectively. df_b and df_w means degree of freedom between group and degree of freedom within group respectively. SS_t ,SS_b and SS_w are named as total sum of squares, between group of sums of squares and within group sum of squares respectively. When SS_b is divided with the degree of freedom between the groups and capable of finding out the numeric results to MS_b. Multiple T-test were used to identify which group has more significant level of dispersion and the authors have used null hypothesis approach in implementing and finding the results. The authors have used Mininet 2.1.0 in Ubuntu 14.04.3-desktop-amd64 with Floodlight Controller version 1.0 and Open vSwitch 2.3.2 was used to simulate the environment. The results obtained have low cost, increased reliability and it is scalable [34]. An algorithm for controlling adaptive load balancing (GCALB) for load balancing in SDN was proposed in [35]. The algorithm proposed by author when compared with the present reference algorithm finds the (GCALB) more efficient based on throughput and response time. The tool used for the experimental setup was mininet and floodlight. The properties analyzed and taken into consideration are throughput and response time [37]. Multiple Controller environments give effective services to the high-performance networks. An algorithm using multiple distributed controllers was proposed and analyzed with existing algorithms for load balancing. The parameter for comparing the performance was considered as CPU utilization. The tool used is mininet for experimental setup and the controller used is Ryu. The algorithms compared are Distributed Hoping (DHA), Switch Migration Based Decision Making (SMDM) and Online Controller load balancing (OCLB) [38]. The authors have identified the latency as a parameter for comparison. The architecture utilized of the controller is centralized, and the controller used is Ryu controller, authors have utilized 3 controllers, (OVS) open virtual switches were used for experimentation and the protocol used is openFlow1.3. The load testing tool was iperf and tests were conducted on the traffic generated from TCP/UDP [36]. The disbursal of the load was calculated by taking ratio of complete load of the controllers and average load of the controllers. An automatic load-balancing architecture which is based on reinforcement learning gives a different direction for managing the load of the network. The proposed design balances the traffic load at peak times and uses deep deterministic policy gradient to find the optimal path. The authors have used mathematical model and reinforcement learning for load balancing. The tools used for experimental setup are Mininet simulations and the controller used is Ryu controller and Open Flow. Pytorch is used for implementing reinforcement learning. The authors have compared OSPF, RSIR, EARS, ALBRL where the ALBRL is the proposed algorithm and it outperformed the other algorithm based on better performance, stability and faster convergence rate [39]. It is a cumbersome task to manage and maintain huge number of network traffic and devices, decision making becomes slow and complicated. So, the ALBRL alone could not reach the optimized solution, and then the authors utilized Deep Deterministic Policy Gradient (DDPG). The DDPG model used Sum-Tree binary structure for optimizing the method. The contribution of the article is the improved algorithm for performing well in the heavy load time on the network and another contribution is a mathematical model which was used to evaluate the network load in heavy traffic situations. The load balancing was achieved with a mathematical model. The authors were inspired by this necessity of software defined networks SDN that the network must be disclosed with the vulnerabilities in the heterogeneous SDN operational scenarios. The authors proposed a framework which uses fuzzing modules to discover unidentified or unknown security issues in SDN. The authors introduced a new framework DELTA which is compatible with Open Day Light Controller and other SDN Controllers. The authors have used fuzzing models to reach the reliability of the proposed framework. Energy efficiency, load balancing and latency was not considerably focused upon in the article, although on the ground of security this is the first article which elaborates the security concerns with all the prominent attacks threatening the security of all the controllers in SDN [40]. Security is mandatorily required in any network and architecture which provides end to end security to multiple SDN domains is the requirement of the networking environment. The authors have used a language-based approach which provides security to the SDN services. Authors have also discussed dynamic security which is again policy based and providing security at service layer which makes the approach more robust in providing protection to the available network devices. The proposed approach dynamically secured multiple devices at the same time, with various types of attacks. The security policies are prepared at granular level and take every untouched corner into consideration [41]. The authors consider that many measures are developed to counter the threat in SDN, but the effectiveness must be evaluated on certain parameters. The authors also believe that it will become difficult to secure SDN without using a systematic approach. The author proposed a new graphical security model which is a systematic approach for evaluating attacks, threats and counter measures [42]. A scheme for establishing security in SDN was proposed in [43]. The authors in this research paper have presented a system model for effective usage of SDN and Internet of Things (IoT) and drawn attention on mitigating the masquerading attacks in SDN-IoT environment [44]. A survey on security threats and challenges in Software Defined Networks (SDN) was discussed in [45]. The SDN increases the visibility of the network. Security challenges can be easily mitigated by securing each plane separately. The author has focused on future and present challenges in security [46]. Security improvement and DDoS detection in SDN-enabled IoT environments necessitates machine learning-based approach

REVIEW ARTICLE

using RF algorithm on a synthetic dataset. The methodology of implementation involves benchmarking against recent hybrid and optimization-based models, this methodology followed by the results achieves superior accuracy 99.99% which outperformed related works in DDoS detection effectiveness [47]. The present requirement is to construct an attack taxonomy and analyze existing attacks behavior and security solutions [48].

Table 1 Summarization of Different Techniques/Tools/Terminologies in SDN

Author	Technique Used	EE	LB	NQP	LL	ST	SC	SE	RB	HG	IoT	EC
B Kazi et. al. [1]	Systematic Literature Review	✓	✓	×	×	×	×	✓	×	×	×	×
Mishra S R et. al. [2]	Literature Review	✓	✓	×	✓	×	×	✓	×	✓	✓	×
A.Gebremariam et. al.[17]	Stochastic Geometric Tool	×	×	✓	✓	✓	✓	✓	✓	✓	✓	✓
L Hongming [18]	Deep Programmability Globally Distributed Manner	✓	×	×	✓	✓	✓	×	×	✓	✓	✓
Y Hu [19]	Centralized Control	×	✓	✓	✓	✓	×	×	✓	✓	✓	✓
N. Pedro [21]	Virtualized Network (SELFNET)	×	×	✓	✓	✓	✓	✓	×	✓	✓	✓
H. Babbar et. al. [35]	Multi Controllers	×	✓	✓	✓	×	✓	×	✓	✓	✓	✓
Zhong et. al. [36]	Switching Technology	×	✓	✓	✓	×	✓	×	×	✓	✓	✓
J. Chen et. al. [39]	Reinforcement Learning	×	✓	✓	✓	✓	×	×	×	✓	✓	✓
L Seungsoo [41]	SDN Penetration Tool	×	×	✓	×	✓	✓	✓	✓	✓	✓	✓
Varadharajan V. [42]	Policy Based Attack Detection (PbAD)	×	×	✓	×	✓	✓	✓	✓	✓	✓	✓
Ali J. et. al. [45]	Clustering (Multi Criteria Decision Making)	×	×	✓	✓	×	✓	×	×	×	✓	✓
Karmous N et. al. [47]	DDoS	✓	×	×	✓	×	×	✓	×	×	✓	✓
Nsaif M. et. al. [48]	Integer Programming Model	✓	✓	✓	×	✓	×	×	✓	×	✓	×
Simoes R. et. al. [49]	Multi Access Edge Computing	×	✓	×	×	×	✓	×	×	×	✓	✓
Simoes R. et. al. [49]	Network Function Virtualization	✓	✓	✓	✓	×	✓	×	×	✓	✓	✓

REVIEW ARTICLE

3. MATERIALS AND METHODS

The materials used for statistical analysis are procured from reputed journals[49], the techniques terminologies and methods used in the articles published were considered as a base for finding out the suitable performance cluster. The cluster formed will give birth to a robust approach for attaining optimized solutions [50]. The method utilized is parallel implementation of similar type terminologies to attain optimized Load Balancing and security in Software Defined Networks. The tools were categorized according to the number of times it is used by the authors in different research aspects. The terminologies identified through literature are based on the number of times the terminologies are utilized in a similar kind of environment and on same or different test beds. While analyzing the tools we find that Mininet became the most utilized tool for testing the virtual scenarios for SDN. There are several topologies which can be utilized for implementing algorithms; according to the literature, topology has provided better results [51][52]. The potential work done in the field of network function virtualization in SDN can be extracted from a considerable number of articles out of 80 articles considered for identifying the thrust area of research in the field of SDN. The potential articles are 15 in number, where a good amount of result-based content can be read by the researchers. However out of 15 articles only 13 articles discuss all the parameters considered for qualitative performance and make 86% of the total articles considered for analysis. However, it leads a path for identifying the prominent tool and technique which can be followed to attain a better result according to the recent trends.

4. RESULTS AND DISCUSSION

The article considered for the review of the current research in the field of Software Defined Networks, pilot an established and collaborative analysis of techniques utilized in SDN and tools for implementing the techniques. Table 1 represents a summary of the techniques and terminologies. The schema of the table considers Author, Technique name, and then 11 parallel Quality of Service parameters which can be utilized for extracting out a better result. Few articles were considered for the comparison and collaborative analysis where a utilization of effective techniques and terminologies was reaped out at the time of review and a comparison table was formed. Stochastic Geometric Tool is the technique used for finding efficient solutions, but it lacks energy saving and maintaining the load balancing of the network. We find that Network Function Virtualization is capable enough to provide energy efficiency and load balancing of the network. Stochastic Geometric Tool outperformed all other techniques, and the worst technique is Multi Access Edge Computing. A total number of 80 articles considered for the review process. However, the articles considered for comparison are 13 out of 80 where 7 articles claim the potential work done for load balancing. Similarly for edge computing 12 articles prominently discussed the potential empowerment of edge computing in improving the network performance out of the 13 articles selected from a cluster of 80 articles. The reinforcement learning technique covered 7 performance evaluations parameters out of 11 parameters. There are techniques covering almost all parameters using a single technique. The process of selecting such a technique is rigorous and cumbersome[53]. Similarly other techniques and tools can be read and considered for evaluation and analysis about the techniques covering number of parameters. Under the label of parallel techniques usage, a graphical representation displays a clear picture of techniques utilization. This suggests that on a parallel basis energy efficiency is not considerably outgrown in the field of Software Defined Networks and Edge Computing. Edge Computing was individually focused by 92 % of the total articles considered for the rigorous analysis and overall considered by 15% of the articles, which suggests that there is a remarkable difference between the parallel techniques used to achieve optimization and efficiency in the field of Software Defined Networks and Software Defined Edge Computing. Similarly, Network Quality Performance (NQP), Low Latency (LL) , Scalability (SC), Heterogeneity (HG) , Load Balancing (LB) are individually focused QoS Parameters with 84%, 75%, 75%, 75% and 58% respectively and as displayed in Figure 3 their contribution in the overall judgement of usage is 12%, 11%, 11%, 11% and 8% respectively. The information generated leads to finding out a thrust area to work upon in the future interest of Software Defined Networks and Software Defined Edge Computing.

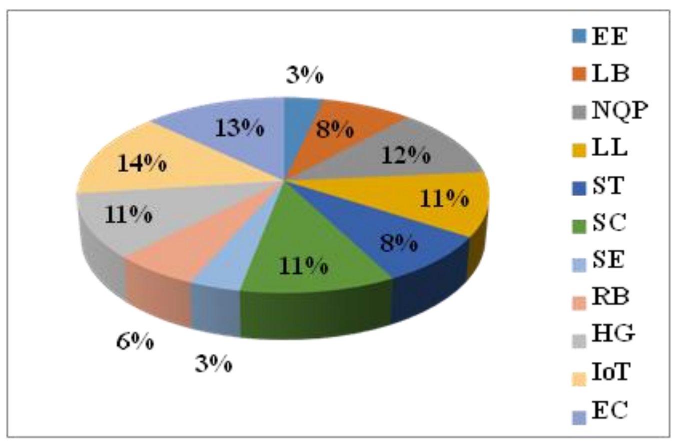

Figure 3 Favorable Relationship Representation Amongst Techniques

The difference found in parallel usage is below average and a good amount of work can be done in parallel optimization of load, energy and scalability. A lot of work is done on these QoS parameters individually and has given an optimized result at the end, but there is a lack in parallel utilization and

International Journal of Computer Networks and Applications (IJCNA)
DOI: 10.22247/ijcna/2025/32 Volume 12, Issue 4, July – August (2025)

optimization. Not every article considers energy and load balancing at equal proportion [54], or a blend of QoS, which may lead to better results. However, this article can be considered as a maiden article to lead us to reach the thrust areas in the fields of SDN and SDEC.

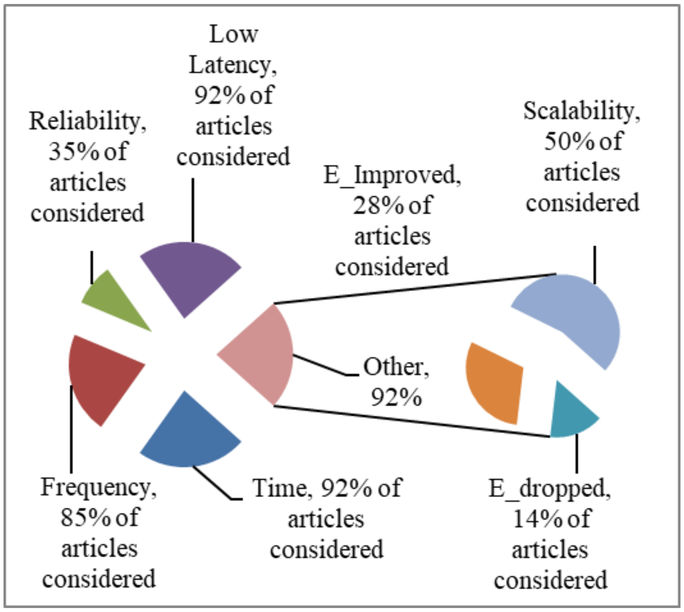

Figure 4 Quality of Service

The summary of additional articles based on QoS parameter metrics can be analyzed and read from Table 2. The schema considers Server Response Time, Packet Frequency, Packet Reliability, Network improved it can be identified from Table 2. The graph also shows that there can be a strong prediction that when energy improves in the system, network latency, packet frequency and response time all of these quality-of- service parameters improve. Figure 5 represents the environmental impact and states that the carbon emission footprint reduces when we use software defined networks. Figure 6 suggests that tool NS-3 is minimally utilized in SDN environment, similarly GEANT, BT North America and CONUS topology are minimally utilized. Mininet simulators and Iperf protocol are maximally utilized in the field of SDN for testing and experimenting purposes.

Latency, Energy Efficiency [55] [56] and Scalability as primary parameters for identifying the improvement in the performance using several select techniques along with the above-mentioned service parameters, the fundamental element for testing and experiment is also considered in the schema along with the algorithm, approach and the utilized method. Using the analysis from Table 2 a graphical representation of the whole table is represented in Figure 4. The graph displays the most focused quality of service parameters equal or parallel to other parameters. Improvement in Response time from SDN controller/server is improved in 92% of the articles considered for the comparison in Table 2.

The total number of 80 articles considered for the complete analysis out of which 15 articles potentially work in software defined and edge computing networks. However, 14 articles pilots about low latency. The E_Improved 28% states that energy consumption improved in 28% of the articles considered and where energy is improved except the reliability, the rest of the parameters have suddenly improved, it can be identified from Table 2. The graph also shows that there can be a strong prediction that when energy improves in the system, network latency, packet frequency and response time all of these quality-of-service parameters improve. Figure 5 represents the environmental impact and states that the carbon emission footprint reduces when we use software defined networks. Figure 6 suggests that tool NS-3 is minimally utilized in SDN environment, similarly GEANT, BT North America and CONUS topology are minimally utilized. Mininet simulators and Iperf protocol are maximally utilized in the field of SDN for testing and experimenting purposes. Python language is utilized 11% and the most prominent Controller can be named Ryu Controller. This article will support future research in the field of SDN, for finding out a good technique and good tool which is prominently utilized in the current field of research. Table 3 gives a summary of articles where the schema of the table is described as research. Challenge, proposed solution in the published article and what can be the future scope for the proposed solution.

Table 2 Summary of Limitations in Different Techniques Used in SDN for Load Balancing

Technique Used/ Limitations	Server Response Time	Packet Frequency	Packet Reliability	Network Latency	Environment Friendly Energy Efficiency	Network Scalability	Tool Used	Algorithm/ Approach/ Method
Stochastic Geometric Tool [17]	Improved	Improved	Improved	Less	Dropped	Not Considered	NS-3	{Joint Optimization Algorithm}
Deep Programmability Distributed Manner [18]	Improved	Improved	Improved	Less	Not Considered	Not Considered	Python and Gurobi	MILP Mathematical Equation
Dynamic	Improved	Improved	Not	Less	Not	Not	Java and	DREAM

REVIEW ARTICLE

Resource Allocation [19]			Considered		Considered	Considered	Floodlight Controller	
Hybrid Clustering [20]	Improv ed	Improve d	Not Considered	Less	Not Considered	Not Consider ed	Python and Iperf	K means++ and Spectral Clustering
Congestion Control [21]	Improved	Not Considered	Improved	Not Considered	Not Considered	Not Considered	Mininet Open Flow Switches, POX Controller	Software Defined Congestion Control
Controllers Deployment [27]	Improved	Improved	Improved	Less	Improved	Yes	Opendaylight Controller, Mininet	Reliable Load Balance Aware Multi-Controller Deployment (RLMD)
Multiple Controllers [35]	Improved	Improved	Improved	Less	Improved	Yes	Ryu Controller, Mininet, OpenFlow, Linear Topology	Multiple Distributed Controller Load Balancing
Switching Technology [36]	NA	Not Consider ed	Not Consider ed	Less	Not Considered	Yes	Mininet 2.1.0 Floodlight Controller Open vSwitch	Approach: Variance Analysis and Probability Assignment
Migrating Switches in SDN [38]	Improved	Improve d	Not Considered	Less	Not Considered	Yes	Ryu Controller, Mininet, OpenFlow,	Load Balancing Algorithm for Migrating Switches
Deep Deterministic Policy Gradient [39]	Improved	Improved	Not Considered	Less	Improved	Yes	Ryu Controller, OpenFlow 1.3 protocol Mininet and Pytorch	Automatic Load Balancing Architecture based on Reinforcement Learning.
Prediction Model [40]	Improved	Improved	Not Considered	Less	Dropped	Not Considered	Ryu Controller, Open vSwitch, OpenFlow 1.3, Mininet and GEANT Topology, Iperf protocol	Adaptive Load-Balancing Architecture based on Link-State Prediction (ALBLP)
Minimum Number of Connections [45] [49] [51]	Improved	Improved	Not Considered	Less	Not Considered	Not Considered	Python and CONUS Topology	Switch Migration Based Controller Placement Algorithm

Switch Migration and Decision Making [50] [54]	Improved	Improved	Not Considered	Less	Improved	Yes	Beacon Controller, Mininet	Switch Migration Decision Making
SALB [52] [55]	Improved	Improved	Not Considered	Less	Not Considered	Yes	Java, Floodlight Controller and Geant Topology	Self Adaptive Load Balancing Scheme

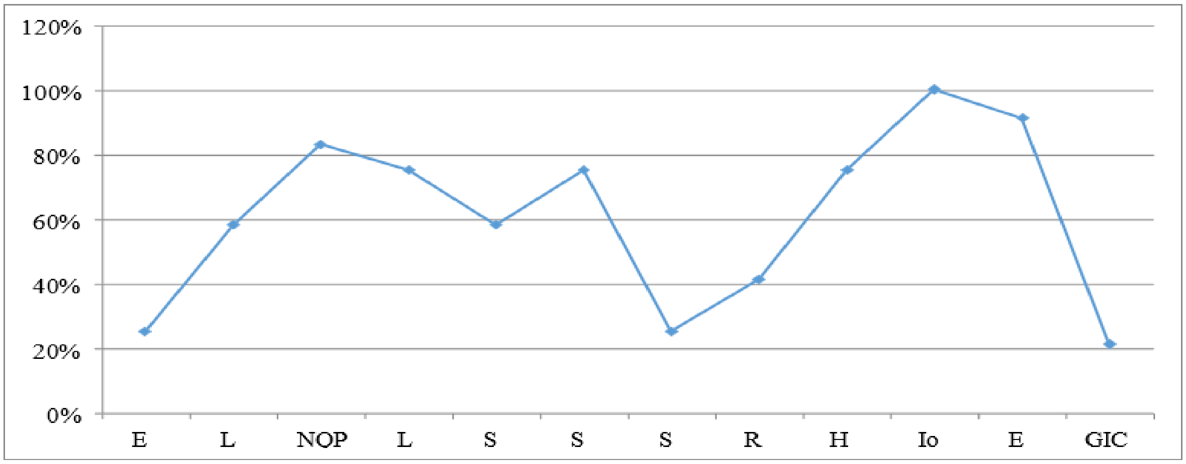

Figure 5 Carbon Emission Improvement

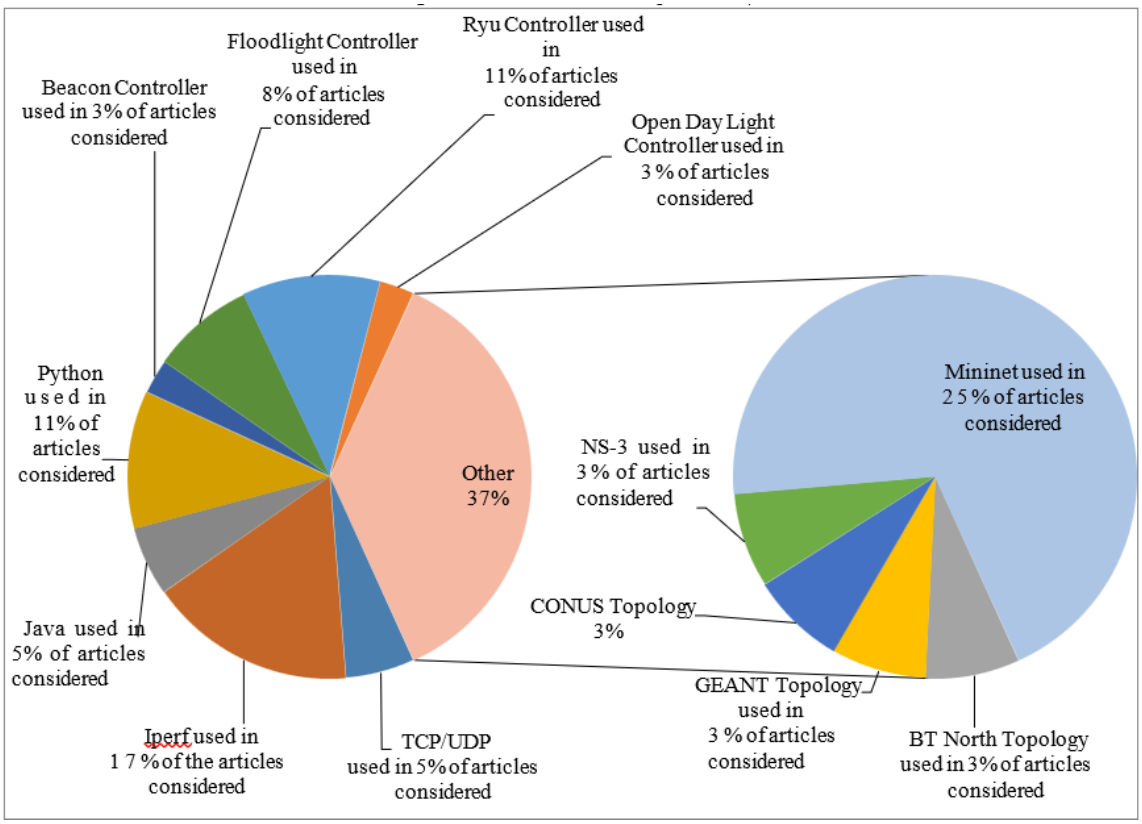

Figure 6 Collaborative Analysis of Tools and Techniques

REVIEW ARTICLE

Table 3 Summary of Articles with Research Challenges and Proposed Solutions

Ref. No.	Research Challenge	Proposed Solution	Future Scope
[20]	Resource allocation is done using several algorithms, but resource allocation requires optimization. At the time of resource allocation memory is not utilized.	A special k-mean++ algorithm along with spectral clustering algorithm is used to form the clusters. Capacitive memory is also utilized in the network.	Data mining methods can be used to extract out the different set of results for comparison.
[22]	The stability and robustness of the network is hampered by congestion. Data flow conditions are less monitored. The judgments for congestion are taken on the basis of packet loss.	A congestion control algorithm which uses link layer discovery protocol and is based on software defined congestion control mechanism.	The work can be extended and can be applied to a complex data network center.
[27]	Decoupling of the plane in SDN causes reliability challenges. With the separation of planes controllers drown in managing and balancing the load of the traffic.	A new strategy proposed by RLMD which improved reliability and balances the load.	Certain scenarios must be considered such as migrating switches and dynamic load balancing.
[36]	Using a single controller to manage the heavy traffic drops the quality of the service. There are very a smaller number of techniques introduced with scalability and also addressing the reliability challenges at the same time.	It identified the root cause of the problem and came up with a new algorithm: MDCLB on ISSDN-IoT.	It can be utilized in the road network and on the network to manage the congestion.
[39]	Open Shortest Path First (OSPF) lacks in dealing with multiple links at the same time and uses fixed packet forwarding scheme which is congestion prone. Equal Cost Multipath Routing (ECMP) it also uses fixed packet forwarding scheme, and it uses numerous links to use maximum of the bandwidth Heuristic Algorithm (HA) due to less or very less experience HA can generate models for only stable/fixed networks.	Proposed a new algorithm ALBRL which improved throughput as compared to OSPF, RSIR and EARS.	Extending the topology and improving the convergence rate of deep reinforcement- learning algorithms.
[53]	Disrupted distribution of the traffic in the network. Distance monitoring of switches from controller and vice-versa.	In high-load conditions a new scheme is proposed which dynamically adjusts the threshold value. The proposed scheme is highly suitable for a large real-time-network.	Due to limitation of hardware Base threshold maximum level cannot be increased, solution related to this can be explored.

REVIEW ARTICLE

[56]	To stop/minimize the wastage of the MEC resources and introduce load balancing. To support scalability in heavy-network traffic scenarios.	Proposed a new agile and flexible architecture capable enough to meet the requirements of IoV.	Remote clouds can be utilized for allocation of the controllers.

5. CONCLUSION

Many techniques can be utilized to attain better performance in a communication network. However, not all of the techniques fulfil the requirement of the framed objectives. To identify prominent and useful techniques a review about potential techniques is discussed and evolved up with transitory results. Although there are very powerful tools in existence for experimental implementation, due to massive growth in the field of networking, there is always a scope of improvement. All the techniques and tools considered are helpful in decision making for researchers in the core field of SDN. The reduction in carbon emission by using SDN can be significant by using the tools considered for evaluation. 25% of the total carbon emission can be reduced by using soft implementations rather than hardware equipment installation. According to the reports 2% of the carbon emission is from network industry. Out of 2% it can be reduced to 1.50%, if SDN is utilized. The QoS table used in the article suggests that almost all the authors have considered these parameters for evaluating and analyzing quality of service. All the tools and techniques performed well according to the specification considered for the implementation environment. The analysis extracted out leads to a clear comprehension for the readers of which tool with which technology and which technique to be implemented to reach an optimized solution. The analytical view in the field of SDN addresses energy efficiency, load balancing and resource allocation challenges in massive user environment. The parallel processing approach plays a vital role in addressing the research challenge and it can be optimized in future. There are 11 QoS parameters considered for performance evaluation where Stochastic geometric tool technique covers 9 QoS parameters out of 11 covering almost 81% of the parameters. The extended part of the article can be used as a base for finding out solutions in SDN's. The gaps identified in the article and the techniques suggested in the article which are mostly utilized can be collaboratively used for energy efficiency, balancing load, improving network quality performance, scalability, environment impact and security.

REFERENCES

[1] Baha Uddin Kazi Md Kawasarul Islam uhammad Mahmudul Haque Siddiqui Muhammad Jaseemuddin, "Asurey on Software Defined Network-Enabled Edge Cloud Networks: Challenges and Future Research Directions", Network, Vol 5(16) 2025 pp.1-28.

[2] Mishra S.R. Shanmugam B Yeo K.C. Thennadil S., "SDN-Enabled IoT Security Frameworks-A Review of Existing Challenges", Technologies, Vol 13(3) 2025 pp. 1-38.

[3] Rafique W Lianyong Qi Ibrar Yaqoob Muhammad Imran Raihan Ur Rasool Wanchun Dou, "Complementing IoT Services Through Software Defined Networking and Edge Computing: A Comprehensive Survey", IEEE Communications Surveys & Tutorials, Vol. 22, No. 3, 2020 pp. 1761-1800.

[4] Hu P Wai Chen Chunming He Yiping Li Huansheng Ning "Software-Defined Edge Computing (SDEC): Principle, Open IoT System Architecture, Applications, and Challenges", IEEE Internet of Things Journal, Vol. 7, No. 7, 2020 pp. 5934-5945.

[5] Qi C Wu J Hu H Cheng G Liu W Ai J Yang C, "An intensive security architecture with multi-controller for SDN", In Proceedings of the IEEE Conference On Computer Communications Workshops (INFOCOM WKSHPS), San Francisco, CA, USA, 10–14 April; 2016, pp. 401–402. doi: 10.1109/INFCOMW.2016.7562109.

[6] Dai M Zhou Su Ruidong Li Shui Yu, "A Software- Defined-Networking-Enabled Approach for Edge-Cloud Computing in the Internet of Things", IEEE Network, IEEE,Vol. 35, No. 5, 2021 pp. 66-73.

[7] Tanha M Sajjadi D Ruby R Pan J, "Capacity-aware and delayguaranteed resilient controller placement for software-defined WANs", IEEE Trans. Netw. Serv. Manag. Vol. 15, No. 3, 2018 pp.991–1005.

[8] Ma Y W Chen J L Tsai Y H Cheng K H Hung W C, "Load-balancing multiple controllers mechanism for software-defined networking", Wirel. Pers. Commun. , 94, 2017 pp. 3549–3574.

[9] Yao G Bi J Li Y Guo L On the capacitated controller placement problem in software defined networks. IEEE Commun. Lett., 18, 2014 pp. 1339–1342.

[10] Heller B Sherwood R McKeown N The controller placement problem. ACM SIGCOMM Comput. Commun. Rev., 42, 2012 pp. 473–478.

[11] Wang G Zhao Y Huang J Wu Y An effective approach to controller placement in software defined wide area networks. IEEE Trans. Netw. Serv. Manag., Vol. 15, 2018 pp. 344–355.

[12] Wang G Zhao Y Huang J Wang W The controller placement problem in software defined networking: A survey. IEEE Netw. Vol. 31, 2017 pp. 21–27.

[13] Jalili A Keshtgari M Akbari R Javidan R Multi criteria analysis of controller placement problem in software defined networks. Comput. Commun. Vol. 133, 2019 pp.115–128.

[14] Choumas K. Giatsios D Flegkas P Korakis T SDN Controller Placement and Switch Assignment for Low Power IoT. Electronics, 9, 2020 pp.325.

[15] Abdelaziz A Fong A T Gani A Garba U Khan S Akhunzada A Talebian H Choo K. K.R Distributed controller clustering in software defined networks. PLoS ONE, 12(4), 2017 e0174715.

[16] Kiran N Liu X Wang S Yin C VNF Placement and Resource Allocation in SDN/NFV-Enabled MEC Networks. In Proceedings of the 2020 IEEE Wireless Communications And Net-Working Conference Workshops (WCNCW), Seoul, Korea, 6–9 April; 2020 pp. 1–6.

[17] Gebremariam Anteneh A Usman Muhammad Bassoli Riccardo Granelli Fabrizio SoftPSN: Software- Defined Resource Slicing for Low-Latency Reliable Public Safety Networks, Wireless Communications and Mobile Computing, Wiley | Hindawi, Vol. 2018, pp. 1-7.

[18] Hongming Li Dongxiu Ou Iftikhar Rasheed Meiting Tu1 2022 A Software-Defined Networking Roadside Unit Cloud Resource Management Framework for Vehicle Ad Hoc Networks, Journal of Advanced Transportation, Wiley | Hindawi, Vol. 2022, pp. 1-13.

[19] Yao Hu Ting Peng Lianming Zhang 2017 "Software- Defined Congestion Control Algorithm for IP Networks", Scientific Programming, Wiley | Hindawi, Vol. 2017, pp. 1-8.

[20] Muhammad Shuaib Qureshi,Muhammad Bilal Qureshi Muhammad Fayaz Wali Khan Mashwani Samir Brahim Belhaouari A comparative analysis of resource allocation schemes for real-time services in high-

performance computing systems, IJDSN, SAGE, Vol. 16 (8), 2020 pp 1-35.

[21] Pedro Neves Rui Calé Mário Rui Costa Carlos Parada Bruno Parreira Jose Alcaraz-Calero Qi Wang et. al. The SELFNET Approach for Autonomic Management in an NFV/SDN Networking Paradigm, International Journal of Distributed Sensor Networks, Hindawi, Vol. 12(2), 2016, pp. 1-17.

[22] Masoud Moshref Minlan Yu Ramesh Govindan Amin Vahdat DREAM: Dynamic Resource Allocation for Software-defined Measurement, SIGCOMM'14, ACM, 2014, pp. 419-430.

[23] Mahdi Sarbazi Mehdi Sadeghzadeh Seyyed Javad Mir Abedini Improving resource allocation in software- defined networks using clustering, Cluster Computing 23, Springer, 2020, pp. 1199-1210.

[24] Thabo Semong Thabiso Maupong Stephen Anokye Kefalotse Kehulakae Setso Dimakatso Gabanthone Boipelo et. al. Intelligent Load Balancing Techniques in Software Defined Networks: A Survey, Electronics, MDPI, Vol.9(7), 2020 pp. 1091.

[25] Lin W Zhang L The load balancing research of SDN based on ant colony algorithm with job classification. In 2nd Workshop on Advanced Research and Technology in Industry Applications (WARTIA-16); Atlantis Press: Paris, France 2016 pp. 472-476.

[26] Hu T Yi P Zhang J Lan J Reliable and load balance- aware multi-controller deployment in SDN. China Commun., Vol. 15, 2018, 184–198.

[27] Moy J T OSPF Version 2 (RFC 2328); Ascend Communications Inc.: Alameda, CA, USA 1998, pp. 1- 244.

[28] Moy J T OSPF: Anatomy of an Internet Routing Protocol; Addison-Wesley Professional: Boston, MA, USA 1998, pp.1-331.

[29] Barreto F Wille E C Nacamura L Jr Fast emergency paths schema to overcome transient link failures in ospf routing. arXiv, Vol. 4(2), 2012, pp.17-34.

[30] Nelakuditi S Lee S Yu Y Zhang Z L Chuah C N Fast local rerouting for handling transient link failures. IEEE/ACM Trans. Netw., Vol. 15, 2007, pp. 359–372.

[31] Aversa L Bestavros A Load balancing a cluster of web servers: Using distributed packet rewriting. In Proceedings of the 2000 IEEE International Performance, Computing, and Communications Conference, Phoenix, AZ, USA, 5–8 February; 2000, pp. 24–29.

[32] Katz E D Butler M McGrath R A scalable HTTP server: The NCSA prototype. Comput. Netw. ISDN Syst., Vol. 27, 1994, pp. 155–164.

[33] Chou L D Yang Y T Hong Y M Hu J K Jean B A genetic-based load balancing algorithm in Open Flow network. In Advanced Technologies, Embedded and Multimedia for Human-Centric Computing; Springer: Berlin/Heidelberg, Germany, 2014 , pp. 411–417.

[34] Suwandika I P A Nugroho M A Abdurahman M Increasing SDN performance using load balancing scheme on Web Server. In Proceedings of the 2018 6th International Conference on Information and Communication Technology (ICoICT), Bandung, Indonesia, 3–5; 2018, pp. 459–463.

[35] Himanshi Babbar Shalli Rani Divya Gupta Hani Moaiteq Aljahdali Aman Singh and Fadi Al-Turjman Load Balancing Algorithm on the Immense Scale of Internet of Things in SDN for Smart Cities, Sustainability, MDPI, 13, 2021, pp. 9587.

[36] Hong Zhong Qunfeng Lin Jie Cui Runhua Shi and Lu Liu An Efficient SDN Load Balancing Scheme Based on Variance Analysis for Massive Mobile Users, Mobile Information Systems, Hindawi, Vol. 2015, pp. 1-9.

[37] Hosny W Fouad Aly Generic Controller Adaptive Load Balancing (GCALB) for SDN Networks, Journal of Computer Networks and Communication, Hindawi, 2019, pp. 6808693.

[38] Himanshi Babbar Shalli Rani Mehedi Masud Sahil Verma Divya Anand Nz Jhanjhi Load Balancing Algorithm for Migrating Switches in Software-Defined Vehicular Networks, Computers, Materials & Continua, Tech Press Science, Vol. 67 No. 1, 2021, pp. 1301-1316.

[39] Junyan Chen Yong Wang Jiangtao Ou Chengyuan Fan Xiaoye Lu et. al ALBRL: Automatic Load-Balancing Architecture Based on Reinforcement Learning in Software-Defined Networking, Wireless Communications and Mobile Computing, Hindawi, Vol. 2022, 2022 pp. 1-17.

[40] Junyan Chen Yong Wang Xuefeng Huang Xiaolan Xie Hongmei Zhang Xiaoye Lu ALBLP: Adaptive Load- Balancing Architecture Based on Link-State Prediction in Software-Defined Networking, Wireless Communications and Mobile Computing, Wiley | Hindawi, Vol. 2022., 2022 Pp. 1-16.

[41] Seungsoo Lee Jinwoo Kim Seungwon Woo Changhoon Yoon Sandra Scott-Hayward Vinod Yegneswaran A Comprehensive Security Assessment Framework for Software-Defined Networks, Computers & Security, Elsevier, 2020, pp.1-20.

[42] Vijay Varadharajan Kallol Karmakar Uday Tupakula Michael Hitchens A Policy-Based Security Architecture for Software-Defined Networks, IEEE Transactions On Information Forensics And Security, Vol. 14 (4), 2019, pp.897-912.

[43] Taehoon Eom Jin B Hong Seongmo An Jong Sou Park Dong Seong Kim A Systematic Approach to Threat Modeling and Security Analysis for Software Defined Networking, IEEE Access, Vol. 7, 2019, pp. 137432-137445.

[44] Abdullah Al Hayajneh Md Zakirul Alam Bhuiyan Ian McAndrew Improving Internet of Things (IoT) Security with Software-Defined Networking (SDN), Computers, MDPI,9,8, 2020, pp. 1-14.

[45] Ali J Roh B-H Lee S QoS improvement with an optimum controller selection for software-defined networks. PLoS ONE, 14, 2019, e0217631.

[46] Ijaz Ahmad Suneth Namal Mika Ylianttila Andrei Gurto Security in Software Defined Networks: A Survey, IEEE Communication Surveys & Tutorials, Vol. 17, No. 4, 2015, pp. 2317-2346.

[47] Karmous N. Aoueileyine M.O. Abdelkader M. Romdhani L. Youssef N., "Software-Defined Networking-Based One-versus-Rest Strategy for Detecting and Mitigating Distributed Denial-of-Service Attacks in Smart Home Internet of Things Devices,Sensors, Vol. 24 (15) 2024, pp. 1-24.

[48] Mohammed Nsaif Gergely Kovásznai Anett Rácz Ali Malik Ruairí de Fréin 2021 An Adaptive Routing Framework for Efficient Power Consumption in Software- Defined Datacenter Networks, Electronics, MDPI, 2022, pp. 1-18.

[49] Simoes, Rhodney, Kelvin Dias, and Ricardo Martins. "Dynamic Allocation of SDN Controllers in NFV-based MEC for the Internet of Vehicles." Future Internet 13.11 (2021): 270.

[50] Raül Muñoz Ricard Vilalta Noboru Yoshikane Ramon Casellas Ricardo Martínez Takehiro Tsuritani et al Integration of IoT, Transport SDN, and Edge/Cloud Computing for Dynamic Distribution of IoT Analytics and Efficient Use of Network Resources, Journal of Light- wave Technology, Vol. 36, No. 7, 2018, pp. 1420-1428.

[51] Farshin A Sharifian S A chaotic grey wolf controller allocator for Soft-ware Defined Mobile Network (SDMN) for 5th generation of cloud-based cellular systems (5G), Computer Communication, Vol.108 No. 2, 2017, pp.94–109.

[52] Ali, Jehad, and Byeong-hee Roh. "An effective approach for controller placement in software-defined internet-of-things (SD-IoT)." Sensors 22, no. 8 (2022): 2992.

[53] Farooq M.S. Riaz S, Alvi A. " Security and Privacy Issues in Software-Defined Networking (SDN): A systematic literature review", Electronics, Vol. 12, 2023,pp. 3077.

[54] Wang C Hu B Chen S Li D Liu B A switch migration-based decision-making scheme for balancing load in SDN. IEEE Access, 5, 2021, 4537–4544.

[55] Askar S K Adaptive load balancing scheme for data center networks using software defined network. Sci. J. Univ. Zakho, Vol. 4, 2016, pp. 275–286.

[56] Alex Mavromatis Carlos Colman-Meixner Aloizio P Silva Xenofon Vasilakos Reza Nejabati Dimitra Simeonidou A Software-Defined IoT Device Management Framework for Edge and Cloud Computing, IEEE Internet of Things Journal, Vol. 7, No. 3, 2020, pp. 1718-1735.

REVIEW ARTICLE

Authors

Dr. Lokesh Pawar has rich experience of 12+ years and currently he is working as an Associate Professor and Head Academic Operations at Chitkara University, Punjab. Earlier in his career he had filed 6 patents whereas 1 is currently in the Grant Phase. He has published his articles in almost all reputed publications, Elsevier, Springer, Taylor and Francis, and IEEE. He has published more than 25+ SCOPUS Indexed articles. He has also bagged a few SCI articles. His current interest is in Internet of Things, SDN and Edge Computing. He is a certified python trainer and carries good knowledge about objective-based education system. He has recently upgraded his armory by getting certified from IIT Kharagpur for Accreditation and objective based education through NPTEL.

Dr. Rohit Bajaj is working as Professor in Chandigarh University, and he is an active member of research and innovation cell in the University. He has bagged number more than 40 Scopus Indexed Articles and SCI articles on his name. He has published several books internationally. Hiskey area of research is Machine Learning and Networks. He has a keen interest in hybrid research.

Prof. Darpan Anand is a distinguished academic and researcher with over 22 years of experience in teaching, industry, and research. Currently serving as a Professor and Associate Dean in the Department of Computer Science and Engineering at Chitkara University, India, he has significantly contributed to both academia and industry.

How to cite this article:

Lokesh Pawar, Rohit Bajaj, Darpan Anand, "Resource Allocation Load Balancing Secure Schemes in Software Defined Networks: An Analytical Review", International Journal of Computer Networks and Applications (IJCNA), 12(4), PP: 523-537, 2025, DOI: 10.22247/ijcna/2025/32.