Webology, Volume 19, Number 1, January, 2022

Optimisation Methods for the Controller Placement Problem in SDN: A Survey

Neamah Sattar Radam
College of Computer Science and Information Technology, University of Anbar, Ramadi, Iraq.
E-mail: nea19c1010@uoanbar.edu.iq

Sufyan Al-Janabi
College of Computer Science and Information Technology, University of Anbar, Ramadi, Iraq.
E-mail: sufyan.aljanabi@uoanbar.edu.iq

Khalid Shaker
College of Computer Science and Information Technology, University of Anbar, Ramadi, Iraq.
E-mail: khalidalhity@gmail.com

Received September 16, 2021; Accepted December 15, 2021
ISSN: 1735-188X
DOI: 10.14704/WEB/V19I1/WEB19207

Abstract

Software-defined networking (SDN) has emerged in response to increasing requirements for new networks and expansion of Internet coverage. Modern needs exceed the limitations of traditional networks, for which, to simplify management, SDN is proposed as a promising paradigm that separates the control and data planes, allowing for the programming of network configuration. SDN deployment and applications are directly affected by the controller position. Single or multiple controllers are used in SDN architecture to enable programmable, flexible, and scalable configurations. Multiple controllers are essential in the current SDN, and various solutions have been recently developed to improve scalability and placement selection. In this study, the Controller Placement Problem (CPP) is explored using objective optimisation with proposed algorithms. An overview of SDN issues and the controller role is provided through its three-plane architecture with a focus on scalability and reliability. In addition, a comprehensive problem review is discussed on the basis of a well-known compendium of available solutions. Finally, relevant open problems and future research challenges are identified.

Keywords

CPP, SDN, Multi-controllers, Data Plane, Control Plane, Optimisation Algorithm.

Introduction

The continuous development of information technology has enabled the Internet to create a complex infrastructure and immense foundation that considerably affect the manner in which people work and live. In advanced regulatory frameworks for modern network systems, software-defined networking (SDN) is a typical paradigm and can become a common offering. Thus, the underlying hardware and software can be isolated from the control logic, which is identified as a software component and located in a server, called a console. In a large-scale organisation, deployment of multiple and different controllers to expand performance is one of the most challenging SDN issues (Sahoo et al., 2017).

The approval of remote objectives in wired and wireless networking systems regulated solutions for the Controller Placement Problem (CPP). SDN is perhaps a novel configuration model that enables flexible and adaptive network management. With the increased organisational capacity, the Single Console SDN presents various disadvantages on both performance and scalability. As a solution, distributed controller multicasting may be a promising strategy to achieve fault tolerance and adaptability (Ateya et al., 2019). Figure 1 shows the various scalability approaches.

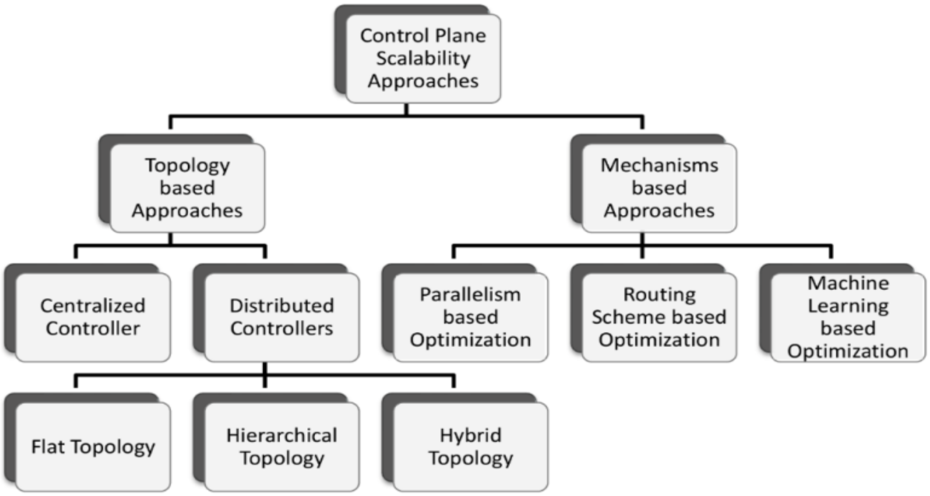

Figure 1 Classification of control plane scalability approaches (Abuarqoub, 2020)

In traditional networks, system complexity and closure cause challenges in partitioning and managing control and data levels, including all components of an organised network such as switches and routers. Creating different networks are apparently affected by the location and number of controller regulators. Here, the CPP (number and position of controllers between switch and controllers) is a dependent factor. Challenges arise for multiple controllers, such as scalability, reliability, consistency and load balancing.

Two typical scenarios of SDNs are as follows. First, a delay-sensitive usage case is applied and second, network congestion may occur in real conditions. Both scenarios have integrated customer navigation. The SDN architecture efficiency should be considered in the design (Košťál, Bencel, Ries, Trúchly, & Kotuliak, 2019) to achieve flexible and adaptive resource management. With the substantial development and traffic volume of network activity, the subdomain division of SDN controllers may cause an unbalanced load distribution and reduce the configuration performance (G. Li, Wang, & Zhang, 2019).

With its feature that separates the control and data (information) planes, SDN is proposed as a new and modern model that provides flexible and adaptable network management (Chai, Yuan, Zhu, & Chen, 2019). In a general framework, a common system can be developed or created to determine the worst-case performance of a control plane. Suitable solutions arrangements are needed for strategies of fault recovery and guessing methodology to reduce the maximum controller utilisation (MCU) in the failure condition (Xie et al., 2019). Indeed, the best cost and location of controllers can be determined by applying a genetic algorithm (GA).

In SDN, the most important concern is the controller placement. The independent controller group policy can be used to ensure distribution with the lowest response times. Such cases require a suitable algorithm based on node scores. The console mode is usually addressed in two stages: first, candidate console instances are identified and second, the network is divided into multiple domains with one controller each (Alowa, 2020). Enabling an efficient load balancing (LB) is the main objective in the SDN with multiple and controllable distributed architectures. As such, several mathematical models for controller placement under multi-module switching mapping are recently developed.

This paper is structured as follows. Section 1 introduces to the research topic. Section 2 presents a more systematic overview of SDN and CPP. Current research status methods and heuristic algorithms are discussed in Section 3, including comparison tables to clarify the characteristics of reviewed literature. Finally, Section 4 presents the conclusions and suggestions for future works.

Overview of SDN and CPP

Figure 2 illustrates the different variants of SDN interconnections, CPP optimisation, and the challenges of solving CPP in data centres and Wide Area Networks (WANs). Figure 2a represents the separation between the control unit and data planes. This problem can be partially solved with redundancy by using a centralised method through network management functions in one control for Node to Controller (N2C) and Controller to

Controller (C2C) connections. Figure 2b shows an SDN with multiple controllers (or a distributed control layer) for the network management level. Each controller handles several switches (Выборнова).

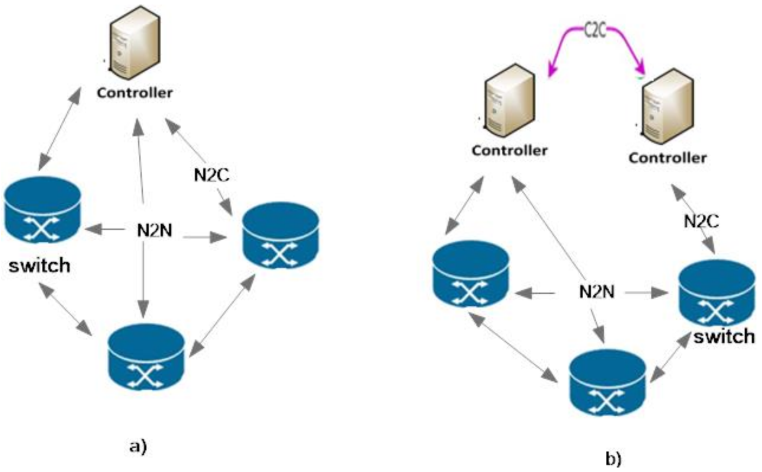

Figure 2 Variants of SDN architecture (Выборнова)

A. SDN Architecture

Figure 3 shows the basic SDN architecture with three layers, namely, upper, lower, and middle. The middle represents the 'brain', a central control to follow and manage connected devices through OpenFlow protocols. The middle layer also represents programmable controllers that work according to different rules and policies. The Southbound API is used to interact and connect with the lower infrastructure (data) layer while the Northbound API is used to interact with the upper application layer and its services. In addition, the middle control layer uses the West–Eastbound (W/E) interfaces to communicate within controller groups (Lu, Zhang, Hu, Yi, & Lan, 2019).

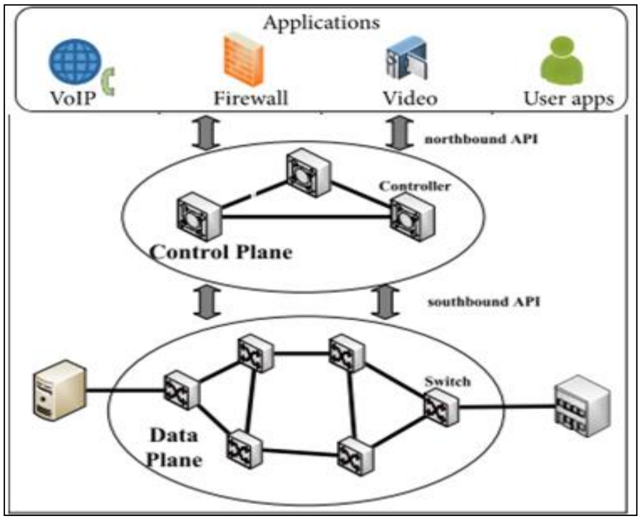

Figure 3 Basic architecture of SDN (Lu et al., 2019)

B. General Formulation of CPP

The typical formulation of SDN for the CPP is solved according to Equation (1) (Isong, Molose, Abu-Mahfouz, & Dladlu, 2020).

$$\mathbf{G} = (V, E, \mathbf{U}), \qquad (1)$$

where G = graph, **U** = set of *k* controllers, E = set of edges (physical links) among switches or controllers and V = set of *n* switches. As such, $n = |V|$ refers to the number of nodes and $k = |\mathbf{U}|$ indicates the number of controllers. Thus, to improve the objective function, the value of *k* and $\mathbf{U} \rightarrow V$ mapping must be obtained.

C. Optimal Objective of CPP

Figure 4 shows the various optimal objectives for CPP, which can be determined using solutions that are commonly based on four different factors: latency, reliability, cost and multi-objective. Each of these factors are summarised below.

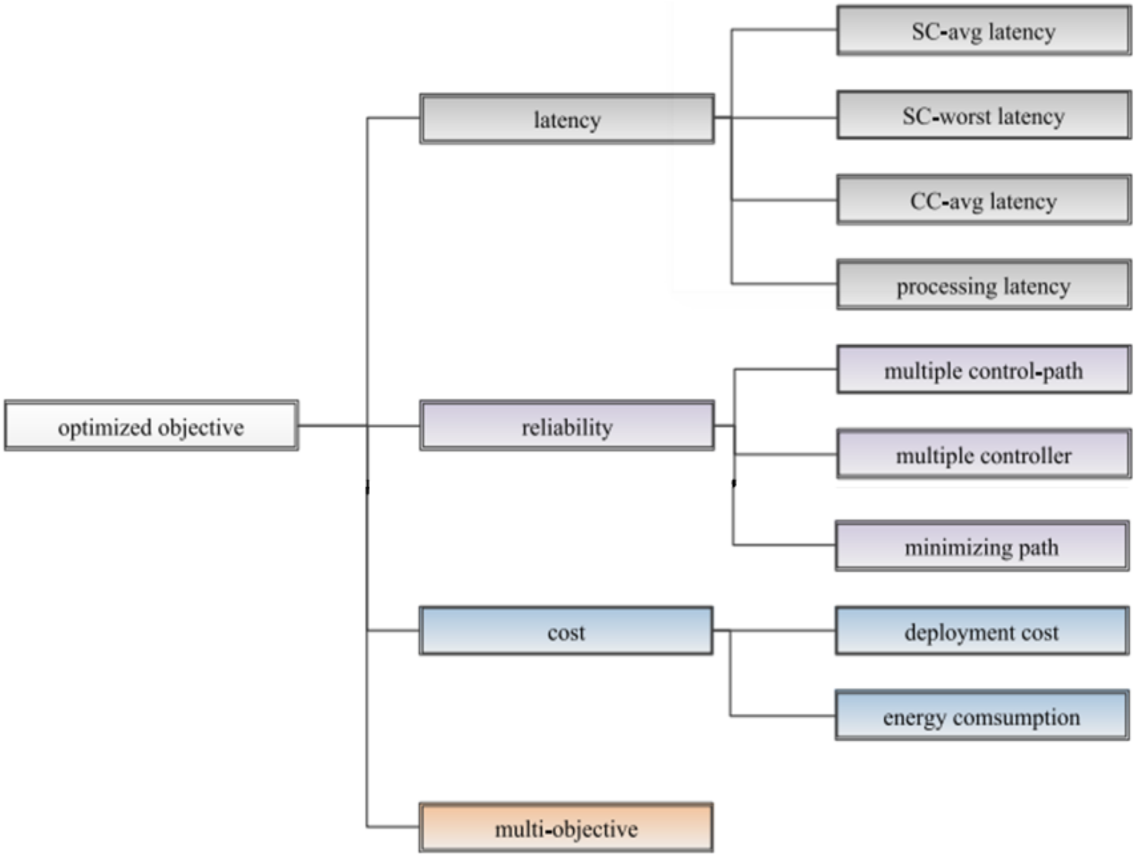

Figure 4 Classifications of optimal objectives (Lu et al., 2019)

Latency: The switches and the controller can exchange packets through SDN latency messages, where the lowest value is calculated using algorithms such as dragonfly (DF) and firefly algorithms (FFA). For a single or multiple control site, the exact objective functions can be calculated using Equation (2) (Maytree Ramasamy, 2019):

$$\text{Latency} = \frac{Number\ of\ iteration}{Number\ of\ controllers}\ . \qquad (2)$$

Latency can be determined using the ping program, which measures a round-trip time (RTT) and thus divides the result into two iterations of approximately one-way latency. Four types of latency can be categorised as follows: two controllers' average latency, switch and controller worst latency, switch and controller average latency, and average latency (Chen et al., 2018).

Reliability: In network failure testing, disconnections of network contents (such as switches or controllers) in the SDN can cause significant packet loss and performance degradation. Thus, reliability is considered as a highly important factor when deploying and placing controllers. A non-congestion plan with properly distributed locations and regions is necessary to ensure that prerequisites on reliability and bandwidth are satisfied (S. Liu, Steinert, & Kostic, 2018).

Cost: Specifically, SDN costs mainly include network construction time and operations and maintenance costs. Cost can be calculated using Equation (3):

$$C = \min\ (Cs + Cl + Ct), \qquad (3)$$

where C = cost, Cs = cost of controllers, Cl = cost of connecting S2C and Ct = cost of controller interconnection.

For example, the resilience and efficient resource use for vehicle network can be determined using the Software Defined Vehicle Network (SDVN) model. This solution allows for the emergence of new unused smart transportation services. In an SDVN context, troubleshooting an issue can be achieved with adaptive controller placement and integration of the replacement cost into the model (Toufga, Abdellatif, Assouane, Owezarski, & Villemur, 2020). This method also considers the deployment to ensure its minimum costs and thereby optimise the network performance.

Multi-objective: Multiple performance metrics for CPP can be used depending on the different implementations or solutions. Thus, CPP can be considered as a multi-objective optimisation problem, as shown in Equation (4) (Isong et al., 2020):

$$\text{Multi-objective } (M_{ob}) = \max\ [\text{Latency, Reliability, Cost}], \qquad (4)$$

Multi-Objective techniques can measure varying aims by using different algorithms. An example is the Multi-Criteria Decision Algorithm (MCDA), which deals with propagation latency, load and failure. Multi-criteria optimisation algorithms are applied to solve the CPP and obtain comprehensive optimisation of controllers by applying a multi-criteria decision algorithm (MCDA), such as Particle Swarm Optimisation (PSO) and FF meta-population to determine the optimum controller locations (Sahoo et al., 2017).

Other options in aggregating controllers and configuring a static network include the Improved PSO (IMPSO) and Multi-Objective Anti-Lion Optimisation (MO-ALO) (Maytree Ramasamy, 2019). Other multi-objective optimisation algorithms depend on multi-criteria decision, including the Multi-Start Hybrid Non-Dominated Sorting Genetic Algorithm (MHNSGA) and Adaptive Bacterial Foraging Optimisation (ABFO) to solve of CPP (Lu et al., 2019).

D. Multi-Controller Placement Strategy

FloodLight and OpenDayLight are popular solutions due to their network performance and unique features. A centralised controller includes programming of administration and configuration instead of separate configurations, and thereby handles decision making for routing and switching of all connected devices in the network. The controller software also runs the Northbound and Southbound APIs to communicate with the Application and Infrastructure layers (Ali, Lee, Roh, Ryu, & Park, 2020).

The control plane comprises one or more controllers that serve as the SDN brain. For a centralised controller, efficiency and scalability are crucial issues. As the network grows in size, the increased flow processing may present challenges for the unique and centralised controller. As such, large-scale networks require multi-controller as the best option. Figure 5 illustrates a typical hierarchical design of the multi-controller architecture. However, the issue of load imbalance remains, which exerts a significant effect on SDN efficiency (Andishmand, Mohammdi, & Mostafavi)

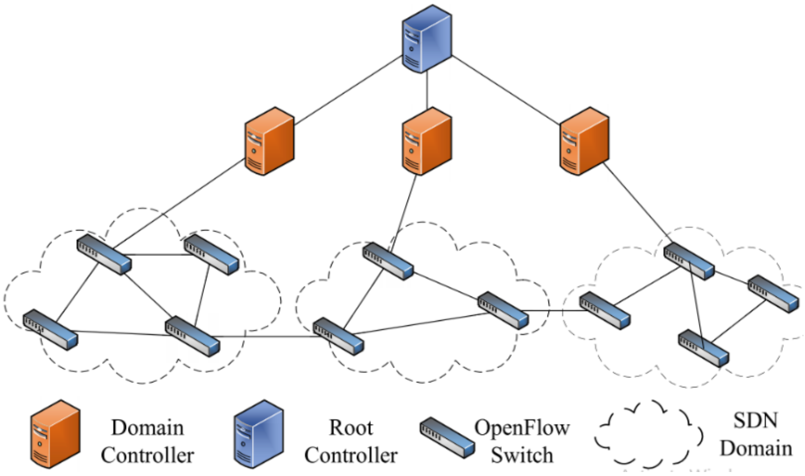

Figure 5 Hierarchical design of multi-controller (Hu, Guo, Yi, Baker, & Lan, 2018)

Location of Controllers by Using different Methods and Algorithms

Various approaches, methods and algorithms are used to improve and solve a CPP with multi-controllers, several of which work on the locations of switches and controllers while others consider only the controller placements. These approaches are summarised in the following subsections.

A. Heuristic Approach

This optimisation approach is widely used to solve CPP. The present study reviews most of the important literature using heuristic approaches and algorithms. Sahoo, Kshira Sagar, et al. (2017) used the CPP–PSO in a population-based stochastic technique. The optimal solution is derived from the population, and the position is updated using FFA, also called CPP-FFA, which depends on characteristics similar to the flickering of fireflies (Sahoo et al., 2017). In Wireless CPP (WCPP) Zilberman, Aviram et al. (2021) considered SBI on the basis of unlicensed 4G LTE network. Another approach applies simulated annealing (SA) of larger networks and use clustering on the basis of Hill-Climbing with Simulated Annealing (HetNet-LTE-U-CPP-SA**)** heuristic, Perturbation operator for (SA-Perturb) and developed the Long-Term Evolution Unlicensed Ray-Shooting (LTE-U-CPP-RS) heuristic algorithm (Zilberman, Haddad, Erlich, Peretz, & Dvir, 2021).

Vybornova (2020) presented a multi-critical method comparison of solutions applying various swarm optimisation algorithms, including financial and capital expenditures (CAPEX) and operating costs (OPEX) for network deployment and maintenance, respectively. In addition, PSO is used to solve the controller placements of CPP. The methods determine the approximate optimal number of controllers that are required for servicing an SDN and their locations and distribution of switches (Выборнова).

Meanwhile, services orchestration and data aggregation (SODA) framework is used by Liu, Yuxin, et al. (2019), wherein the network is split into three layers: vehicle network layer (VNL), middle routing layer (MRL), and finally the data centre layer (DCL).

To solve the position and number of controllers under dynamic traffic based on the generalisation of the K-centre algorithm and graph (G) theory, Ibrahim et al. **(**2020) proposed a heuristic multi-objective optimisation method using a Dynamic Capacitated Controller Placement (DCCPP) algorithm on a distributed 5G-CN Network Function Virtualisation (NFV)/SDN for 5G-CN. Meanwhile, dynamic assignment and Scheduling Algorithms (Assignment) and algorithm Rescheduling (Reassignment) of nodes are solved using a Greedy Random Search (GRS) algorithm and the K-centre problem to solve CCPP (Ibrahim et al., 2020).

B. Clustering Approach

Another approach to solve optimisation in CPP is the cluster approach. Chai et al. (2019) considered that the formulated specific optimisation may be a complex nonlinear number programming. Thus, an administrative entity (local and global) with a model (LME and GME) is used in this clustering approach. Control algorithms (link switching and matching capabilities) are also proposed. Dijkstra and K-mean delay algorithms are used to solve the associated sub-problem while the Kuhn–Munkres (K–M) algorithm is used to address the corresponding capacitance controller (Chai et al., 2019). In addition, Andishmand et al. (2020), presented a detailed study, summarised and classified the load balancing schemes in SDN. The proposed methods use genetic and Distributed Hopping (DHA) Algorithms to determine the benefits and disadvantages (Andishmand et al.). To prolong the lifetime of WSN, Mostafavi et al. (2020) applied the Approximate Rank–Order Wireless Sensor Networks (ARO-WSN) clustering algorithm (Mostafavi & Hakami, 2020).

Shahraki et al. (2020) focused on modern networking and computing models. For efficient processing in the SDN, the distributed structure control can be essential components for the decentralisation into cluster-based structures (Shahraki, Taherkordi, Haugen, & Eliassen, 2020).

C. Linear Programming and Mathematical Approach

Solutions and method improvements are also proposed using linear programming (LP) and mathematical approaches. From a mathematical paradigm, Ashrafi, Mohammad et al. (2020) developed LP to use switchable multiple controllers that can be mapped according to flexibility, latency and capability (Ashrafi, Faroq, & Correia, 2020). On the basis of dynamic location with integer linear programming (ILP), Toufga et al. (2020) proposed a method that adaptively alters the controller number and position according to traffic activity (Toufga et al., 2020).

Marques et al. (2019) introduced the In-band Network Telemetry Orchestration (INTO) issue and proposed the solution via an integer linear programming (ILP) paradigm (Marques, Luizelli, da Costa Filho, & Gaspary, 2019).

Other approaches include Best and Worst Multiple Criteria Decision Making (MCDMs) techniques, which are powerful mathematical tools used to solve complex issues related to different objectives, as used by Amiri, Esmaeil et al. (2020) (Amiri, Alizadeh, & Rezvani, 2020), and a mathematical model for single and multiple mapping that minimises metrics related to the deployment of controllers in SDN.

D. Tested Analysis Approach

Optimal objectives are also determined using the tested analysis approach. For locally hosted (baseline) and cloud-hosted SDN, Henriksson et al. (2019) considered experiments consisting of two network topologies that use Zodiac FX switches and Linux hosts for testing. The throughput, latency, jitter, packet loss, SDN Flow Table Update Rate (SDN-FTUR) and time to add new hosts are measured and results show a large fluctuation in throughput and packet loss (Henriksson & Magnusson, 2019). Meanwhile, Vestin and Jonathan (2020) investigated Control Monitoring Resiliency, Data Plane Programming (DPP) and NFV using the connection between SDN and NFV to improve the reliability, flexibility, and programmability of next-generation networks. SDN provides resiliency and traffic control to enhance connections in Split-MAC networks, especially under high network congestion. Meanwhile, packet delivery in both SDN-based industrial automation and 5G mm wave small cell backhaul networks are improved using fast failover combined with Bidirectional Forwarding Detection (BFD) (Vestin, 2020).

E. Queuing Analysis Approach

CPP is also solved using the queuing analysis approach. Abbasi, Aaqif Afzaal et al. (2019) carried out a survey on Software-Defined Cloud Computing (SDCC), which implements virtualisation administrations of all organised asset networks side-by-side in SDN and cloud computing. Challenges in usage and restrictions are also discussed. In addition, the potential of the SDCC approach are also explored in two areas, namely, application development and resource coordination (Abbasi et al., 2019).

F. Heuristic and Clustering Approach

Combined approaches are also used in literature. Heuristic and clustering approaches are used in Li, Guoyan et al. (2019) for a dynamic multi-console deployment schema that is dependent on load distribution, and suggested two major factors affecting the multi-controller deployment (OS3E) topology model. Based on (PSO), an Affinity Propagation (PSOAP) algorithm is used to improve the performance of static propagation multi-controller clustering. Then switches are reset in various subdomains when the network traffic changes dynamically, and Control–Domain Adjustment Algorithm (CDAA) of a depended on Breadth-First Search (BFS) is used with the dynamic traffic network. For multi-CPP solution to achieve controller load balancing, a heuristic algorithm is also used. PSOAP and Affinity Propagation (AP) can solve CPP without initialising the number of controllers, unlike the Genetic Algorithm (GA) (G. Li et al., 2019). Sminesh et al. (2020) proposed to partition a network segmentation. Furthermore, the controller mode strategy is used in a modified AP algorithm that consequently calculates the required number of clusters and uses specific SDN filter models (Sminesh, Kanaga, & Sreejish, 2020).

In large-Scale SDN, Li, Yi et al. (2020) used the Parameter Optimisation Model (POM) of Heuristic Algorithms to solve the CPP using synthetical-delay controller placement mode (SDCPM), PSO-parameter optimisation Algorithm (PSOPOA), FA-controller placement algorithm (FACPA), BA-controller placement algorithm (BACPA) and varna-based optimisation (VBO)-controller placement algorithm (VBOCPA). In synthetical-delay, VBO performs better than PSO, FA, Bat algorithm (BA) and TLBO (Y. Li, Guan, Zhang, & Sun, 2020). In addition, Torkamani-Azar et al. (2020) used Garter Snake Optimisation Capacitated CPP (GSOCCPP) as a meta-heuristic algorithm with new iterations and temperate mating conditions (Torkamani-Azar & Jahanshahi, 2020).

G. Heuristic and Linear Programming Approach

Other scholars combined the heuristic and linear programming approaches to solve CPP. Alowa and Abdunasser (2020) discussed the following four stages: 1) In SDN, CPP can be solved by exploiting the autonomous controlling group policy and is proposed in a new node degree-based algorithm, High Degree Independent Dominating Set (HDIDS); 2) The controller placement is enhanced using the range control network; 3) A band-control protection module and an execution arrange is planned by finding a group of optimal paths for the control canal beneath fail conditions; Finally, 4) A practical approach is presented to solve CPP. In the survivable support console mode, network throughput and performance is enhanced by Virtual Backup Domain (VBD), Full Enumeration (FE), used Controller Selection Max Degree with Short Distance (MDSD), Modified Density peaks clustering

(MDPC), Low Degree plus Short Distance (LDSD) and Inter Domain Adjacent Plus Short Distance (IDASD) (Alowa, 2020).

Maytree Ramasamy and Sanjay Pawar (2021) differentiated between single type and multiple consoles through DF multi-POX controllers, NOX, BEACON and FLOODLIGHT. Routing is implemented on both POX and NOX controllers. In addition, single type and multiple console targets are also analysed using Pareto-based Optimal Control Mode (POCO) tool for Internet Topology 2 and Routing by Network Emulator (NS2). Results lead to IMPSO, MO-ALO, DF and FFA. The comparison results improve the proposed system compared with the previous methods (Maytree Ramasamy, 2019).

Tubishat, Mohammad, et al. (2021) used SALP Swarm Algorithm (SSA) to solve the local optima problem and explore the balance of problem population diversity and its occurrence in local optima using the Dynamic Scalp Swarm Algorithm (DSSA). For feature selection (FS) problems, new Dynamic SSA (DSSA) is combined plus the K-nearest neighbour classifier (KNN) in the assembler mode. DSSA is proposed and is evaluated to outperform SSA. The results are compared with other known optimisation algorithms, including the original SSA, PSO, Grasshopper optimisation algorithm (GOA), GA and ant lion optimiser (ALO). The statistical analysis shows the accuracy results of DSSA (Tubishat et al., 2021).

Inspired by nature, Tahmasebi, Shirin, et al. (2021) used optimisation and population-based metaheuristic algorithms. CPP is the first formulated optimisation of the multi-objective issue by comparison with ILP. The Cuckoo SYNchronisation Controller Placement (SYNCOP) is used to solve the proposed ILP model using the algorithms Multi-Objective Genetic Algorithm (MOGA), Non-Dominated Sorting Genetic Algorithm (NSGA), Quantum Annealing (QA) and SA (Tahmasebi, Rasouli, Kashefi, Rezabeyk, & Faragardi, 2021).

In determining the best location for SDN controllers and optimisation model to solve the CPP, Similarly, Tahmasebi, Shirin, et al. (2020) proposed the Cuckoo Placement of Controllers (Cuckoo-PC), a meta-heuristic algorithm inspired by nature and achieves nearly similar results with ILP (Tahmasebi et al., 2020).

H. Heuristic, Clustering and Linear Programming Approach

Further combinations of the heuristic, clustering, and linear programming approaches are used to find suitable or improved CPP solutions. Based on SALP Swarm Optimisation Algorithm (SSOA), Ateya, Abdelhamied et al. (2019) proposed a dynamic optimised Chaotic SALP Swarm Algorithm (CSSA) that is created with the presentation of chaotic maps to improve the optimiser implementation. The optimal number of controllers and ideal associations between S2C in a large range are powerfully and dynamically assessed. In terms of reliability, quality and execution time in SDNs, different simulation results show that the proposed method outflanks metaheuristic and game theory algorithms (Ateya et al., 2019).

This combination is also used by Shetty, Vikas et al. (2018) to solve the K-Centres problem with the minimum cover technique (Shetty, Mukherjee, & Senthilkumar, 2018). In addition, Singh et al. (2019) indicated that these works solve CPP in trend classification.

I. Analysing Topological Mechanisms and Scalability Approach

The topological mechanisms and scalability approach are also used to find suitable CPP solutions. To obtain higher scalability, Abuarqoub and Abdelrahman (2020) applied Deep Deterministic Policy Gradient (DDPG) along with Deterministic Routing Optimisation Mechanism (DROM), which uses neural networks, the disconnect recovery in the SDN domain and mixed environment and machine learning (ML) systems (Abuarqoub, 2020). Košťál, Kristián et al. (2019) also analysed the performance and scalability issues of various SDN controllers and the topology mechanism approach (Košťál et al., 2019).

Ali, Jehad et al. (2020) analysed the proactive and reactive schemes in SDN using the OpenDayLight controller and Mininet (Ali et al., 2020).

J. Optimised K-Means

As a type of unsupervised learning, K-means clustering is used in the case of unlabelled data without specific groups or categories. This algorithm finds a dataset wherein the variable K represents the number of sets based on available features. Toufga, Soufian et al. (2020) considered the concept of subnetting as a network to solve CPP employing the optimised K-means algorithm. The assembly-based approach is modified for K to reduce latency between N2C and to decrease computational complexity (Toufga et al., 2020)

Comparing current cluster systems such as K-means and K-medians, Sminesh et al. (2020) found results in minimal mean-state, worst-case, and controller latency, and an improved controller imbalance factor that evidences the optimal number and position of SDN controllers (Sminesh et al., 2020).

The third section can be summarised in terms of the algorithm used, its pros and cons, and performance. Table 1 shows the results. Another useful summary can be based on a comparison of objectives regarding the controller placement in CPP solution, optimisation methods approaches, and the proposed model or tools used. Table 2 shows the results.

Table 1 Comparison of algorithms in terms of performance

Ref. No.	Algorithm	Pros and cons in terms of best results and performance
(Sahoo et al., 2017)	PSO and FFA	The results and performance show that FFA performs better than PSO and stochastic approach under different statuses to solve CPP.
(Ateya et al., 2019)	SSOA, SSA and CSSA for optimal number and switches	SSOA is an improvement version and presents better performance than CSSA in latency and cost-aware control. CSSA obtains the best performance in terms of average time.
(G. Li et al., 2019)	PSOAP, CDAA, BFS, AP and GA	PSOAP solves the problem of placing the cluster controller into the initial static state and reduces CDAA time of response by 50% on average compared with AP to solve the control field problem under dynamic traffic. Thus, CDAA outperforms GA by 25% on average. BFS obtains balancing better than AP and GA.
(Chai et al., 2019)	K–M, (controllers Dijkstra with an association and controller capacity matching)	The Dijkstra and K-means algorithms are used to solve delay and subproblems associated with the control switch. The K–M algorithm is used to solve the sub-problem compatible with the capacitance of the controller and to obtain the best position strategy.
(Alowa, 2020)	Controller Selection, MDSD, LDSD**,** IDASD and MDPC	Connected Dominating Sets (CDS) is a new technology that determines the number of controllers and placement within SDNs to reduce the lag time between C2S. VBD, FE, MDSD, MDPC, LDSD and IDASD algorithms are used to survive the standby controller mode approach that improves network performance and throughput.
(Zilberman et al., 2021)	HetNet-LTE-U-CPP-SA, SA-Perturb and (LTE-U-CPP-RS)	LTE-U is used to calculate the objective function of the optimal position of the control plane. Spatial throughput, probability of spatial correlation failure also calculates objective function by SA. Better than LTE-U-CPP-SA, an algorithm based on LTE-U-CPP-RS ray-shooting shows accurate simulation results.
(Выборнова)	PSO	PSO swarm intelligence algorithms and methods can determine an approximate optimal number of controllers required for servicing an SDN and their location and distribution of switches.
(Maytree Ramasamy, 2019)	FFA and DFA	The IPSO-DF algorithm can be compared with MALO-FF where single and multiple controllers can be analysed by the POCO tool, and routing is implemented by NS2. The performance of the proposed algorithm is good in reducing latency and processing CPP.
(Chen et al., 2018)	Genetic and DHA	DHA is better than schemes by reducing flow setup time and enhancing average CP distributed. A GA is used to determine the optimal values for different parameters (such as traffic, latency and distance) to reduce system cost.
(Y. Liu, Zeng, Liu, Zhu, & Bhuiyan, 2019)	Adjust routing path	The routing path algorithm in the MRL is adjusted to reduce the amount of redundant data representing new service of response time. The results show that the number of data collecting devices in the cloud edge based on SDN is 100 and the aggregation ratio is 0.4.
(Sminesh et al., 2020)	CP strategy based on modified AP and swarm intelligence methods	The algorithm determines the optimal number of controllers and their location better than K-means and K-medians, as the modified AP results in minimal latency between the controller while improving balance.
(Ibrahim et al., 2020)	GRS, DCCPP Scheduling and Reassignment	The results of the DCCPP indicate that the optimal number of controllers is allocated within an effective decentralisation policy and thus achieves a higher efficiency in resource allocation to form an ideal network. GRS is used for scheduling and switching assignments.
(Guillen, Takahira, Izumi, Abe, & Suganuma, 2020)	Assignment, Function to (Select and Calculate)	All algorithms' results demonstrate network survival and service continuity by ensuring connectivity to the controller, reducing the number of inoperable components and devices.
(Tubishat et al., 2021)	DSSA, PSO, GA, ALO, GOA and SSA	The results of the DSSA confirms the ability with better accuracy than the SSA algorithm in statistical analysis and are used to improve classification accuracy while choosing the minimum number of the most useful features.
(Tahmasebi et al., 2021)	SYNCOP, SA MOGA, NSGA and QA	SYNCOP determines the best location of controllers in SDN and increases the output and synchronisation cost of WSN performance, representing the average improvement of SYNchronisation CP versus QA.
(Shetty et al., 2018)	Dual-Tree graph with standard Approximate Greedy	Dual-Tree heuristic is used to solve K-Centre's problem in ILP and clustering approach.
(Huang, Chen, Fu, & Wen, 2019)	GA and Gradient Descent (GD)	A new algorithm combining the use of GA and GD is proposed for performance analysis of different network settings. The highest employment of control plane and competitively low response time is achieved compared with the widely-used heuristic methods.
(Singh, Kumar, & Srivastava, 2019)	PSO and TLBO	The results show that TLBO has better reliability than PSO in publicly available topologies to solve CPP.

Table 2 Literature comparison in solving the controller placement and optimisation

Authors/Ref.	Objective	Method	Tool used for improvement	Target parameters
Sahoo, Kshira Sagar, et al. (Sahoo et al., 2017)	Finds the optimal number and position of controllers; reduces the latency of access from S2C	Heuristic approach	CPP- SDN-based WAN architecture	Location of controllers and distribution of switches
Ateya, Abdelhamied A., et al. (Ateya et al., 2019)	Capacity and incapacity of latency and performance	Heuristic, Clustering and Linear approach	OpenFlow	Location of controllers and distribution of switches
Abuarqoub and Abdelrahman (Abuarqoub, 2020)	To performance and solve scalability; Parallelism (routing-scheme and machine learning) optimisation	Analysing Topological Mechanisms and Scalability Approaches	Parallelism scheme machine learning (DDPG) and (DROM)	Location of controllers and distribution of switches
Li, Guoyan, Xinqiang Wang, and Zhigang Zhang (G. Li et al., 2019)	Improve the scalability, reliability with the Load Balancing Scheme of the control plane	Heuristic and Clustering approach	(OS3E)	Location of controllers and distribution of switches
Chai, Rong, et al. (Chai et al., 2019)	To provide flexibility and the ability to solve CP capacity and delay	Clustering approach	(LME) and (GME) model	Location of controllers and distribution of switches
Ashrafi, Mohammad, et al. (Ashrafi et al., 2020)	Solve the CP under multi-controller S2C assignment by the switch assigned to multiple controllers; With resiliency, scalability, and inter-plane latency	Linear programming	mathematical model	Location of controllers and distribution of switches
Alowa and Abdunasser (Alowa, 2020)	To improve the productivity and performance of the network and selecting the available controller with associated	Heuristic and Linear programming	HDIDS technique and VBD	Location of controllers and distribution of switches
Zilberman, Aviram, et al. (Zilberman et al., 2021)	To design an effective wireless control offload data traffic via Wi-Fi networks	Heuristic approach	WCPP-4G LTE-U	Location of controllers
Vybornova A. (Выборнова)	To solve the CPP for the multi-controller using swarm intelligence	Heuristic approach	CAPEX, OPEX	Location of controllers and distribution of switches
Mythrayee Ramasamy and Sanjay Pawar (Maytree Ramasamy, 2019)	Monitor and enhance the network performance	Heuristic and Linear programming	POCO and NS2	Location of controllers and distribution of switches
Toufga, Soufian, et al. (Toufga et al., 2020)	Bring flexibility and efficient use of resources to vehicle networks	Linear (theoretical) programming	SDVN and ILP	Location of controllers
Andishmand, Rahmatollah, Mostafavi, et al. (Andishmand et al.)	Classify the load balancing schemes	Clustering approach	Load balancing schemes	Location of controllers and distribution of switches

Liu, Yuxin, et al. (Y. Liu et al., 2019)	For load balancing, low response delay, reduce service and data redundancy	Heuristic approach	SODA the scheme, DCL, MRL and VNL	Location of controllers and distribution of switches
Sminesh, C. N., et al. (Sminesh et al., 2020)	Latency and improved controller imbalance factor and partition (Load balancing)	Heuristic and clustering approach	GARR, GEANT and SwitchL	Location of controllers and distribution of switches
Ibrahim, Abeer AZ, et al.(Ibrahim et al., 2020)	Develop a resource management allocation in multi-Control 5G	Heuristic approach	SDN/NFV for 5G-CN	Location of controllers and distribution of switches
Singh, Ashutosh Kumar, et al. (Singh, Maurya, & Srivastava, 2020)	Capacity and incapacity of latency	Heuristic approach	OS3E	Location of controllers
Tubishat, Mohammad, et al. (Tubishat et al., 2021)	To improve diversity and balance between exploration and exploitation and avoid falling in local optima, classification accuracy and feature selection	Heuristic and Linear Programming	KNN classifier and FS methods	Location of controllers
Tahmasebi, Shirin, et al.(Tahmasebi et al., 2021)	Optimising Cost and Network Performance in WSNs with increased scalability	Heuristic and Linear Programming	CPLEX ILP Solver	Location of controllers
Abbasi, Aaqif Afzaal, et al. (Abbasi et al., 2019)	Programmability, scalability, interoperability and security for SDCC	Queuing analysis approach	SDCC	Location of controllers and distribution of switches
Henriksson, Johannes, and Alexander Magnusson (Henriksson & Magnusson, 2019)	To evaluate the performance, including latency, jitter, packet loss, throughput and SDN-FTUR	Tested analysis approach	Baseline- and cloud-hosted SDN, the topology used Zodiac FX switches and Linux hosts	Location of controllers and distribution of switches
Vestin and Jonathan (Vestin, 2020)	For increased network flexibility and monitoring capacity	Tested analysis approach	NFV of DPP, MAC WLANs, 5G	Location of controllers and distribution of switches
Tanha and Maryam (Tanha, 2019)	To solve the resilient CPP with resilient switch reassignment problem and incremental CPP	Heuristic and Mathematical approach	Software-Defined Wide Area Networks (SD-WANs) model	Location of controllers and distribution of switches
Li, Yi, et al. (Y. Li et al., 2020)	To verify the effectiveness of CPP reduce the delay between S2C and C2C	Heuristic and Clustering approach	Parameter Optimisation Model (POM)	Location of controllers and distribution of switches
Torkamani-Azar et al. (Torkamani-Azar & Jahanshahi, 2020)	To obtain the lowest execution time among the analysis algorithms and the least memory consumption	Meta-heuristic and Clustering approach	GSOCCPP	Location of controllers and distribution of switches

Conclusion and Future Work

A highly important and essential aspect of SDN scaling and management is CPP, wherein active controller placement aims to improve scalability and performance metrics such as latency, reliability, load distribution, failure resilience and cost. In this case, the quantitative the qualitative views of SDN are necessary. This study presents a brief introduction of SDN, the shortcomings of a single controller, and the evolution of a multiple (distributed) controller and its architecture. The results reveal that multiple controllers are required for a large-scale SDN to ensure scalability and reliability. Constructive comparisons of the proposed solutions for CPP can advance the research efforts in this direction. The findings of the present study can help obtain more efficient and reliable solutions for CPP in SDN. Although future research effort is necessary, especially for real-life large-scale SDN scenarios that require multi-objective optimisations. Therefore, various criteria of different types of methods can be used to select algorithms for the purpose of optimisation. In addition, a multi-objective comparison table literature of optimisation methods used to solve CPP is presented.

SDN has important advantages and characteristics in terms of deployment and applications, which makes CPP an important point for continuous active research. The present study focuses on performance measures and several other goals. All of these challenges in controller placement can be addressed in by future processors. Further study is necessary for this topic to identify various issues and future research directions. Despite the series of solutions proposed for CPP, major issues remain and may be resolved in the future. Priorities and directions can be subsequently discussed.

References

Abbasi, A.A., Abbasi, A., Shamshirband, S., Chronopoulos, A.T., Persico, V., & Pescapè, A. (2019). Software-defined cloud computing: A systematic review on latest trends and developments. *IEEE Access, 7,* 93294-93314.

Abuarqoub, A. (2020). A review of the control plane scalability approaches in software defined networking. *Future Internet, 12*(3), 49.

Ali, J., Lee, G.M., Roh, B.-h., Ryu, D.K., & Park, G. (2020). Software-defined networking approaches for link failure recovery: A survey. *Sustainability, 12*(10), 4255.

Alowa, A. (2020). *Scalable Reliable Controller Placementin Software Defined Networking.* Concordia University,

Amiri, E., Alizadeh, E., & Rezvani, M.H. (2020). Controller selection in software defined networks using best-worst multi-criteria decision-making. *Bulletin of Electrical Engineering and Informatics, 9*(4), 1506-1517.

Andishmand, R., Mohammdi, H., & Mostafavi, S. Load Balancing In Multi-Controller Software-Defined Networks.

Ashrafi, M., Faroq, A.T., & Correia, N. (2020). Placement of Controllers in Software Defined Networking under Multiple Controller Mapping. *KnE Engineering,* 394–404-394–404.

Ateya, A.A., Muthanna, A., Vybornova, A., Algarni, A.D., Abuarqoub, A., Koucheryavy, Y., & Koucheryavy, A. (2019). Chaotic salp swarm algorithm for SDN multi-controller networks. *Engineering Science and Technology, an International Journal, 22*(4), 1001-1012.

Chai, R., Yuan, Q., Zhu, L., & Chen, Q. (2019). Control plane delay minimization-based capacitated controller placement algorithm for SDN. *EURASIP Journal on Wireless Communications and Networking, 2019*(1), 1-17.

Chen, Y., Wen, X., Leng, X., Yang, B., Li, L., Zheng, P., & Hu, C. (2018). Optimization framework for minimizing rule update latency in SDN switches. *ZTE Communications, 16*(4), 15-29.

Guillen, L., Takahira, H., Izumi, S., Abe, T., & Suganuma, T. (2020). On Designing a Resilient SDN C/M-Plane for Multi-Controller Failure in Disaster Situations. *IEEE Access, 8,* 141719-141732. http://doi.org/10.1109/access.2020.3013323

Henriksson, J., & Magnusson, A. (2019). Impact of using cloud-based SDN controllers on the network performance. In.

Hu, T., Guo, Z., Yi, P., Baker, T., & Lan, J. (2018). Multi-controller based software-defined networking: A survey. *IEEE Access, 6,* 15980-15996.

Huang, V., Chen, G., Fu, Q., & Wen, E. (2019). *Optimizing controller placement for software-defined networks.* Paper presented at the 2019 IFIP/IEEE Symposium on Integrated Network and Service Management (IM).

Ibrahim, A.A., Hashim, F., Noordin, N.K., Sali, A., Navaie, K., & Fadul, S.M. (2020). Heuristic resource allocation algorithm for controller placement in multi-control 5g based on sdn/nfv architecture. *IEEE Access, 9,* 2602-2617.

Isong, B., Molose, R.R.S., Abu-Mahfouz, A.M., & Dladlu, N. (2020). Comprehensive review of SDN controller placement strategies. *IEEE Access, 8,* 170070-170092.

Košťál, K., Bencel, R., Ries, M., Trúchly, P., & Kotuliak, I. (2019). High performance SDN WLAN architecture. *Sensors, 19*(8), 1880.

Li, G., Wang, X., & Zhang, Z. (2019). SDN-Based Load Balancing Scheme for Multi-Controller Deployment. *IEEE Access, 7,* 39612-39622. http://doi.org/10.1109/access.2019.2906683

Li, Y., Guan, S., Zhang, C., & Sun, W. (2020). Parameter Optimization Model of Heuristic Algorithms for Controller Placement Problem in Large-Scale SDN. *IEEE Access, 8,* 151668-151680. http://doi.org/10.1109/access.2020.3017673

Liu, S., Steinert, R., & Kostic, D. (2018). *Flexible distributed control plane deployment.* Paper presented at the NOMS 2018-2018 IEEE/IFIP Network Operations and Management Symposium.

Liu, Y., Zeng, Z., Liu, X., Zhu, X., & Bhuiyan, M.Z.A. (2019). A novel load balancing and low response delay framework for edge-cloud network based on SDN. *IEEE Internet of Things Journal, 7*(7), 5922-5933.

Lu, J., Zhang, Z., Hu, T., Yi, P., & Lan, J. (2019). A Survey of Controller Placement Problem in Software-Defined Networking. *IEEE Access, 7*, 24290-24307. http://doi.org/10.1109/access.2019.2893283

Marques, J.A., Luizelli, M.C., da Costa Filho, R.I.T., & Gaspary, L.P. (2019). An optimization-based approach for efficient network monitoring using in-band network telemetry. *Journal of Internet Services and Applications, 10*(1), 1-20.

Maytree Ramasamy, S. P. (2019). Single and Multi-Type Controllers with Soft Computing Methods and Routing in Software-Defined Network. *International Journal of Engineering and Advanced Technology, 9*(1).

Mostafavi, S., & Hakami, V. (2020). A new rank-order clustering algorithm for prolonging the lifetime of wireless sensor networks. *International Journal of Communication Systems, 33*(7).

Sahoo, K.S., Sarkar, A., Mishra, S K., Sahoo, B., Puthal, D., Obaidat, M.S., & Sadun, B. (2017). Metaheuristic solutions for solving controller placement problem in SDN-based WAN architecture. *Paper presented at the ICETE 2017-Proceedings of the 14th International Joint Conference on e-Business and Telecommunications.*

Shahraki, A., Taherkordi, A., Haugen, Ø., & Eliassen, F. (2020). A survey and future directions on clustering: From WSNs to IoT and modern networking paradigms. *IEEE Transactions on Network and Service Management, 18*(2), 2242-2274.

Shetty, V. S., Mukherjee, A., & Senthilkumar, K. (2018). Examining Heuristics for the K-Centers Problem.

Singh, A. K., Kumar, N., & Srivastava, S. (2019). PSO and TLBO based reliable placement of controllers in SDN. *International Journal of Computer Network and Information Security, 11*(2), 36-42.

Singh, A. K., Maurya, S., & Srivastava, S. (2020). Varna-based optimization: a novel method for capacitated controller placement problem in SDN. *Frontiers of Computer Science, 14*(3), 1-26.

Sminesh, C., Kanaga, E. G. M., & Sreejish, A. (2020). A multi-controller placement strategy in software defined networks using affinity propagation. *International Journal of Internet Technology and Secured Transactions, 10*(1-2), 229-253.

Tahmasebi, S., Rasouli, N., Kashefi, A. H., Rezabeyk, E., & Faragardi, H.R. (2021). SYNCOP: An evolutionary multi-objective placement of SDN controllers for optimizing cost and network performance in WSNs. *Computer Networks, 185*, 107727.

Tahmasebi, S., Safi, M., Zolfi, S., Maghsoudi, M.R., Faragardi, H. R., & Fotouhi, H. (2020). Cuckoo-PC: an evolutionary synchronization-aware placement of SDN controllers for optimizing the network performance in WSNs. *Sensors, 20*(11), 3231.

Tanha, M. (2019). *Resilient controller placement problems in software defined wide-area networks.*

Torkamani-Azar, S., & Jahanshahi, M. (2020). A new GSO based method for SDN controller placement. *Computer Communications, 163*, 91-108. doi:10.1016/j.comcom.2020.09.004

Toufga, S., Abdellatif, S., Assouane, H.T., Owezarski, P., & Villemur, T. (2020). Towards dynamic controller placement in software defined vehicular networks. *Sensors, 20*(6).

Tubishat, M., Ja'afar, S., Alswaitti, M., Mirjalili, S., Idris, N., Ismail, M. A., & Omar, M. S. (2021). Dynamic salp swarm algorithm for feature selection. *Expert Systems with Applications, 164*, 113873.

Vestin, J. (2020). *SDN-Enabled Resiliency, Monitoring and Control in Computer Networks.* Karlstads universitet.

Xie, J., Guo, D., Qian, C., Liu, L., Ren, B., & Chen, H. (2019). Validation of distributed SDN control plane under uncertain failures. *IEEE/ACM Transactions on Networking, 27*(3), 1234-1247.

Zilberman, A., Haddad, Y., Erlich, S., Peretz, Y., & Dvir, A. (2021). SDN Wireless Controller Placement Problem-The 4G LTE-U Case. *IEEE Access, 9*, 16225-16238. http://doi.org/10.1109/access.2021.3052892

Vybornova, A. *A Review of Methods for using Swimming Intelligence Algorithms for Construction and Optimization of Software-configured Networks.*