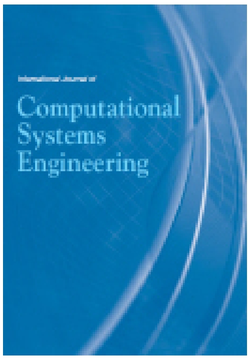


International Journal of Computational Systems Engineering

ISSN online: 2046-3405 - ISSN print: 2046-3391
https://www.inderscience.com/ijcsyse

Research on end-to-end computing power network architecture based on SDN and MIH technology

Fang Cui, Jianhua Gu, Hengjiang Wang, Mao Ni, Ting Zhou

DOI: 10.1504/IJCSYSE.2025.10059362

Article History:

Received:	14 June 2023
Last revised:	07 August 2023
Accepted:	10 August 2023
Published online:	13 December 2024

Research on end-to-end computing power network architecture based on SDN and MIH technology

Fang Cui, Jianhua Gu, Hengjiang Wang, Mao Ni* and Ting Zhou

China Mobile Group Device Co., Ltd.,
Beijing, 100053, China
Email: cuifang@cmdc.chinamobile.com
Email: gujianhua@cmdc.chinamobile.com
Email: wanghengjiang@cmdc.chinamobile.com
Email: mao_ni_mn@163.com
Email: zhouting@cmdc.chinamobile.com
*Corresponding author

Abstract: For ubiquitous access, the study selects software-defined networks with independent switching, control upgrades, and centralised management using the manager to achieve the same control. Pre-filtering and D-VIKOR are used in network switching decisions to help perform network decisions. The experimental results show that the network architecture is suitable for mobile network scenarios. In the 30-m/s in-vehicle scenario, the latency is reduced by 27.55%, and the overall switching effectiveness is improved by 10.21%. For the situation that the increase in arithmetic power will lead to more nodes, which will increase switching errors and blocking, the relevant test results show that the number of vertical switching errors for the SDN architecture is less than that of the D-TOPSIS architecture 285 times less. This indicates that the end-side network architecture constructed in the study can handle the switching requirements of heterogeneous networks and also shows excellent performance on different problems in multiple scenarios.

Keywords: count-network convergence: end-side network; architecture; switching delay; mobile scenario.

Reference to this paper should be made as follows: Cui, F., Gu, J., Wang, H., Ni, M. and Zhou, T. (2024) 'Research on end-to-end computing power network architecture based on SDN and MIH technology', *Int. J. Computational Systems Engineering*, Vol. 8, No. 7, pp.1–13.

Biographical notes: Fang Cui obtained her Master's in Communication and Information System from the Institute of Telecommunication Science and Technology, MIIT in 2004. She currently serves as the General Manager and Senior Engineer in the Technology Department of China Mobile Device Company. She has been engaged in the work of mobile terminal chips, terminal technology standards, and terminal industry promotion. She has presided over and participated in many major projects related to the terminal field under MIIT and NDRC. She has published five papers, authorised over ten patents, and led teams to develop the XR device industry standard. Her areas of interest include 5G terminals, eSIM, multimedia processing, cloud terminals, computing power networks, quantum computing, XR, etc.

Jianhua Gu obtained his Master's in Microelectronic Application Technology from Hefei University of Technology. He is currently the General Manager at China Mobile Device Company, responsible for technological innovation, technical research, terminal software/hardware R&D in the terminal field. His areas of interest include, terminal software/hardware development, computing force network, artificial intelligence, etc.

Hengjiang Wang obtained his Master's in Computational Engineering from Tokyo Institute of Technology in 2005. He is currently the Deputy General Manager and Senior Engineer at China Mobile Device Company, responsible for technological innovation, technical research, terminal software/hardware R&D in the terminal field. Meanwhile, he serves as a member of MIIT Economic Expert Committee, a member of CAICT Technical Committee, and Vice Chairman of TAF. He has presided over terminal-related tasks in many major projects under the Ministry of Science and Technology, MIIT, and NDRC. He has published five papers, authorised over 30 patents, led the development of 12 industry standards and 35 enterprise standards, as well as 44 software copyrights. His areas of interest include end-to-end communication protocols for network terminals, multimedia processing, terminal software/hardware development, distributed operating systems, cloud terminals, computing force network, artificial intelligence, etc.

Mao Ni obtained his Master's in Computer Technology from Beijing University of Posts and Telecommunications in 2012. He is currently responsible for product development at China Mobile Communications Group Terminal Co., Ltd. He has led multiple major research and development projects for China mobile and participated in national major projects related to computing force network. He has authored more than ten patents, and his research interests include computing force network, 5G terminal technology, operating systems, and artificial intelligence.

Ting Zhou obtained a Master's in Information and Communication Engineering from Beijing University of Posts and Telecommunications in 2014. She currently works at China Mobile Communications Group Terminal Co., Ltd. engaged in long-term research on mobile terminal product planning, terminal communication technology, and computing force network technology. She led multiple major projects of China mobile in computing force network, participated in national major projects related to terminal-side computing force network. She has published five papers, and wrote over ten patents. Her areas of interest include computing force network, 5G terminal technology, etc.

1 Introduction

With the gradual maturation of 5G technology, new infrastructure projects built digitally penetrate various industries, and cloud-network convergence will face a double test of computing power and network (Constantinides et al., 2018). With access to terminals such as autonomous driving and IoT, deep learning and other mathematical models can improve data processing in heterogeneous networks. An excellent terminal network-side architecture can adjust the network optimum from the overall mechanism, thus avoiding the situation of wasted computing power. Hence, the network system from the end-side architecture becomes one of the main directions of computing network convergence optimisation (Chen et al., 2020). Analysis of the current terminal device interventions reveals that operators are equipped with a variety of network interventions, such as universal packet radio services and wireless network technologies, so a generic connection of these technologies can enable vertical switching of mobile nodes and thus increase the end-side network experience (Siriwardhana et al., 2021). Because of this, the services at the end layer need to be micro-serviced in the new arithmetic network architecture to reach the scheduling and ubiquitous coexistence of communication modules and computing power. To soften the switching of heterogeneous networks, the network is sliced into data and control layers, and the layered network can be centrally controlled and operable using software-defined techniques (Raja et al., 2020). The new trend of network convergence requires a network architecture with flexible scheduling capabilities and transmission timeliness. With the support of the upper layer services, the terminal side will use software-defined networks as the core of the optimisation of the architecture. In this process, research will optimise the traditional VIKOR algorithm to reduce the complex impact of the time element. SDN technology will also be used to separate the data in the plane to match the work task of independent media switching, providing optimisation exploration for the ubiquitous access service of network convergence.

The comparison between the research results and the latest paper is as follows: in order to solve the seamless switching problem in heterogeneous networks, the IEEE 802.21 working group proposed a grafting independent switching framework. The purpose of this standard is to separate the details of different MAC layer technologies and provide necessary services for the upper layer to promote switching between heterogeneous networks. When the network Selection algorithm is not defined in this method, it needs to be developed based on the MIH function. In addition, in order to make the switching process more effective, it is necessary to pre configure global control and management of network entities before switching is executed. There are still areas to be optimised for seamless switching and network decision-making mechanisms in current heterogeneous networks. To address this issue, a new end-to-end computing power network architecture based on SDN and MIH technology has been studied and constructed. Compared to the research results of the latest existing papers, the contributions of this study are as follows:

1 A novel optimised vertical switching framework was proposed, which integrates SDN technology and MIH technology, and is redesigned to customise the signalling of the switching process for the proposed framework, achieving seamless link recommendations and agile network path reconfiguration.

2 The research adds a pre filtering mechanism before the Selection algorithm of the confidence network, which uses the security policy matching technology to eliminate the network with incompatible security capabilities in advance.

3 Research and propose an optimised network Selection algorithm D-VIKOR, which considers multiple attributes during network selection. D-VIKOR often evaluates dynamic attributes, not static attributes, and can reduce time complexity compared with standard VIKOR.

2 Related work

Computational network technology has received much attention as the basis for a high degree of convergence of arithmetic and network, which has been explored by major operators and scholars, which is particularly important for subsequent application push. scholars such as Nawaz et al. (2019) believe that only a small part of 5G network can provide management support for innovative services, so they use the relevant content for the pre-design of 6G network, in the assumption of network architecture, the new network services cope with high precision domain with high precision real time corresponding state, based on this uncertainty machine learning and quantum computing will be used as 6G technology core, for this they developed related technologies in 5G network to provide auxiliary architecture for new network. Qiu et al.'s (2019) experimental team believes that UAVs will be auxiliary access devices for 5G, so they used software custom network for UAV network for Kato N's team believes that the development of new cellular communication technologies in the context of maturing 5G technology can enable critical upgrades, in which artificial intelligence will be emphasised in the development of new technologies, and therefore machine learning in the future 6G architecture technology will face key ten challenges. Their study noted the contribution of machine learning to arithmetic power, which is enlightening for end-side network construction (Kato et al., 2020). Vehicle communication is one of the applications of arithmetic network convergence, Zhang et al. (2018a, 2018b) experimental team considers advanced sensing and control devices as the foundation of intelligent transportation systems, on which vehicle communication networks are tasked to connect devices, despite the fruitful VCN research, there is still room for progress in sensing optimised control in autonomous driving, and their case study highlights the need for real-world scenarios (Zhang et al., 2018b). In the management of heterogeneous devices IoT and arithmetic network systems face the same problem, Tseng et al. (2020) argue that blockchain enables decentralised coordination to solve the heterogeneous problem of IoT, for which they outline an architecture to manage large-scale IoT, and this decentralised structure has enlightening implications for the cooperation of arithmetic networks across disciplinary domains. Arithmetic power enhancement is one of the goals of arithmetic network convergence, Fang et al. (2018) argue that non-orthogonal multiple access technology is the key to 5G networks, and in heterogeneous environments with enhanced arithmetic power demand, non-orthogonal multiple access technology can allocate channels to the optimal and thus improve efficiency, considering the interference of the same channel and cross-layer interference, they transform the configuration problem and propose algorithms to solve the allocation of macro-cells to good sub-cells, and their results prove that their scheme is shown to improve the system performance (Fang et al., 2018).

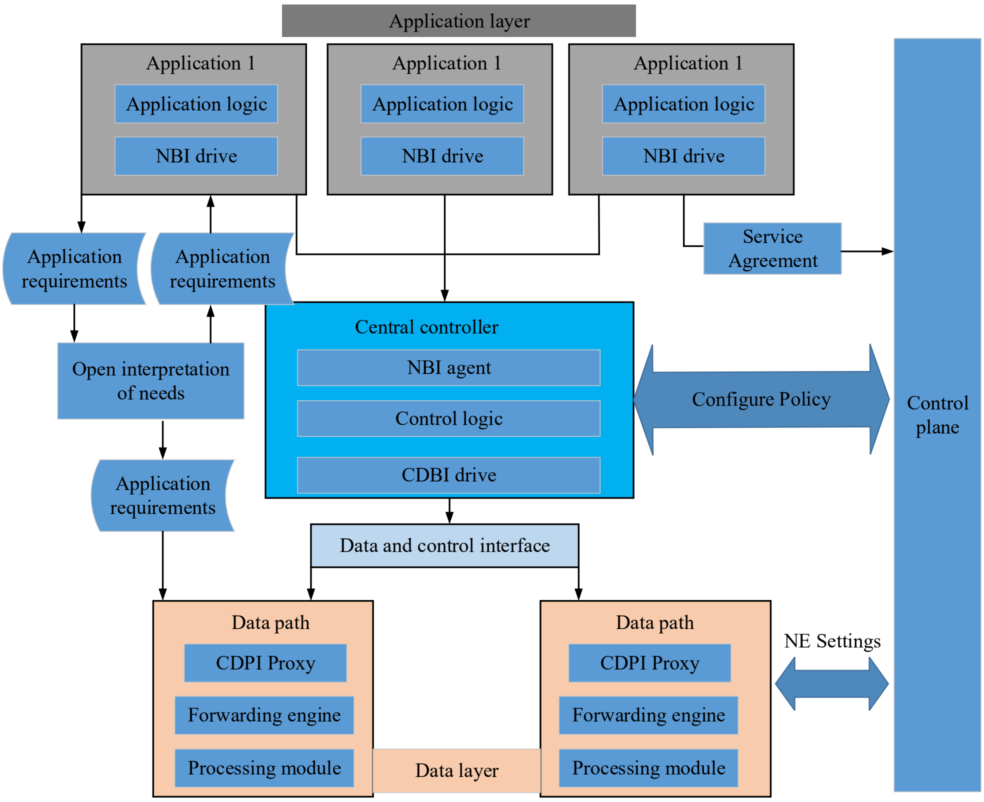
Vertical switching at the end layer of the network can be achieved by technologies such as software-defined networks, and scholars have made some progress in exploring the technology and architecture of end-side networks. Sahoo et al. (2020) believe that software-defined networks are good architectures that can provide hardware control for operators, but controllers can have potential network threats, so they perform traffic monitoring for DDoS attacks, and the implementation path is through component analysis with genetic algorithms The combination to improve the monitoring accuracy and also introduce the kernel function into the noise reduction, and the results show that the model has good performance. On the basis of arithmetic power enhancement, traffic prediction has planning significance, Wang et al. (2021) proposed a lightweight traffic prediction algorithm to perform traffic statistics on the basis of SDN traffic forwarding, the method reduces the statistical cost while obtaining time series, the test results prove the effectiveness of the method, and their research provides demanded preparation for arithmetic power enhancement. Badotra and Panda's (2020) research focused on the exploration of operating system performance for SDN, they argued that the advantage of SDN is open network operation, so finding a better controller helps in operation improvement, and the performance comparison results of ODL and ONOS showed that the former outperformed the latter in terms of burst rate throughput, etc. Wang et al. (2020) argued that the performance of DPI-based flow statistics could not meet expectations, so they propose a spatio-temporal co-sampling framework for flow-aware SDNs that characterises the sampling accuracy of switches in the spatio-temporal dimension, propose optimisations for the inherent problems of the framework, and address top-K switches and local maximum sampling time slot allocation, and their research breaks through the sampling framework for SDNs. With the increase in arithmetic power, routing protocols need to be upgraded to adapt to traffic changes, Casas-Velasco et al. (2020) introduced a new SDN routing approach, the implementation path is to add reinforcement learning algorithms to SDN to make decisions on information about the link state, experimental results prove the approach as an optimal solution for intelligent routing.

Computing network convergence on computing and network architecture and other technologies put forward new needs, especially in specific scenarios, the need for hardware and requirements based on the consideration of scholars on the control side of the research has been somewhat effective, while the end-side network architecture and arithmetic research is relatively lacking, especially in the seamless switching and softening of heterogeneous networks there is a lack of processing, at this time from the software-defined network level to build a network system to help the overall architecture. The content of all literature with different parameters related to the work is shown in Table 1.

Table 1 Literature table for different parameters of related work

Types of work involved	*Author*	*A particular year*	*Journal name*	*Literature title*
Network computing technology	Nawaz et al.	2019	*IEEE Access*	Quantum machine learning for 6G communication networks: state-of-the-art and vision for the future.
	Qiu et al.	2019	*IEEE Wireless Communications*	Air-ground heterogeneous networks for 5G and beyond via integrating high and low altitude platforms.
	Kato et al.	2020	*IEEE Wireless Communications*	Ten challenges in advancing machine learning technologies toward 6G.
	Zhang et al.	2018b	*IEEE Communications Magazine*	Vehicular communication networks in the automated driving era.
	Tseng et al.	2020	*IEEE Network*	Blockchain for managing heterogeneous internet of things: a perspective architecture.
	Fang et al.	2018	*IEEE Transactions on Vehicular Technology*	Joint energy efficient subchannel and power optimization for a downlink NOMA heterogeneous network.
Software Defined Network	Sahoo et al.	2020	*IEEE Access*	An evolutionary SVM model for DDOS attack detection in software defined networks.
	Wang et al.	2021	*Mobile Networks and Applications*	A new traffic prediction algorithm to software defined networking.
	Badotra and Panda	2020	*Cluster Computing*	Evaluation and comparison of OpenDayLight and open networking operating system in software-defined networking.
	Wang et al.	2020	*IEEE Journal on Selected Areas in Communications*	STCS: Spatial-temporal collaborative sampling in flow-aware software defined networks.
	Casas-Velasco et al.	2020	*IEEE Transactions on Network and Service Management*	Intelligent routing based on reinforcement learning for software-defined networking.
	Xue et al.	2020	*Optics Express*	SDN enabled flexible optical data center network with dynamic bandwidth allocation based on photonic integrated wavelength selective switch.
	Lin et al.	2021	*IEEE Journal of Solid-State Circuits*	ADC-DSP-based 10-to-112-Gb/s multi-standard receiver in 7-nm FinFET.
	Ikpehai et al.	2018	*IEEE Internet of Things Journal*	Low-power wide area network technologies for internet-of-things: a comparative review.
5G technology	Siriwardhana et al.	2021	*Progress in Electromagnetics Research M*	Compact, broadband and reliable lateral MEMS switching networks for 5G communications.
	Zhang et al.	2018a	*Mobile Networks and Applications*	5G technologies for future wireless networks.
Digital communication	Constantinides et al.	2018	*Information Systems Research*	Introduction – platforms and infrastructures in the digital age.
	Chen et al.	2020	*IEEE Communications Surveys & Tutorials*	Caching in vehicular named data networking: architecture, schemes and future directions.
Network switching	Raja et al.	2020	*IEEE Internet of Things Journal*	Intelligent reward-based data offloading in next-generation vehicular networks.
	Zhang et al.	2018a	*Mobile Networks and Applications*	5G technologies for future wireless networks.
	Gao et al.	2018	*IEEE Transactions on Communications*	Sum rate optimization of multi-standard IEEE 802.11 WLANs

Figure 1 Terminal side network architecture (see online version for colours)



3 SDN-based end-side network system and heterogeneous switching

3.1 End-side-based network architecture

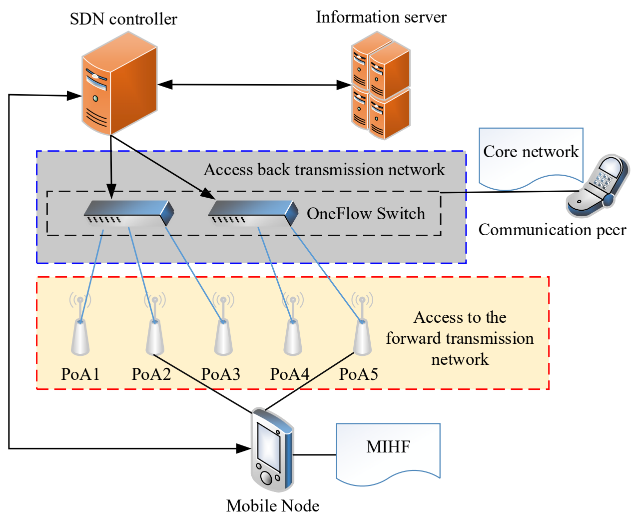
With the overall idea of computational network convergence, the network architecture needs enough operable space to accommodate the user side as well as the demand for arithmetic enhancement. SDN, as a new generalised architecture, is able to communicate at the interface between the control and data planes, and the advantage of this architecture is that the data and control layers can operate independently (Xue et al., 2020). In addition, the open architecture makes it easier to program network innovations, while centralised management provides a platform for a global view of the network. The SDN architecture has a centralised control plane and a distributed forwarding plane, and this generalised architecture is shown in Figure 1.

The architecture as in Figure 1 consists of four defined planes, where the application plane is the open programmable interface where users can control the network resources. The interface protocols of the data plane and control plane use the mainstream protocols of the southbound interface; the northbound protocols responsible for communication functions can be developed according to the end-side user requirements. The centralised controller will collect the real-time status of the network and pass it to the application layer, at which time the application layer's program will be transformed into a grassroots command to notify the bottom-level devices. The standardised southbound interface protocol makes the network equipment free from the control of the vendor's switching equipment, thus realising the autonomous control of the panel. The specific steps of Figure 1 are as follows.

The SDN architecture is mainly divided into application layer, controller layer, and infrastructure layer. Among them, the application layer focuses on the development of network business logic and is responsible for resource allocation; the controller layer manages the global network; the infrastructure layer is responsible for forwarding data to various network devices. From the perspective of the controller layer, the interface with the application layer is defined as the north bound interface (NBI), and the interface with the infrastructure is defined as the south bound interface (SBI). By encapsulating the northbound interface, the application layer calls various network resources and controls the resource status of the entire network in the form of software programming, and uniformly schedules resources. Ideally, the application layer encapsulates all 'how' operations and hides network related technical information from users. When the upper layer application calls the application layer's services, it only needs to describe what it wants. However, there is currently a lack of industry recognised standards for northbound interfaces. The main reason is that the northbound interface directly serves business applications, and its design needs to be closely related to business application requirements, with diverse characteristics that are difficult to unify. The southbound interface protocol in this SDN architecture is OpenFlow, which is used for communication between controllers and switches. The controller can control the switch through the flow table issued by OpenFlow, and the switch can also provide feedback to the controller. At the same time, OpenFlow also specifies the forwarding method for messages by the switch.

In the practice of counting network construction, the construction of network equipment will face the problem of multi-access coordination, but the existing hardware does not have the function of seamless switching, so the independent switching of media is required. An IEEE802.21 has been proposed, which is a standard that enables seamless switching of different networks, so the end-side network framework is based on it (Zhang et al., 2018a, 2018b). In a media independent switching-based network framework, the SDN controller separated radio access network will be optimised and the switching of radio links can be changed with the movement of users in a software defined environment. Medium-independent services provide the mechanism for such a framework, where logical entities can discover the intrinsic properties of the wireless access network and facilitate network switching based on the information. The media-independent switching framework for software-defined wireless access networks can be controlled by a single or multiple network operators, and the scenario of this framework entity is built as shown in Figure 2.

Figure 2 Independent switching of media in real scenarios (see online version for colours)



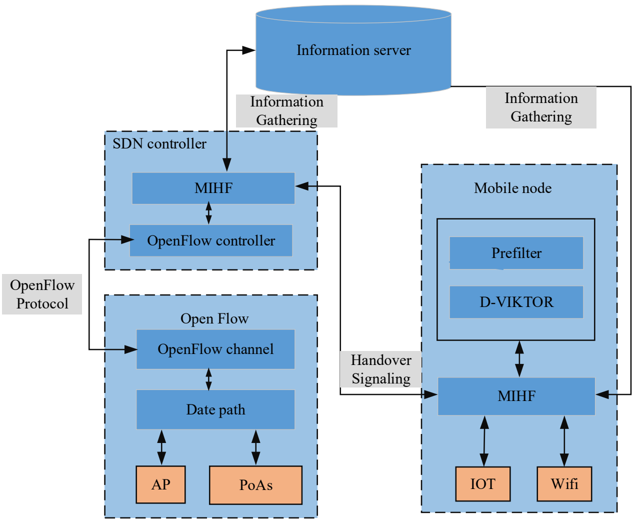
As in the network architecture of Figure 2, a variety of hardware support is required, where the mobile node is the core of the end-side network, including smartphones, in-vehicle networks or IoT-related devices, and the node's network is the entity that needs to provide exchange information in order to build the network exchange entity function (MIH) in the protocol stack. At the end of the link there is a point of attachment of MIH, whose function is to exchange information with end-side devices, in the traditional network construction PoA as a cellular network and WLAN access point, can send operational channels and power so as to make control decisions at the radio layer. When the PoA needs to switch and resource control requires hardware to support this control function, this controller needs to meet the collaborative switching function with neighbouring PoA to meet the minimisation of delay and network efficiency in the scenario group. The controller is also responsible for making data flow control decisions for newly added endpoints to reach new forwarding tasks. The SDN's controller is responsible for managing the hardware facilities for accessing resources, and it performs flow control in the form of data forwarding to send lower layer PoA services to the nodes. Connected to the SDN server is the information server, which also manages the end-side devices, and therefore requires hardware that meets the overall view requirements of the endpoints while being able to generate media-independent access and distribution. The framework built for the study tries to get rid of the strong control over the medium and infrastructure network, and in the design the wireless access network has the ability to operate at different types of interfaces and is also equipped with a switching management protocol that allows the user to achieve a flexible switching of the heterogeneous network with the collaboration of each controller, while also being able to query the prerequisites for switching.

The framework is to add IEEE802.21 to the routing execution in an integrated manner, where the detection process is performed first for permissions, and then through dynamic behaviour to obtain and vectorise and secure detection results, all processes will collaborate in parallel. During the event processing, mobile nodes have unknown nature, so the switching architecture needs to trigger an auxiliary pre-disconnect switch for a more comprehensive information selection. The extended network decision module allows network selection with sub-modules consisting of pre-filtering and Dynamic-Vlsekriterijumska Optimizacija I Kompromisno Resenje (D-VIKOR), which eliminates networks that are not compatible with the security policy, while the latter will perform optimisation algorithms to select a better target network (Lin et al., 2021). In scenarios such as the IoT, mobile nodes commonly used for vertical switching consume high amounts of energy and also require time to buffer in order to complete the switch, SDN addresses this situation by separating the planes to reduce complexity and latency, this process is supported in hardware by OpenFlow switches, this centralised control replaces one-time interactions (Ikpehai et al., 2018). Based on the above functionality, the switch needs to have various types of flow table entries for data forwarding, where the rule fields come first and are followed by actions and statistics. The update of the content of the flow table entries relies on the instructions of the protocol controller, so the controller of the SDN needs specific channels to control this hardware and the channel establishment facilitates the communication path between the nodes and the overall controller.

3.2 Soft switching based on end-side heterogeneous network

The end-side network is usually exposed to heterogeneous networks with different hardware architectures, so the switching process will be softened to accommodate the high arithmetic network response. In SDN architecture wireless access network can be controlled by MIH for various network forms, while protocol messages are forwarded between interfaces via switches and controllers. Vertical switching is the service process of cutting the data response for a session from one communication technology to another, and the specific logical flow is shown in Figure 3.

Figure 3 Vertical network switching process (see online version for colours)



The network switching process as in Figure 3 is usually initiated by the mobile node, when the node is a prerequisite for continuous monitoring of the link, and triggers the relevant indication when the link threshold drops. The end-side device then actively scans for neighbouring network information to prepare for the switch, and extracts upper and lower layer logical information and decision content from the controller through application messages. When the preparation is over, the node will have sufficient information for selection, when pre-filtering and selection algorithms are required for decision making. After the decision is made the node will request a connection through application messages, the controller receives the information and updates the flow table, finally completing the switching process, and although the network is switched, the network before and after will remain active for subsequent switching. The pre-filtering of the whole process is set based on the conflicting security policies, which aims to reduce the required time and delay possibilities, and the nodes can pre-learn the authentication and encryption related contents with the support of the information server, which will be used as an alternative based on the security needs of the end-side users. In addition, the pre-filtering mechanism is also an optimal path to reduce the computational cost of D-VIKOR (Gao et al., 2018). The studied VIKOR algorithm will incorporate dynamic elements to reduce the time cost of network execution. Let the attributes remain unchanged for the n^{th} execution, define $dynamic(n)$ as dynamic attributes, $static(n)$ as a static set of attributes, and the i^{th} choice as in equation (1).

$$f_i(n) = \left(static_i(n-1);\, dynamic(n)\right) \tag{1}$$

Under the precondition of equation (1), the set of available networks $a(n)$ is changed and the result will be reused and continuously selected, which is the standard VITOR. When weighting and normalisation are processed in the algorithm, the Manhattan distance is divided into dynamic and static parts, and if it is static, the distance remains unchanged, and then the normalised value is calculated as in equation (2).

$$N\left(static_{ij}(n)\right) = \begin{cases} N\left(static_{ij}(n-1)\right); & \text{if } a(n) = a(n-1) \\ \dfrac{static_{ij}(n)}{\sqrt{\sum_{i=1}^{a(n)} static_{ij}(n)^2}}; & \text{else} \end{cases} \tag{2}$$

$i = 1, \ldots, a(n)$, $j = 1, \ldots, ts$ and ts of equation (2) are the number of static attributes, and the normalised values of dynamic attributes are as in equation (3).

$$N\left(dynamic_{ij}(n)\right) = \frac{dynamic_{ij}(n)}{\sqrt{\sum_{i=1}^{a(n)} dynamic_{ij}(n)^2}} \tag{3}$$

The, $i = 1, \ldots, a(n)$ $j = 1, \ldots, ds$ and ds of equation (3) are the number of dynamic attributes. The positive $f^*(n)$ ideal solution is as in equation (4).

$$f*(n) = \begin{cases} (static^*(n-1),\, dynamic^*(n)); \\ \quad \text{if } a(n) = a(n-1) \\ \left(static^*(n),\, dynamic^*(n)\right); \\ \quad else \end{cases} \tag{4}$$

The $dynamic^*(n)$ in equation (4) is the dynamic positive solution, which is calculated as in equation (5).

$$\begin{aligned} dynamic^*(n) &= \left(d_1^*(n), \ldots, d_j^*(n), \ldots, d_{ds}^*(n)\right) \\ &= \left(\max_i dynamic_{ij}(n) \middle| j \in J_{dynamic1}\right), \\ &\quad \left(\min_i dynamic_{ij}(n) \middle| j \in J_{dynamic2}\right) \end{aligned} \tag{5}$$

The $J_{dynamic1}$ and $J_{dynamic2}$ of equation (5) are maximising the set of positive dynamic attributes and minimising the set of dynamic attributes, respectively. The static positive solution $static^*(n)$ is calculated as in equation (6).

$$\begin{aligned} static^*(n) &= \left(s_1^*(n), \ldots, s_j^*(n), \ldots, s_{ts}^*(n)\right) \\ &= \left(\max_i static_{ij}(n) \middle| j \in J_{static1}\right), \\ &\quad \left(\min_i static_{ij}(n) \middle| j \in J_{static2}\right) \end{aligned} \tag{6}$$

The $J_{static1}$ and $J_{static2}$ of equation (6) are maximising the set of positive static attributes and minimising the set of negative static attributes, respectively. The negative $f^-(n)$ solution is shown in equation (7).

$$f^-(n)=\begin{cases} static^-(n-1),\ dynamic^-(n); \\ \quad \text{if } a(n)=a(n-1) \\ static^-(n),\ dynamic^-(n); \\ \quad \text{else} \end{cases} \tag{7}$$

The $dynamic^-(n)$ of equation (7) is the dynamic negative solution and is calculated as in equation (8).

$$\begin{aligned} dynamic^-(n) &= \left(d_1^-(n),\ldots,d_j^-(n),\ldots,d_{ds}^-(n)\right) \\ &= \left(\min_i dynamic_{ij}(n) \middle| j\in J_{dynamic1}\right), \\ &\quad \left(\max_i dynamic_{ij}(n) \middle| j\in J_{dynamic2}\right) \end{aligned} \tag{8}$$

The $J_{dynamic1}$ and $J_{dynamic2}$ of equation (8) are maximising the set of positive dynamic attributes and minimising the set of dynamic attributes, respectively. The static negative solution $static^-(n)$ is calculated as in equation (9).

$$\begin{aligned} static^-(n) &= \left(s_1^-(n),\ldots,s_j^-(n),\ldots,s_{ts}^-(n)\right) \\ &= \left(\min_i static_{ij}(n) \middle| j\in J_{static1}\right), \\ &\quad \left(\max_i static_{ij}(n) \middle| j\in J_{static2}\right) \end{aligned} \tag{9}$$

The $J_{static1}$ and $J_{static2}$ of equation (9) are maximising the set of positive static attributes and minimising the set of negative static attributes, respectively. After deciding the ideal solution the partial distance of the static set of attributes *Set* is calculated, where $SetS_i(n)$ is calculated as in equation (10).

$$SetS_i(n)=\begin{cases} SetS_i(n-1); \quad \text{if } a(n)=a(n-1) \\ \sum_{j=1}^{ts}\dfrac{static_j^*(n)-static_{ij}^*(n)}{static_j^*(n)-static_j^-(n)}; \quad \text{else} \end{cases} \tag{10}$$

i = 1, …, $a(n)$ of equation (10). The $SetDynamic_i(n)$ calculation is shown in equation (11).

$$SetDynamic_i(n)=\begin{cases} \sum_{j=1}^{ds}\dfrac{static_j^*(n)-static_{ij}^*(n)}{static_j^*(n)-static_j^-(n)}; \\ \sum_{j=1}^{a(n)}\omega_j=1 \end{cases} \tag{11}$$

In equation (11), ω_j is the association weight of the j attribute. At this point, the normalised Manhattan distance and Chebyshev distance values of S_i and R_i will be calculated, as in equation (12).

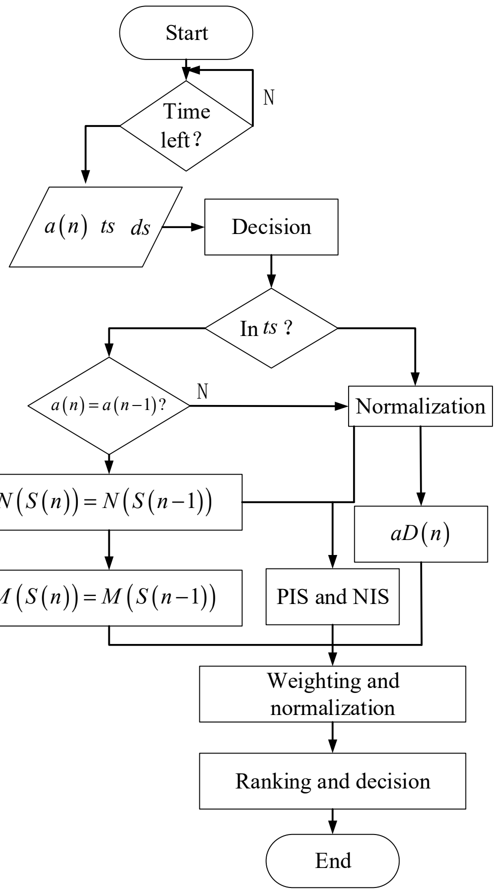
$$\begin{cases} Statci_i(n)=SetStatic_i(n)+Dynamic_i(n) \\ R_i(n)=\max\left(\dfrac{Static_j^*(n)-Static_{ij}^*(n)}{Static_j^*(n)-Static_j^-(n)}\cdot\omega_j\right) \end{cases} \tag{12}$$

After the calculation of equation (12), the value of Q_i will be calculated and executed, and then the values will be sorted as in equation (13).

$$Q_i(n)=\frac{v\left(S_i(n)-S_i^*(n)\right)}{S_i^-(n)-S_i^*(n)}+\frac{(1-v)\left(R_i(n)-R_i^*(n)\right)}{R_i^-(n)-R_i^*(n)} \tag{13}$$

The v of equation (13) is the compromise parameter. In the specific switching decision, four attributes exist for the mobile node to choose, where the network monetary cost and reserved pants pocket eh static, and the mobile consumption and signal strength indication are dynamic. During network communication, the dynamic values need to be measured at a fixed time, and the mobile signal strength can be determined by PoA, so the pseudo code and execution process is shown in Figure 4.

Figure 4 D-VIKROR Indicates the running flow



As in Figure 4 the switching process exists in a situation where cellular switching is an unnecessary process when the node is at the Wi-Fi coverage boundary and therefore this switching needs to be omitted. Also to reduce the switching vertical frame will use equation (14) for network measurements.

$$LT=ORT-DTP \tag{14}$$

The LT of equation (14) is the remaining lifetime, ORT is the current PoA out-of-range time, and DTP is the data transfer cycle timeout value, at which point ORT is calculated as in equation (15).

$$ORT=\frac{2R-D}{S} \tag{15}$$

The *R* of equation (15) is the coverage diameter of the PoA, *S* is the network speed, and *D* is the distance between the node and the PoA. The network lifetime is evaluated using (14) and (15), and a positive value results in a switching decision, while a negative value results in staying in the current network.

4 Heterogeneous switching effect and performance analysis of end-side network

The study will test the performance of the proposed network architecture framework under sufficient arithmetic power and network conditions, as well as perform arithmetic analysis on the decision performance of the algorithm to verify the reasonableness of the switching decision, and finally, a scenario will be built to test the end-side network in a real environment. The performance testing environment for this study is the NIST-provided NS-2.29 platform; the heterogeneous network includes one Wi-Fi, one mobile node, one UMTS and one WiMAX; the topology area radius is 2 km, and the controller will manipulate the access points of each base station. The end-side nodes use IEEE802.21 standard. The tested metrics include packet loss rate (PLR), switching delay and energy consumption, some of these metrics cannot be measured directly, so the metrics such as execution time and efficiency will be extended to enrich the coverage of the metrics. The network load and throughput will be tested first, and the results based on the number of users and the number of networks are shown in Figure 5.

Figure 5(a) reflects the switching frequency of the terminal while moving. It can be seen that as the number of networks increases, the switching frequency of the D-TOPSOS architecture network always exceeds 40, while the switching frequency of the SDN-based network architecture is always lower than that of the D-TOPSIS network architecture. When the number of networks is 5, the number of SDN switches is 18, while when the number of networks is 20, the number of SDN switches is 36. The average switching frequency of SDN is 26, while the average switching frequency of D-TOPSIS architecture network is 42, which is 16 times more than SDN. This ensures the time waste caused by excessive switching times, and also reflects the accurate multi-attribute decision making of SDN architecture, which can meet network requirements with fewer times. Figure 5(b) simulates the throughput of users using network services, and the results show that as the number of terminal sides increases, the throughput of both network architectures shows a first decrease and then an increase. The throughput of SDN is always higher than that of D-TOPSIS architecture. The average throughput of SDN is 206 Mbit/s, while the average throughput of D-TOPSIS is 168 Mbit/s, which is 34 Mbit/s less than SDN.

Figure 5 (a) Network switching times (b) User throughput for both architectures (see online version for colours)

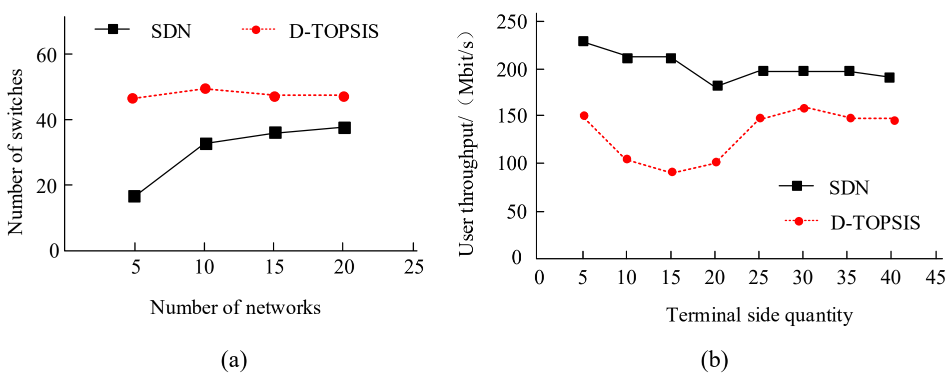
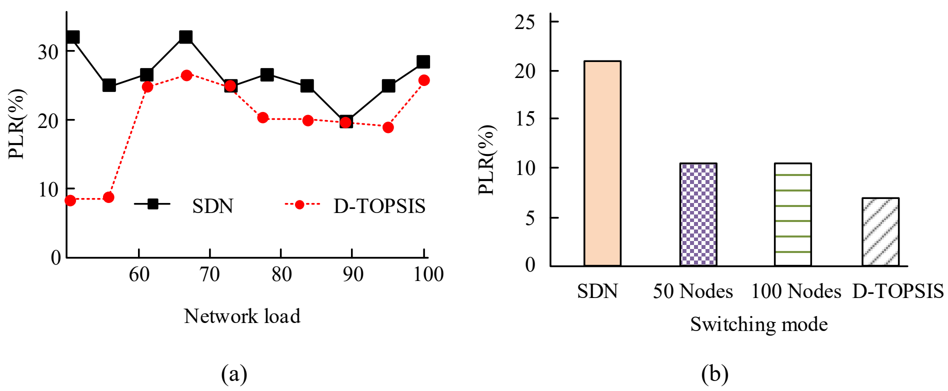


Figure 6 Packet loss performance of algorithms and switching modes, (a) network load and packet loss performance (b) average packet loss rate (see online version for colours)



To test the relationship between network load and PLR based on SDN network architecture, we studied the recording of packet loss rate changes under different network loads, as shown in Figure 6.

In the packet loss pairing shown in Figure 6(a), as the network load increases, the packet loss rate of SDN fluctuates between 20% and 30%. When D-TOPSIS reaches 55% and 65% of the network load, the packet loss rates of SDN are 9.5% and 24.3%, respectively, with a difference of 14.8%. After 70% of the network load points, the packet loss rate of D-TOPSIS remains slightly fluctuating between 20% and 30%. The load fluctuation of SDN is relatively small, with an overall fluctuation range of within 10%. The overall number of data packets in SDN is greater than that of D-TOPSIS because the switching decision algorithm includes both dynamic and static performance. When the load increases, the SDN controller will switch to a better network as a substitute, while the power consumption of D-TOPSIS is static and cannot respond to the load in a timely manner, making it difficult to achieve the expected switching. The switching process also cannot respond to the load. Figure 6(b) shows the comparative test results of multiple experimental results. There is also a difference in the number of PoA in the SDN framework. When the overall network nodes are 50 and 100, there is a delay in the data packet, making it difficult to make decisions in a timely manner. However, responsive switching can achieve good results, with an average PLR of 21.03%.

To test the relationship between network load and total power consumption of SDN-based network architecture, and compare the energy consumption effect of D-TOPSIS. Study recording the total power consumption under different network load conditions. The results are shown in Figure 7.

In the energy consumption test results of Figure 7, the D-TOPSIS energy consumption is always higher than the SDN network architecture, while the energy consumption of SDN fluctuates around 0.04 W with a small amplitude. This is because it appropriately switches to the connection of the backup network by treating power consumption as a dynamic attribute. For example, when the network load reaches 40% and 100%, the SDN-based network architecture consumes 0.0382 watts of power, while when the network load is 100%, SDN reduces power consumption by 24.15% compared to D-TOPSIS. In contrast, the energy consumption of D-TOPSIS is prone to significant fluctuations during load, and the energy consumption is more than twice that of SDN-based vertical switching. The reason for this is that when cellular network nodes are under high load, intelligent allocation incorporates energy consumption into the switching criteria, and the controller makes decisions towards the nodes with fewer allocations, resulting in smaller fluctuations in energy consumption. When D-TOPSIS is under high load, long-term connections will generate a large amount of energy consumption, and load related standards are not considered in network decision making, so energy consumption fluctuations will fluctuate widely with the load.

Figure 7 Energy consumption under different loads (see online version for colours)

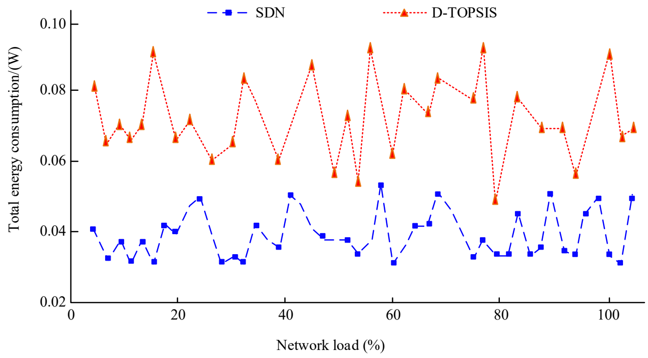


Figure 8 Mobile node speed and switching utility test, (a) switching utility of different architectures (b) SDN switching mode (see online version for colours)

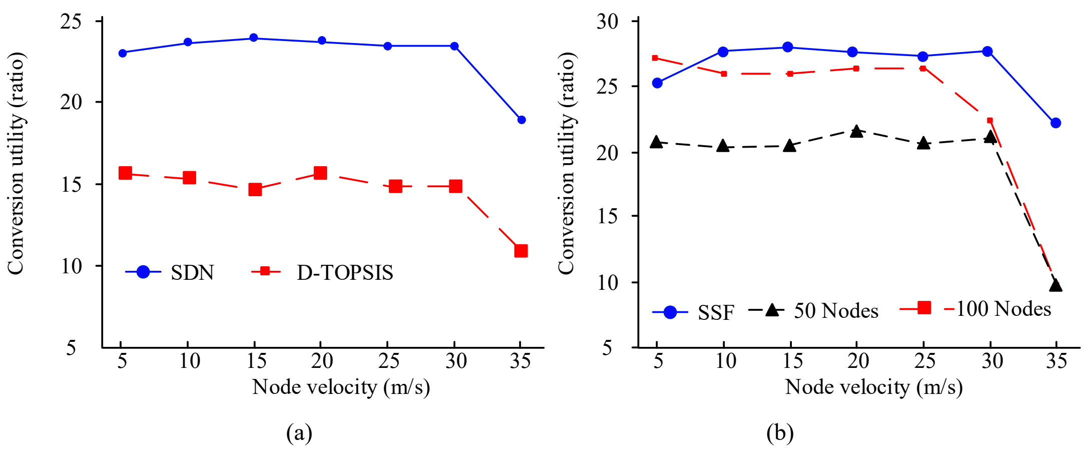
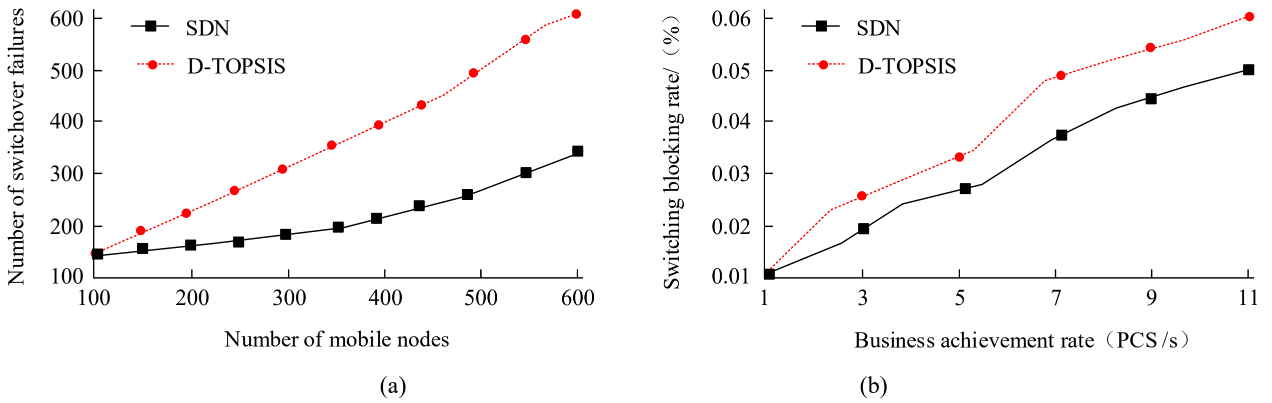


Figure 9 Toggle blocking and failure tests, (a) number if failed switchover tests (b) switching blocking rate (see online version for colours)



To further test the performance of the network architecture designed by the research institute, the switching utility of SDN and D-TOPSIS was obtained by changing the speed of mobile nodes. And record the switching utility changes between the two network architectures as the mobile node speed changes in Figure 8(a). To test the performance of the switching forms selected in the study, the switching efficiency of different switching forms in four SDN architectures was compared. The comparison results are shown in Figure 8(b). The relationship between the speed of mobile nodes and switching utility is also one of the indicators to measure the performance of network architecture. Figure 8 shows the test results based on node speed and utility, which includes comparisons between architectures and intra architecture switching methods.

Figure 8(a) shows the performance comparison between the network architecture of the research architecture and D-TOPSIS. As the node speed increases, the performance of SDN switching changes relatively little and remains basically unchanged. This is because the switching speed of the network architecture is measured based on average network usage and node speed, which minimises unnecessary switching, which is the key to maintaining switching performance. But there is also a certain threshold for this speed, and when the speed reaches 35 m/s, the aging energy will decrease. In contrast, the switching utility of D-TOPSIS tends to decrease in average with the increase of node speed. This is because this switching mode does not minimise invalid switching, resulting in a gradual decrease in performance as computing power increases, maintaining an overall rate of around 15. Figure 7(b) shows the comparison of switching modes within the SDN framework. When the nodes are 50 and 100, the efficiency is lower than that of SSF. This is because the responsive switching response speed and decision-making process are fast, and the switching performance can be directly adjusted. The overall efficiency is maintained at 25–30 ratios, which increases the efficiency by 10.21% compared to D-TOPSIS. Overall, the SDN network architecture designed by the research institute and the selected switching forms have good switching effectiveness, which can effectively improve switching effectiveness.

In the context of high computing power, the number of switching failures increases with the increase of nodes. To test the performance of the network architecture designed by the research institute, mobile nodes were added to test the number of switching failures and the achievement of switching tasks. The results are shown in Figure 9.

In the statistics of switch failures in Figure 9(a), the increase in the number of nodes is consistent with the background of network integration. Based on this, the number of switch failures in the end side network of SDN architecture is significantly better than D-TOPSIS. After 100 nodes, the difference between the two is even more significant, reaching a maximum of 600 times, while the number of vertical switch errors in SDN architecture is up to 315 times, which is 285 times less than D-TOPSIS. When the number of nodes is 100, the number of vertical switching errors in the SDN architecture is 158, while the number of errors in TOPSIS is 212, a difference of 54. After more than 100 attempts, the gap between the two gradually widened. When the number of nodes is 500, the number of vertical switching errors in SDN is 256, while the number of errors in TOPSIS is 511, which is 255 more than in SDN. This indicates that switching decisions can centrally control switching performance and achieve optimal switching results. In the switching blocking experiment in Figure 9(b), as the traffic rate increases, the likelihood of blocking will continue to increase. When the traffic rate is 3 PCS/s, the constrictivity of SDN is 0.016%, while the constrictivity of D-TOPSIS is 0.025%, an increase of 0.009% compared with SDN. When the traffic rate is 11 PCS/s, the constrictivity of SDN is 0.041%, and the constrictivity of D-TOPSIS is 0.059%, an increase of 0.018% compared with SDN. It can be seen from Figure 9(b) that the vertical handover constrictivity of SDN is significantly lower than that of D-TOPSIS. When blocking mapping is applied in actual scenarios, an increase in users will lead to a decrease in available networks. D-TOPSIS alone will choose the network with the strongest signal strength, but in actual scenarios, strength is not the only criterion for network performance, and the vertical switching of SDN is more in line with real scenarios.

In actual network scenarios, nodes are usually mobile, such as the vehicular Internet of Things. Therefore, the delay caused by node movement speed is also one of the standards for measuring the end-to-end network system. The study recorded the total switching delay changes of two network systems by increasing the speed of mobile nodes, and the test results are shown in Figure 10.

Figure 10 Node movement speed and total switching delay (see online version for colours)

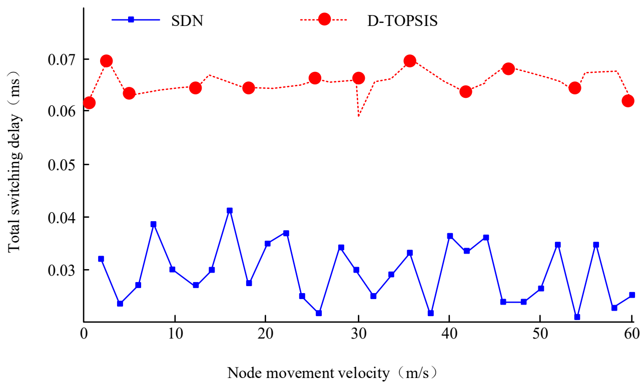


Figure 10 shows the test results under node mobility, where the vertical switching of SDN can stabilise the delay at around 0.03 ms under both low and high speed conditions, while the network delay of D-TOPSIS is above 0.06 ms. When the node mobility is 30 m/s, the handover delay of SDN is 0.037 ms, while the handover delay of TOPSIS is 0.052 ms. Compared to TOPSIS, SDN reduces the handover delay by 27.55%. The reason is that vertical switching measures the lifespan of PoA before making decisions, and pre filtering and D-VIKOR can reduce incompatible content, resulting in lower latency. However, D-TOPSIS switching has extended switching delay, which requires frequent switching under high-speed conditions, which reduces network performance and increases latency. 30 m/s is a common speed for urban vehicle network movement, and under this condition, the vertical handover delay is reduced by 27.55%, indicating that the end side architecture meets the mobility conditions.

5 Future research scope and discussion

The research work can provide new solutions for future ubiquitous access networks, promoting seamless integration of heterogeneous networks and intelligent decision-making at the network end by combining SDN and MIH technologies. However, based on this research work, further in-depth research can be conducted on network decision making and switching authentication directions. The specific work is as follows:

1. Further research is needed on security-based pre filtering mechanisms, and in the future, fine-grained strategies can be developed based on application and user needs to achieve adaptive and dynamic pre filtering.
2. In the process of integrating SDN and MIH technologies, the protocols designed for each technology are not the same, and there may be compatibility issues during the integration process. There is still a large optimisation space for communication loads. In the future, we can consider developing a unified new protocol based on media independence and network side decision-making ideas.
3. In the future, consideration can be given to implementing the proposed framework and algorithms in practical environments.

6 Conclusions

In the context of computing network integration, network nodes face the dual challenges of network heterogeneity and complex scenarios. At the same time, increasing computing power will also increase the difficulty of network architecture. Therefore, finding a system that can handle various problems flexibly and ensure network smoothness has become a research focus. Based on this, a new network architecture has been constructed with the support of SDN and media independent switching technology. This architecture will achieve intelligent development by separating data and operating platforms, and adding controllers to centrally manage the overall architecture, while increasing operability. With the support of pre filtering and D-VIKOR algorithm, network switching is more reasonable, providing support for SDN centralised decision-making and heterogeneous integration. As the number of networks increases, the switching frequency of the D-TOPSOS architecture network always exceeds 40, while the switching frequency of the SDN-based network architecture is always lower than that of the D-TOPSIS network architecture. The average switching frequency of SDN is 26 times, while the average switching frequency of D-TOPSIS architecture network is 42 times, which is 16 times more than SDN, indicating that SDN network has obvious advantages in transmitting data. When D-TOPSIS reaches 55% and 65% of the network load, the packet loss rates of SDN are 9.5% and 24.3%, respectively, with a difference of 14.8%. As the network load increases, the packet loss rate of SDN fluctuates between 20% and 30%, and the load fluctuation of SDN is relatively small, with an overall fluctuation range of within 10%. It can be seen that the responsive switching of the end-to-end network system constructed by the research institute can achieve good results, with an average PLR of 21.03%. In performance testing, when the mobile speed reached 30 m/s, the network architecture performance improved by 27.55%. It can be seen that the terminal constructed by the research institute has a low number of end-to-end network switching times and high efficiency when in a mobile state. In the switching efficiency experiment, the responsive switching efficiency was maintained at a 25–30 ratio, which increased by 10.21% compared to D-TOPSIS. Therefore, the results demonstrate that the end-to-end network architecture constructed in the study can achieve centralised control of the network, while switching and softening processing can improve the efficiency of network resource allocation.

References

Badotra, S. and Panda, S.N. (2020) 'Evaluation and comparison of OpenDayLight and open networking operating system in software-defined networking', *Cluster Computing*, Vol. 23, No. 2, pp.1281–1291.

Casas-Velasco, D.M,, Rendon, O.M.C. and da Fonseca, N.L.S. (2020) 'Intelligent routing based on reinforcement learning for software-defined networking', *IEEE Transactions on Network and Service Management*, Vol. 18, No. 1, pp.870–881.

Chen, C., Wang, C., Qiu, T., Atiquzzaman, M. and Wu, D.O. (2020) 'Caching in vehicular named data networking: architecture, schemes and future directions', *IEEE Communications Surveys & Tutorials*, Vol. 22, No. 4, pp.2378–2407.

Constantinides, P., Henfridsson, O. and Parker, G.G. (2018) 'Introduction – platforms and infrastructures in the digital age', *Information Systems Research*, Vol. 29, No. 2, pp.381–400.

Fang, F., Cheng, J. and Ding, Z. (2018) 'Joint energy efficient subchannel and power optimization for a downlink NOMA heterogeneous network', *IEEE Transactions on Vehicular Technology*, Vol. 68, No. 2, pp.1351–1364.

Gao, Y., Sun, X. and Dai, L. (2018) 'Sum rate optimization of multi-standard IEEE 802.11 WLANs', *IEEE Transactions on Communications*, Vol. 67, No. 4, pp.3055–3068.

Ikpehai, A., Adebisi, B., Rabie, K.M., Anoh, K., Anoh, K., Hammoudeh, M., Gacanin, H. and Mbanaso, W.M. (2018) 'Low-power wide area network technologies for internet-of-things: a comparative review', *IEEE Internet of Things Journal*, Vol. 6, No. 2, pp.2225–2240.

Kato, N., Mao, B., Tang, F., Kawamoto, Y. and Liu, J. (2020) 'Ten challenges in advancing machine learning technologies toward 6G', *IEEE Wireless Communications*, Vol. 27, No. 3, pp.96–103.

Lin, H., Boecker, C., Hossain, M., Tangirala, S., Vu, R., Vamvakos, S.D., Groen, E., Li, S., Choudhary, P., Wang, N., Shibata, M., Taghavi, H., Marcus, V.I., Maniyar, A.H., Wodkowski, A., Brar, K., Nguyen, N. and Desai, S. (2021) 'ADC-DSP-based 10-to-112-Gb/s multi-standard receiver in 7-nm FinFET', *IEEE Journal of Solid-State Circuits*, Vol. 56, No. 4, pp.1265–1277.

Nawaz, S.J., Sharma, S.K., Wyne, S., Patwary, M.N. and Asaduzzaman, M. (2019) 'Quantum machine learning for 6G communication networks: State-of-the-art and vision for the future', *IEEE Access*, Vol. 7, pp.46317–46350.

Qiu, J., Grace, D., Ding, G., Zakaria, M.D. and Wu, Q. (2019) 'Air-ground heterogeneous networks for 5G and beyond via integrating high and low altitude platforms', *IEEE Wireless Communications*, Vol. 26, No. 6, pp.140–148.

Raja, G., Ganapathisubramaniyan, A. and Anbalagan, S. (2020) 'Intelligent reward-based data offloading in next-generation vehicular networks', *IEEE Internet of Things Journal*, Vol. 7, No. 5, pp.3747–3758.

Sahoo, K.S., Tripathy, B.K., Naik, K., Ramasubbareddy, S., Balusamy, B., Khari, M. and Burgos, D. (2020) 'An evolutionary SVM model for DDOS attack detection in software defined networks', *IEEE Access*, Vol. 8, pp.132502–132513.

Siriwardhana, Y., Gür, G., Ylianttila, M. and Liyanage, M. (2021) 'The role of 5G for digital healthcare against COVID-19 pandemic: opportunities and challenges', *ICT Express*, Vol. 7, No. 2, pp.244–252.

Tseng, L., Wong, L., Otoum, S., Aloqaily, M. and Othman, J.B. (2020) 'Blockchain for managing heterogeneous internet of things: a perspective architecture', *IEEE Network*, Vol. 34, No. 1, pp.16–23.

Wang, X., Li, X., Pack, S., Han, Z. and Leung, V.C.M. (2020) 'STCS: spatial-temporal collaborative sampling in flow-aware software defined networks', *IEEE Journal on Selected Areas in Communications*, Vol. 38, No. 6, pp.999–1013.

Wang, Y., Jiang, D., Huo, L. and Zhao, Y. (2021) 'A new traffic prediction algorithm to software defined networking', *Mobile Networks and Applications*, Vol. 26, No. 2, pp.716–725.

Xue, X., Nakamura, F., Prifti, K., Pan, B., Yan, F., Tsuda, H.G.X. and Calabretta, N. (2020) 'SDN enabled flexible optical data center network with dynamic bandwidth allocation based on photonic integrated wavelength selective switch', *Optics Express*, Vol. 28, No. 6, pp.8949–8958.

Zhang, H., Jiang, C., Feng, Z., Zhang, Z. and Leung, V.C.M. (2018a) '5G technologies for future wireless networks', *Mobile Networks and Applications*, Vol. 23, No. 6, pp.1459–1461.

Zhang, S., Chen, J., Lyu, F., Cheng, N., Shi, W. and Shen, X. (2018b) 'Vehicular communication networks in the automated driving era', *IEEE Communications Magazine*, Vol. 56, No. 9, pp.26–32.