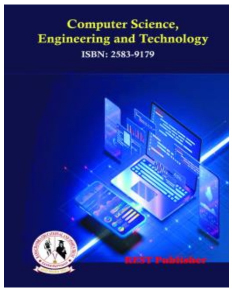
Computer Science, Engineering and Technology
Vol: 2(2), June 2024
REST Publisher; ISSN: 2583-9179 (Online)
Website: https://restpublisher.com/journals/cset/
DOI: https://doi.org/10.46632/cset/2/2/5



Network Function Virtualization (NFV) for flexible network services by using TOPSIS method

***[1]Nihari Paladugu, [2]Deepak Das, [2]Somesh Nagalla**
[1]IEEE Member, Ohio, USA.
[2]IEEE Member, Texas, USA.
**Corresponding Author Email: Niharinani@gmail.com*

Abstract: *Network Function Virtualization (NFV) has emerged as a transformative paradigm in the field of telecommunications and networking. It introduces a revolutionary approach by decoupling network functions from dedicated hardware appliances and instead, vitalizing them to run on general-purpose servers. This shift from traditional, hardware-centric infrastructure to a more flexible and scalable virtualized environment offers numerous advantages, including improved resource utilization, reduced operational costs, and enhanced agility in deploying and managing network services. As the telecommunications industry continues to evolve towards a more software-centric model, NFV stands out as a pivotal enabler for achieving flexibility, scalability, and efficiency in the delivery of next-generation network services. The ongoing evolution of NFV technologies, coupled with collaborative industry efforts, holds the promise of unlocking new possibilities and driving innovation in the dynamic world of telecommunications. NFV allows for the virtualization of network functions, enabling them to run on commodity hardware. This leads to better resource utilization and reduced dependency on specialized hardware appliances, translating to cost savings for service providers. Research in NFV provides valuable insights into real-world applications and case studies, demonstrating how organizations can leverage NFV to solve specific challenges, improve efficiency, and deliver innovative network services. Method: TOPSIS, Using the positive ideal solution and the negative ideal solution as reference solutions, this method calculates the geometric distance between each alternative solution. The fundamental tenet of TOPSIS is that the performance criteria are assumed to be ascending, with higher numbers denoting superior performance. In order to accommodate different dimensions or scales between the criteria, the TOPSIS architecture frequently uses normalization. Result: From the result Isolation and security is got the first rank and Vendor lock in mitigation is having the lowest rank.*

1. INTRODUCTION

The concept behind NFV aimed to reduce (CapEx) and (OpEx) associated with various proprietary equipment by harnessing virtualization technologies. Through the utilization of virtualization technologies and standard servers, network operators and service providers can execute network functions in software rather than relying on specialized hardware, courtesy of (NFV). The integration of NFV with Software-Defined Networking (SDN) is shaping a software-defined NFV architecture, driven by recent trends such as increased user information demands, a surge in traffic, and diverse service requirements. This architecture empowers operators to deliver services and model them with heightened flexibility, programmability, and automation. The optimization of Service Function Chaining (SFC) deployment within cloud systems heavily relies on the utilization of a (NFV)-Aware Orchestrator. This orchestrator strategically allocates virtualized network services, considering factors such as performance and resource availability. By doing so, it enhances the overall flexibility and responsiveness of network services, ensuring efficient service delivery, scalability, and optimal resource utilization in dynamic cloud environments.[3] A simulation framework named CloudSim SDN-NFV was developed to simulate and (SFC) and (NFV) within edge computing environments. Its primary goal is to enable researchers to assess the efficiency, scalability, and performance of virtualized network operations in edge computing scenarios, incorporating Software-Defined Networking (SDN) principles. This framework facilitates a better understanding of the dynamic interactions among NFV, SFC, and edge computing, thereby simplifying the creation of optimized edge-based services and infrastructures [4].One of the biggest innovations in the world of communication network technology is

SDN/NFV. To achieve full network programmability, SDN decouples the network devices' data and control planes. In a real or virtual network environment, SDN handles traffic shaping and flexible forwarding. Conversely, network function virtualization (NFV) manages the flexible placement of virtualized network functions throughout the network and cloud. To achieve complete network programmability, flexibility, and cost effectiveness, SDN and NFV are complementary technologies.[6]Network function virtualization (NFV) Dynamic Reliability-Aware Service Placement is a proactive method of optimizing virtualized network function placement based on reliability considerations. This technique improves the overall dependability of NFV-based services by dynamically evaluating resource reliability and making real-time placement adjustments. This ensures strong and resilient network function deployments in response to changing conditions or probable failures.[7]Incorporating NFV concepts into cloud environments represents a pivotal move in transforming Network Function Virtualization (NFV) into a service within the realm of Cloud Computing. This transition enables the delivery of network functionalities as on-demand services, leveraging the infrastructure of cloud computing. Through this adaptation, NFV enhances scalability, flexibility, and accessibility, aligning with the essential characteristics of cloud services that prioritize the efficient and dynamic utilization of network resources [8].The Virtualized Network Function Placement (VNF-P) model serves as a framework for the efficient implementation of virtualized network functions (VNFs). This model strategically organizes VNFs within network architecture to enhance efficiency, considering factors such as cost, performance, and resource availability. Especially in dynamic and evolving network environments, VNF-P aims to enhance the overall deployment efficiency of virtualized network functions by optimizing resource utilization and refining the placement process. [9] A network service encompasses all the functionalities provided by network operators to users, spanning from initiation to completion. Typically, it consists of various Service Functions (SFs) interconnected in a sequence. Service Function Chaining (SFC) involves the creation of these chains and the routing of packets or flows through them. Essentially, a service function chain is a predetermined sequence of service functions that need to be executed in a specific order based on classification and/or policy. To address this, the Internet Engineering Task Force (IETF) has taken proactive measures by establishing a formal architecture for SFC built upon the ETSI NFV architecture. This initiative aims to establish an efficient, dynamic, and automated framework for Service Function Chaining.SDN's distinct advantage lies in its ability to provide an abstraction of the underlying network infrastructure. Its contribution to simplified network management is notable, achieved through network-wide programmability—a capability that allows for comprehensive alterations in the network's behavior. This heightened programmability, facilitated by SDN, permits the customization and optimization of multiple network slices for diverse service deployments while maintaining consistent physical and logical network architecture [11].One of the primary challenges confronting Network Function Virtualization (NFV) revolves around meeting the diverse array of management and orchestration needs associated with network services. Service descriptors offer a means for service developers to outline general management requirements, encompassing Virtual Network Function (VNF) resource specifications. However, they lack the ability to execute specialized management tasks such as VNF-specific configuration [12].

2. MATERIALS AND METHOD

2.1. Alternative parameters: Resource Utilization Efficiency, Orchestration Complexity, Isolation and Security, Fault Tolerance and High Availability, Vendor Lock-In Mitigation

2.2. Evaluation parameters: Performance, Customer Satisfaction (%), Resource Utilization, Deployment Time (days)

2.3. Resource Utilization Efficiency: Resource Utilization Efficiency pertains to the optimal use of computing resources in a system. It involves maximizing the performance of components like CPU, memory, and storage while minimizing wastage. Efficient resource utilization ensures that the available infrastructure is fully leveraged, promoting cost-effectiveness, scalability, and overall system responsiveness in network function virtualization and similar environments.

2.4. Orchestration Complexity: Orchestration Complexity refers to the intricacy involved in coordinating and managing the lifecycle of virtualized network functions within a system. It assesses how efficiently orchestration systems can deploy, scale, and decommission services. A lower orchestration complexity indicates streamlined processes, facilitating easier adaptation to changing network demands and enhancing overall operational efficiency in network function virtualization environments.

2.5. Isolation and Security: Isolation and Security in the context of network function virtualization (NFV) refer to the degree of protection and segregation among virtualized network functions to prevent unauthorized access and potential breaches. It assesses the robustness of measures in place to ensure data integrity, confidentiality, and overall resilience against security threats, providing a secure operational environment for NFV deployments.

2.6. Fault Tolerance and High Availability: Fault Tolerance and High Availability in network function virtualization (NFV) describe the system's ability to withstand and recover from failures while ensuring continuous service delivery. These parameters assess how well NFV solutions can maintain operational integrity, minimize downtime, and swiftly adapt to faults, enhancing overall reliability and resilience in dynamic network environments.

2.7. Vendor Lock-In Mitigation: Vendor Lock-In Mitigation refers to the strategies and technologies implemented to reduce the risk of being dependent on a single NFV vendor. It assesses how easily organizations can switch between different NFV vendors or migrate to alternative solutions without significant disruptions, fostering a more flexible and vendor-agnostic network infrastructure.

2.8. Performance: In the context of network function virtualization (NFV), performance refers to the efficiency and effectiveness of virtualized network functions (VNFs) and the overall system. It encompasses metrics such as throughput, latency, and response times, evaluating how well the NFV solution meets the performance requirements and objectives of the network services are provided. High-performance NFV ensures optimal service delivery and responsiveness, contributing to a positive user experience.

2.9. Customer Satisfaction (%): Customer Satisfaction (%) refers to the percentage of customers who express contentment or fulfillment with a particular product, service, or overall experience. It is a metric commonly used to gauge the level of satisfaction among customers and is often obtained through surveys, feedback forms, or other customer engagement channels. A higher percentage indicates a more satisfied customer base, while a lower percentage may signal areas that require improvement in a product or service.

2.10. Resource Utilization: In the context of network function virtualization (NFV) or other IT infrastructure, resource utilization typically includes aspects such as the optimal use of CPU (Central Processing Unit), memory, storage, and network bandwidth. High resource utilization implies that the system is effectively leveraging its available resources, ensuring optimal performance and minimizing waste. Monitoring and optimizing resource utilization are essential for maintaining system efficiency, scalability, and cost-effectiveness.

2.11. Deployment Time (days): Deployment Time (days) refers to the duration required to implement and activate a particular system, solution, or service within an organization. In the context of network function virtualization (NFV) or other IT deployments, A shorter deployment time is often desirable as it signifies faster implementation, reducing downtime and allowing organizations to quickly adapt to changing business needs or technological advancements.

2.12. Method: Modern TOPSIS methodology aims to efficiently select alternatives that are significantly close to the optimal solution while being noticeably distant from the worst-case scenario solution, achieved through the application of an effective and advanced ranking mechanism known as TOPSIS. When a superior response falls short, it results in a price increase, whereas an improved response from a superior broadens the criteria for advantages while narrowing down the criteria for price. The utilization of the TOPSIS technique [14] is based on comprehensive attribute records, encompassing essential FMCDM properties, two fuzzy membership activities, the TOPSIS algorithm, and a data collection spreadsheet. The title of this methodology delves into its rationale for use, ongoing challenges, limitations, and recommendations for researchers to enhance the adoption and utilization of FMCDM [15]. TOPSIS serves as an additional metric due to its unique characteristics, such as reduced components, increased stability, and a range of response values that capture various shifts in value, making it a more advantageous alternative to heuristics. The decision to develop TOPSIS was made [16].TOPSIS, short for "Technique for Order of Preference by Similarity to Ideal Solution," ranks alternatives using five different distance metrics. It does this by providing a numerical example involving randomly generated issues of various magnitudes for calculation. This method involves a comprehensive comparison of preference ranking sequences, considering factors like the consistency ratio, odds ratio of ideal alternatives, and average Pearson correlation coefficients. The first aspect addresses the relationship between two variables, while the second assesses the impact of measurements by comparing hypothetical outcomes to the mean count of coefficients. This method utilizes regression on rows. The compromise programming system introduces the concept of "Proximity to Ideal," which considers two criteria: "majority" and "minimum," aiming to maximize "group utility" for each grievance. These distance metrics are employed to determine solutions in the TOPSIS strategy, which effectively addresses both short-term and long-term challenges. It's important to note that the relevance of these factors is not considered. While TOPSIS appears to be logical, it has faced criticism. One critique is that it was adapted for addressing multi-objective decision-making (MODM) issues without adequately accounting for the relative importance of specific criteria or the problem's nature. PIS represent the shortest distance, while NIS represents the longest distance. Subsequently, a "condition of satisfiability" is defined for each criterion, followed by a maximum-minimum operator for these criteria. The application of Harmony, as mentioned in a previous study, helps resolve overlapping usages. TOPSIS is regarded as an efficient approach for achieving optimal regulatory performance. This method involves analyzing, contrasting, and evaluating various possibilities. Building on this foundation, the current study aims to expand TOPSIS's application to real-world group decision-making scenarios focused on assignments. The study outlines a comprehensive and successful selection method. The operation of TOPSIS is then concluded. The study initially examines the impact of the Weighted Euclidean (EW) approach on decision-making and evaluation processes, considering various statistical data and theoretical judgments [20].

3. RESULT AND DISCUSSION

TABLE 1. Network Function Virtualization (NFV) for flexible network services

	Performance	Customer Satisfaction (%)	Resource Utilization	Deployment Time (days)
Resource Utilization Efficiency	8.5	92	70	30
Orchestration Complexity	9	88	85	25
Isolation and Security	8.2	95	60	35
Fault Tolerance and High Availability	9.5	90	75	28
Vendor Lock-In Mitigation	8.8	87	80	32

Table 1 shows compare above values Performance: Fault Tolerance and High Availability have the highest rating at 9.5, indicating a very high level of performance in ensuring system availability. Orchestration Complexity and Resource Utilization Efficiency also have high ratings, at 9 and 8.5 respectively. Isolation and Security, as well as Vendor Lock-In Mitigation, have slightly lower ratings but are still above 8. Customer Satisfaction: Isolation and Security have the highest customer satisfaction rating at 95%. Fault Tolerance and High Availability, Resource Utilization Efficiency, and Vendor Lock-In Mitigation also have high satisfaction ratings, ranging from 87% to 92%. Orchestration Complexity has the lowest customer satisfaction rating at 88%. Resource Utilization: Orchestration Complexity has the highest resource utilization rating at 85%, indicating a higher level of resources required for orchestration. Vendor Lock-In Mitigation and Fault Tolerance and High Availability have resource utilization ratings of 80% and 75% respectively. Isolation and Security have the lowest resource utilization rating at 60%. Deployment Time: Orchestration Complexity has the lowest deployment time rating at 25 days, indicating a faster deployment process. Fault Tolerance and High Availability, Vendor Lock-In Mitigation, and Resource Utilization Efficiency have deployment time ratings ranging from 28 to 32 days. Isolation and Security have the highest deployment time rating at 35 days.

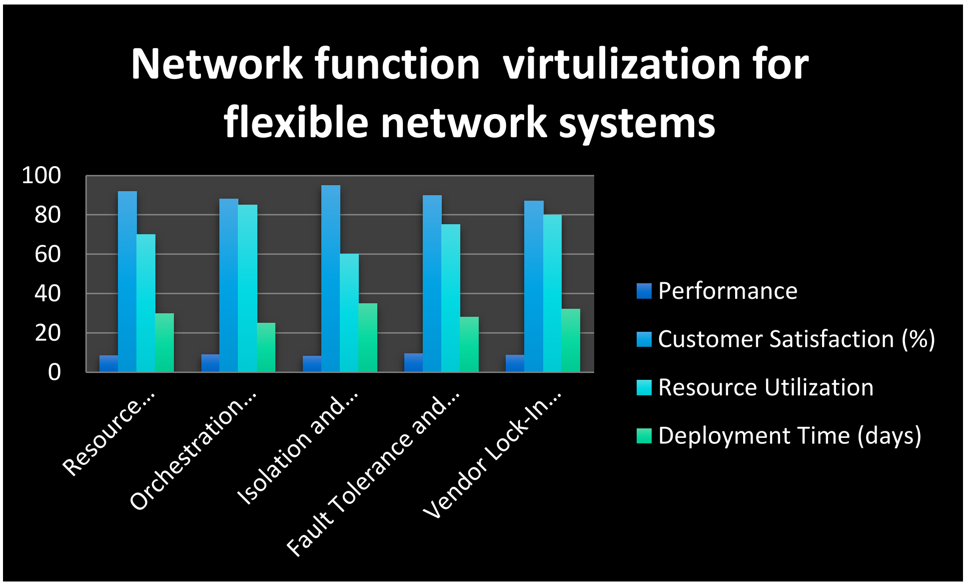


FIGURE 1. Network Function Virtualization (NFV) for flexible network services

Figure 1 illustrate graphical representation of Network Function Virtualization (NFV) for flexible network services

TABLE 2. Normalized Data

Normalized Data			
Performance	Customer Satisfaction (%)	Resource Utilization	Deployment Time (days)
0.4314	4.6695	0.4202	0.4444
0.4568	4.4665	0.5103	0.3703
0.4162	4.8218	0.3602	0.5184
0.4822	4.5680	0.4502	0.4147
0.4466	4.4157	0.4802	0.4740

Table 2 shows normalized value. Resource Utilization Efficiency: Normalized value: 0.4314 this indicates the efficiency of resource utilization, with a value between 0 (least efficient) and 1 (most efficient). Orchestration Complexity: Normalized value: 0.4568 represents the complexity of orchestrating tasks, processes, or components in the system. The value is normalized between 0 and 1. Isolation and Security: Normalized value: 0.4162 reflects the level of isolation and security measures in the system. Higher values suggest better isolation and security. Fault Tolerance and High Availability: Normalized value: 0.4822 Indicates the system's capability to tolerate faults and maintain high availability. A higher value represents better fault tolerance and availability. Vendor Lock-In Mitigation: Normalized value: 0.4466 Measures the degree to which the system avoids being tightly bound to a specific vendor. Higher values suggest better mitigation of vendor lock-in. For each metric, the normalization process likely involved converting the raw data into a scale from 0 to 1 using a formula like: Normalized Value = Original Value − Min Value/ Max Value−Min Value This ensures that each metric is comparable on the same scale, allowing for a more straightforward analysis of their relative importance or performance.

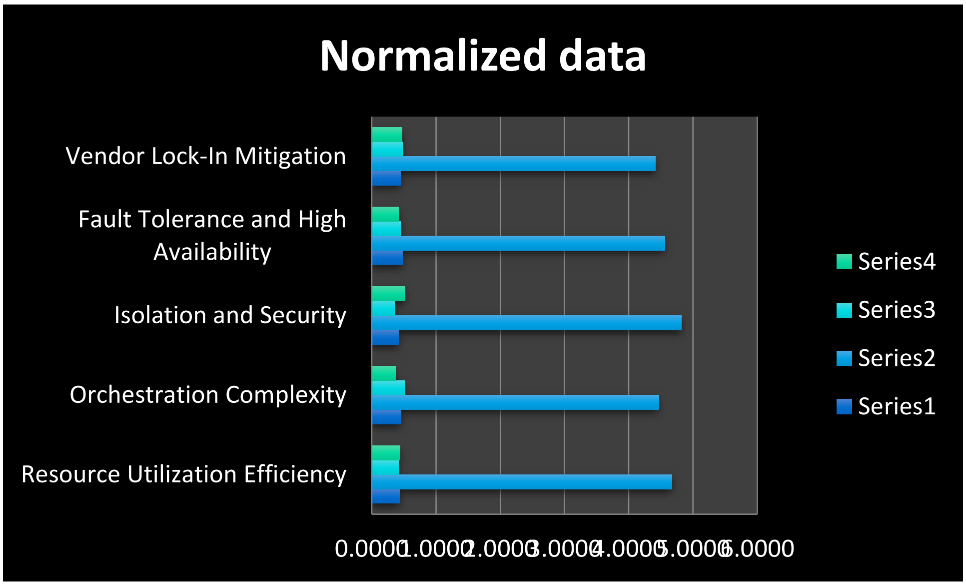


FIGURE 2. Normalized Data

Figure 2 illustrate graphical representation of Normalized data All the values in the matrix are the same (e.g., 0.25 in this case), which could indicate that each element in a set is given equal weight or importance. This could be used in various applications, such as when calculating averages or distributing resources equally.

TABLE 3. Weighted normalized decision matrix

Weighted normalized decision matrix			
0.1079	1.1674	0.1051	0.1111
0.1142	1.1166	0.1276	0.0926
0.1040	1.2054	0.0900	0.1296
0.1205	1.1420	0.1126	0.1037
0.1117	1.1039	0.1201	0.1185

Table 3 shows weighted normalized decision matrix Performance Metrics: "Resource Utilization Efficiency," "Orchestration Complexity," "Isolation and Security," "Fault Tolerance and High Availability," and "Vendor

Lock-In Mitigation." These metrics have been normalized, meaning they are scaled to a common range, typically between 0 and 1. Weights: The weights assigned to each metric (in each row) indicate the relative importance or priority of that metric in the decision-making process. Higher weights suggest that the corresponding metric is more critical in the evaluation. Values in the Matrix: Each cell in the matrix represents the product of the normalized value for a specific metric and the corresponding weight for that metric. These values are calculated by multiplying the normalized value from the first table with the corresponding weight in the second table. For example, for "Resource Utilization Efficiency," the calculation would be: 0.1079 × 0.4314, where 0.1079 is the weight for "Resource Utilization Efficiency," and 0.4314 is the normalized value for that metric.

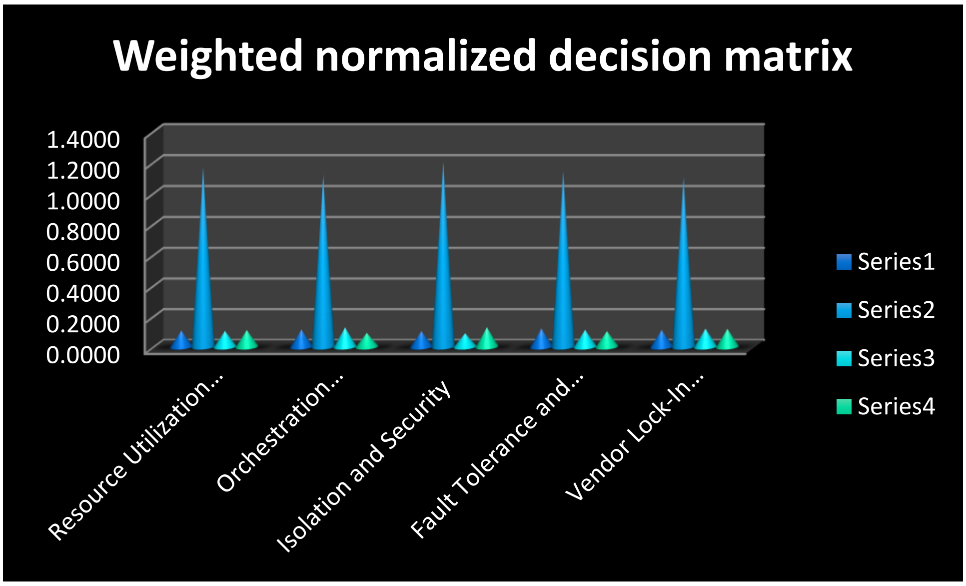


FIGURE 3. Weighted normalized decision matrix

Figure 3 illustrate graphical representation of Weighted normalized decision matrix has done

TABLE 4. Positive and Negative matrix

Positive Matrix				**Negative matrix**			
0.1205	1.2054	0.0900	0.0926	0.1040	1.1039	0.1276	0.1296
0.1205	1.2054	0.0900	0.0926	0.1040	1.1039	0.1276	0.1296
0.1205	1.2054	0.0900	0.0926	0.1040	1.1039	0.1276	0.1296
0.1205	1.2054	0.0900	0.0926	0.1040	1.1039	0.1276	0.1296
0.1205	1.2054	0.0900	0.0926	0.1040	1.1039	0.1276	0.1296

Table 4 shows positive and negative matrix. The positive matrix is used to highlight the positive aspects or strengths of each alternative across various criteria. each row represents a performance metric ("Resource Utilization Efficiency," "Orchestration Complexity," etc.), and the values in the matrix are typically the product of the normalized metric values and corresponding weights, just like in the weighted normalized decision matrix. the negative matrix is used to identify the negative aspects or weaknesses of each alternative. It often contains the complementary information to the positive matrix, highlighting areas where an alternative may perform poorly.

TABLE 5. Final result of Network Function Virtualization (NFV) for flexible network services

SI Plus	**Si Negative**	**Ci**	**Rank**
0.0467	0.0699	0.5997	2
0.0966	0.0404	0.2950	4
0.0405	0.1082	0.7275	1
0.0682	0.0512	0.4286	3
0.1093	0.0154	0.1236	5

Table 5 shows SI plus and SI negative and CI values SI Plus (Strength Index Plus): The values under "SI Plus" represent the strength or positive influence of each performance metric. These values indicate the positive impact or strength of each metric in contributing to a desirable outcome. Higher values in the "SI Plus" column suggest that a higher value of the corresponding metric is considered a strength or positive factor. SI Negative (Strength Index Negative): On the other hand, the values under "SI Negative" represent the weakness or negative influence of each performance metric. These values indicate the negative impact or weakness of each metric in contributing to a desirable outcome. Higher values in the "SI Negative" column suggest that a higher value of the corresponding metric is considered a weakness or negative factor. CI (Composite Index): The weights used in the combination might be determined by the importance or priority assigned to each metric. The formula for calculating the "CI" value for each metric might look like: CI=(SIPlus)−(SINegative) This formula suggests that the composite index is obtained by subtracting the negative influence from the positive influence. Higher "SI Plus" values indicate stronger positive contributions or strengths of the corresponding metrics. Higher "SI Negative" values indicate stronger negative contributions or weaknesses of the corresponding metrics. The "CI" values provide a composite measure, considering both positive and negative aspects. Positive "CI" values suggest an overall positive impact, while negative values suggest an overall negative impact.

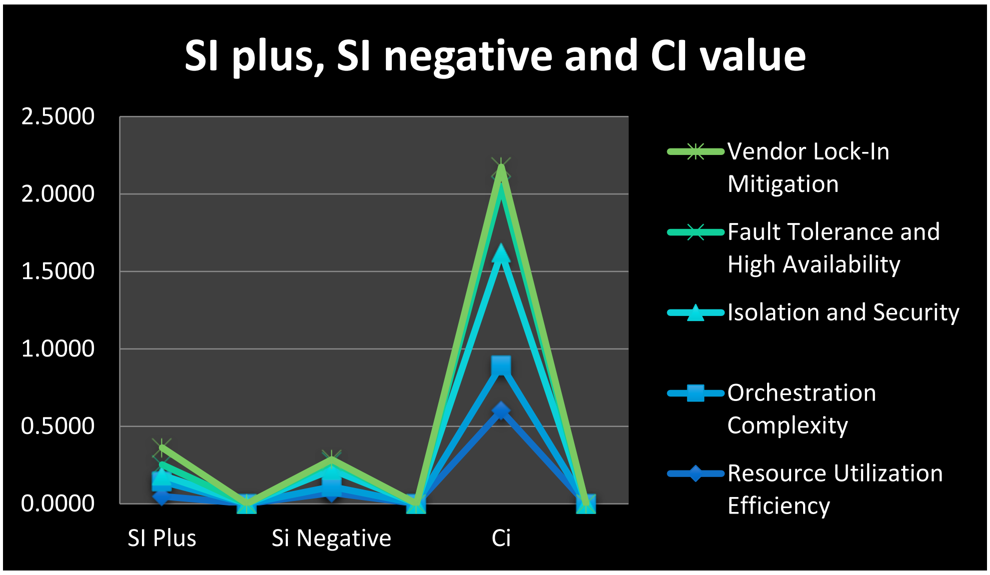


FIGURE 4. Si plus and Si negative and Ci value

Figure 4 illustrate graphical representation of final result for Network Function Virtualization (NFV) for flexible network services si positive, si negative and ci value.

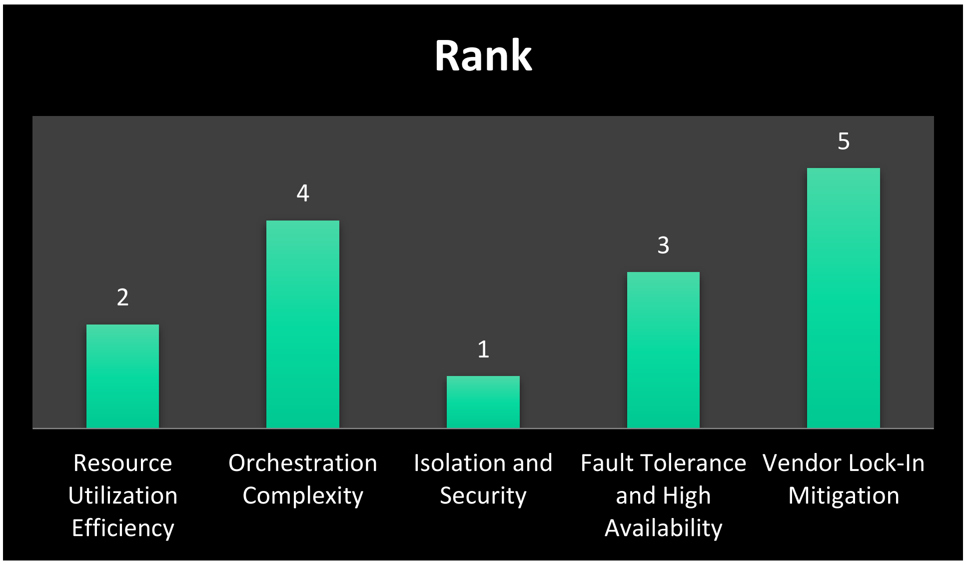


FIGURE 5. Rank

Figure 5 Shows the Rank for Network Function Virtualization (NFV) for flexible network services. Isolation and security is got the first rank and Vendor lock in mitigation is having the lowest rank.

4. CONCLUSION

Network Function Virtualization (NFV) represents a pivotal shift in networking paradigms, offering unparalleled flexibility and efficiency. This paradigm shift not only optimizes costs by eliminating the need for specialized hardware but also accelerates service innovation. Despite challenges like security concerns and interoperability, the industry is moving toward a more software-driven, cloud-native future. NFV's integration with technologies like Software-Defined Networking (SDN) and 5G further solidifies its role in reshaping network architectures, ensuring a responsive, scalable, and future-ready foundation for flexible network services.

REFERENCES

[1]. Li, Yong, and Min Chen. "Software-defined network function virtualization: A survey." IEEE Access 3 (2015): 2542-2553.
[2]. Hawilo, Hassan, Manar Jammal, and Abdallah Shami. "Network function virtualization-aware orchestrator for service function chaining placement in the cloud." IEEE Journal on Selected Areas in Communications 37, no. 3 (2019): 643-655.
[3]. Son, Jungmin, TianZhang He, and Rajkumar Buyya. "CloudSimSDN-NFV: Modeling and simulation of network function virtualization and service function chaining in edge computing environments." Software: Practice and Experience 49, no. 12 (2019): 1748-1764.
[4]. Kim, Hyuncheol, Seunghyun Yoon, Hongseok Jeon, Wonhyuk Lee, and Seungae Kang. "Service platform and monitoring architecture for network function virtualization (NFV)." Cluster Computing 19 (2016): 1835-1841.
[5]. Karimzadeh-Farshbafan, Mohammad, Vahid Shah-Mansouri, and Dusit Niyato. "A dynamic reliability-aware service placement for network function virtualization (NFV)." IEEE Journal on Selected Areas in Communications 38, no. 2 (2019): 318-333.
[6]. Rankothge, Windhya, Jiefei Ma, Franck Le, Alessandra Russo, and Jorge Lobo. "Towards making network function virtualization a cloud computing service." In 2015 IFIP/IEEE International Symposium on Integrated Network Management (IM), pp. 89-97. IEEE, 2015.
[7]. Moens, Hendrik, and Filip De Turck. "VNF-P: A model for efficient placement of virtualized network functions." In 10th international conference on network and service management (CNSM) and workshop, pp. 418-423. IEEE, 2014.
[8]. Mirjalily, Ghasem, and Zhiquan Luo. "Optimal network function virtualization and service function chaining: A survey." Chinese Journal of Electronics 27, no. 4 (2018): 704-717.
[9]. Akyildiz, Ian F., Shih-Chun Lin, and Pu Wang. "Wireless software-defined networks (W-SDNs) and network function virtualization (NFV) for 5G cellular systems: An overview and qualitative evaluation." Computer Networks 93 (2015): 66-79.
[10]. Hmaity, Ali, Marco Savi, Francesco Musumeci, Massimo Tornatore, and Achille Pattavina. "Virtual network function placement for resilient service chain provisioning." In 2016 8th International Workshop on Resilient Networks Design and Modeling (RNDM), pp. 245-252. IEEE, 2016.
[11]. Kouchaksaraei, Hadi Razzashi, Sevil Dräxler, Manuel Peuster, and Holger Karl. "Programmable and flexible management and orchestration of virtualized network functions." In 2018 European Conference on Networks and Communications (EuCNC), pp. 1-9. IEEE, 2018.
[12]. Behzadian, Majid, S. KhanmohammadiOtaghsara, MortezaYazdani, and Joshua Ignatius. "A state-of the-art survey of TOPSIS applications." Expert Systems with applications 39, no. 17 (2012): 13051-13069.https://doi.org/10.1016/j.eswa.2012.05.056
[13]. Salih, Mahmood M., B. B. Zaidan, A. A. Zaidan, and Mohamed A. Ahmed. "Survey on fuzzy TOPSIS state-of-the-art between 2007 and 2017." Computers & Operations Research 104 (2019): 207-227.https://doi.org/10.1016/j.cor.2018.12.019
[14]. Shukla, Atul, Pankaj Agarwal, R. S. Rana, and Rajesh Purohit. "Applications of TOPSIS algorithm on various manufacturing processes: a review." Materials Today: Proceedings 4, no. 4 (2017): 5320-5329.https://doi.org/10.1016/j.matpr.2017.05.042
[15]. Opricovic, Serafim, and Gwo-HshiungTzeng. "Compromise solution by MCDM methods: A comparative analysis of VIKOR and TOPSIS." European journal of operational research 156, no. 2 (2004): 445-455.https://doi.org/10.1016/S0377-2217(03)00020-1
[16]. Jahanshahloo, Gholam Reza, F. HosseinzadehLotfi, and Mohammad Izadikhah. "An algorithmic method to extend TOPSIS for decision-making problems with interval data." Applied mathematics and computation 175, no. 2 (2006): 1375-1384.https://doi.org/10.1016/j.amc.2005.08.048
[17]. Kuo, Ting. "A modified TOPSIS with a different ranking index." European journal of operational research 260, no. 1 (2017): 152-160.https://doi.org/10.1016/j.ejor.2016.11.052
[18]. Shih, Hsu-Shih, Huan-JyhShyur, and E. Stanley Lee. "An extension of TOPSIS for group decision making." Mathematical and computer modelling 45, no. 7-8 (2007): 801-813.https://doi.org/10.1016/j.mcm.2006.03.023
[19]. Chen, Pengyu. "Effects of the entropy weight on TOPSIS." Expert Systems with Applications 168 (2021): 114186.https://doi.org/10.1016/j.eswa.2020.114186.