

**PROPOSED ENERGY AWARE ROUTING ALGORITHM FOR SOFTWARE  
DEFINED NETWORK**

**Arun Sadanand<sup>1</sup>, Isha Batra<sup>2\*</sup>, Arun Malik<sup>3</sup>**

<sup>1</sup>School of Computer Application, Lovely Professional University, Punjab.  
arunpalayad@gmail.com

<sup>2</sup>School of Computer Science and Engineering, Lovely Professional University, Punjab.  
\*isha.batra2487@gmail.com

<sup>3</sup>School of Computer Science and Engineering, Lovely Professional University, Punjab.  
arunmalikhisar@gmail.com

**Abstract**

Software defined networking enhances network flexibility by split the control plane and the data plane, revolutionizing legacy network architecture by detaching control logic from forwarding devices. While SDN offers increased flexibility and programmability, routing algorithms play a vital role in determining how packets are forwarded. Proper assignment of controllers and switches is essential for creating energy-efficient networks. This paper reviews current routing techniques in SDN and proposes an algorithm aimed at reducing energy consumption in the network. The algorithm is designed to ensure efficient server allocation across various optimal scales, considering specific demands and link capacities, with the objective of minimizing overall power usage. We also compare the performance of this method with two other energy-aware routing (EAR) techniques, using shortest path routing algorithms for evaluation.

**Keywords:** Energy aware routing, Optimization, Dynamic routing, IoT, Cloud computing

**1. Introduction**

Most of the fields like Industries, Infrastructure, Health care, Smart Home, Transportation, and Security Surveillance utilizes today's emerging technologies similar to IoT, Big Data, Cloud Computing, Fog computing [1]. The Industrial Internet of Things (IIoT) is gaining traction as a powerful communication framework within factory settings, driven by the growing adoption of IoT technologies [2]. Concurrent with the escalation of big data and M2M interactions [3], demands on factory settings become escalating, and new trends are continuously being integrated into IIoT communication traffic and data transmission [4]. As networks continue to expand, diverse data flows are emerging within industrial applications and smart factories. The path of data transmission is critical, ensuring effective communication and connectivity between various devices [5].

A routing algorithm determines the route a packet must take as it travels from the root node to its target node [6]. Traditional protocols like Routing Information Protocol [7], Open Shortest Path First [8] successfully run in last two decades although now Certain constraints,

such as Quality of Service and variations in network scale, must be considered. [9]. New frameworks integrating various digital infrastructures matching to wireless sensor systems also IoT have been introduced, along with examples of specific applications for each [10]. SDN has integrated multi-path with segment based routing for effective traffic optimization, reducing the disk space demands of IoT. Numerous studies have shown that SDN has a significant impact on the Internet of Things. These advancements have led to countless devices being connected to the internet; this has triggered a substantial rise in internet traffic, which subsequently results in issues such as cyber attacks like DoS, difficulties with network scalability, and poor service availability [10]. Software-Defined Networking [6] was integrated into the IoT ecosystem to address these challenges; It focuses on offering centralized management and distinguishing between control and data pathways. [11]. Proposed new frameworks combine various digital solutions for IoT and wireless sensor networks, along with specific applications. SDN has incorporated multi-path and segment-based routing to optimize network flow control, thereby lowering storage requirements for smart connected devices. Various studies have demonstrated that Software-Defined Networking significantly influences the Internet of Things. [12].

Our contributions include the following advancements in programmable networking

- Perform a comprehensive analysis of existing algorithms in programmable networking to determine the most effective routing paths.
- Design an algorithm to identify the most efficient path between the controller and other resources, emphasizing the optimization of resource utilization and minimizing energy consumption.

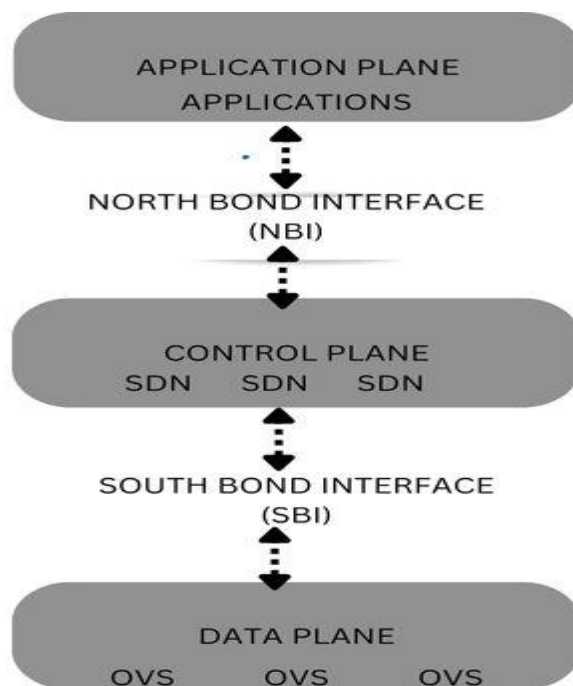
The structure of this paper is organized as follows: In Section II, a review of related works is provided. In Section III, the problem formulation related to the shortest path algorithm is discussed Section IV provides proposed shortest path algorithm for SDN. Section V covers performance evaluation of proposed algorithm. Finally paper ended with conclusion.

## **2. Related Work**

Software-Defined Networking (SDN) enables a revolutionary method to network design, providing increased flexibility, programmability, and control compared to traditional network systems. The core concept of SDN involves decoupling the management layer from the forwarding layer, allowing for centralized control of network devices and facilitating programmable networking. Above mentioned separation enables more responsive network management and adaptability, allowing real-time enforcement of network policies without altering each device individually [13].

Centralized controllers such as Open Daylight, ONOS, and Ryu have been widely adopted as essential components for SDN architecture, providing an interface to programmatically manage network flows and optimize resource utilization [14]. Early studies on SDN architecture have emphasized its advantages, including simplified network management, enhanced programmability, and the capacity for dynamic scaling and

configuration of networks. Kreutz et al. emphasized the flexibility of SDN, noting how centralized controllers enable granular control over traffic flows, allowing network administrators to optimize traffic management [15]. However, challenges like controller placement optimization, scalability, and security vulnerabilities were also discussed. In contrast to traditional networks, where routing decisions are distributed across individual devices, SDN consolidates routing decisions in the control plane. This centralized approach offers significant advantages but also presents unique challenges related to controller scalability and fault tolerance. For routing algorithms in SDN, the most widely used approach is shortest path routing. Controllers like OpenDaylight and ONOS utilize algorithms namely Dijkstra's to identify the path connecting Origin node to endpoint which is lowest, setting flow rules stored in switches to implement routing decisions [16]. The centralized control model in SDN makes implementing multipath routing more efficient compared to traditional networks, as the controller can monitor and adjust paths based on real-time traffic conditions. In more advanced implementations, dynamic flow scheduling algorithms have been proposed to enhance load balancing and optimize path utilization, especially in scenarios with fluctuating traffic patterns [17]. QoS-aware routing is another area that has gained attention in SDN. Many studies have focused on how SDN can be leveraged to ensure that specific traffic flows, such as video streaming or real-time voice communications, receive the necessary bandwidth and low-latency treatment to meet Quality of service (QoS) requirements.



**Figure 1.** Software defined Networking Architecture [15] commonly used basic version..

Centralized SDN controllers can enforce traffic prioritization policies across the entire network, ensuring that critical traffic receives priority over less sensitive data flows. Algorithms that manage bandwidth allocation, such as Max-Min Fairness and Weighted Fair Queuing, have been integrated into SDN platforms to achieve these QoS objectives [18].

Energy-efficient routing has also been explored in SDN to minimize energy usage networks, especially in data center environments. With SDN's centralized control, it is possible to implement energy-aware algorithms that selectively power down or idle underutilized switches and links during periods of low traffic [19]. This approach requires monitoring traffic patterns and rerouting traffic through fewer devices to save energy. Several studies have demonstrated the potential of SDN to improve energy efficiency without significantly compromising performance, further showcasing the advantages of SDN in large-scale network deployments.

Recent years have seen the rise of routing methods driven by machine learning techniques, applying artificial intelligence techniques to improve routing decisions in SDN. By analyzing historical network data and learning traffic patterns, machine learning algorithms can predict future network conditions and proactively adjust routing paths [20]. This proactive approach can significantly enhance performance by avoiding congestion before it occurs. Reinforcement learning has been particularly promising, where the SDN controller learns optimal routing policies over time based on continuous feedback from the network environment. Load balancing is a critical concern in SDN, particularly in cloud environments and data centers where traffic demands can vary dramatically. Numerous load-balancing algorithms have been introduced to achieve a more balanced distribution of traffic across the network. SDN's centralized control enables real-time monitoring of network congestion and utilization, enabling dynamic traffic distribution to avoid overloading any single path. Algorithms like Traffic-Aware Load Balancing (TALB) use this real-time data to adjust flow rules dynamically, optimizing resource utilization and ensuring better overall network performance [21].

Despite its benefits, SDN still faces significant challenges, particularly related to controller scalability and fault tolerance. As networks expand, a single centralized controller can become a bottleneck, and its failure could potentially disrupt the entire network. To tackle these challenges, research on distributed SDN architectures focuses on enabling multiple controllers to work together, providing scalability and redundancy. Security is also a key issue, as the centralized design of SDN controllers makes them appealing targets for attacks. Mitigating these vulnerabilities continues to be a critical focus in SDN research. [22].

SDN provides a flexible platform that enables advanced routing and network management strategies not feasible in traditional networking. Routing algorithms in SDN, from shortest path to machine learning-based approaches, demonstrate the potential for SDN to optimize performance, improve load balancing, ensure QoS, and reduce energy consumption [23]. However, issues related to controller scalability, security, and fault tolerance remain open areas of investigation, indicating that while SDN holds significant promise, further advancements are needed to fully realize its potential in large-scale network environments [24].

The study in [25] utilized the SARSA algorithm to enhance Quality of Service (QoS) within a layered SDN framework. For every incoming data flow, switch interacts with the controller to infer QoS needs implicitly. Based on these requirements, the system determines

the next hop, moving data from the source switch toward the destination. QoS metrics, including delay, data loss and transmission rate are utilized to specify requirements, the reward function is formulated as a weighted combination of these metrics. In [26], it is necessary for the network administrator to configure various parameters specific to each traffic type. For example, real-time multimedia flows demand a routing method that minimizes delay, maximizes throughput, and reduces jitter.

According to the author in [26], an intelligent routing algorithm for Software-Defined Networking should consider the security requirements and data throughput demands of every service provided by the network. In [28], the GA algorithm is proposed for a resource distribution pool with varying demands. This GA implementation utilizes single-location crossover, single-location mutation, and probabilistic selection, applying the coefficient of variation across a set of KKK servers with fluctuating workloads to define the fitness function with the aim of reducing the average coefficient of variance as the performance objective.

The author in [29] presents a routing algorithm designed to dynamically adjust the path within active traffic without downtime and develops a tabular-based learning method for flow management in Software-Defined Networking. Meanwhile, the author [30] proposes a pathway model for extensive networks organized into multi-step clusters, which improves energy efficiency performance.

The author [31] proposes SADFIR routing protocol to manage heterogeneity awareness and application sensitivity of sensed application. To avoid computational overhead from consistent periodic reconfiguration, delays are extended up to the tenth round. Architecture proposed in the article [31] utilize multi-path inter cluster communication to existing type of environmental report.

The author in [32] presents a Machine Learning-driven and Software-Controlled approach to path selection, where ML utilizes link state information to identify optimal routes for smart traffic management. The RL routing employs Q-learning to develop a Q-function that consists of a sequence of steps. The author in [33] compares the RSIR algorithm with Dijkstra's algorithm, showing that the RSIR algorithm outperforms Dijkstra's in performance results. The authors in [34] propose employing machine learning methods, including deep networks and adaptive learning, to improve path selection decisions. While these proposals are not applied in Software-Defined Networking (SDN), they explore routing strategies within distributed environments. However, signaling overhead can lead to network congestion as a limitation. Authors in [35] utilize routing strategies that combine machine learning with Software-Defined Networking. Network programmability enables the integration of machine learning into routing solutions, whereas traditional protocols tend to select the same path when similar traffic patterns reoccur.

### 3. Research Gaps

The Dijkstra algorithm is among the first methods developed to calculate the shortest path between two nodes in a network. The Open Shortest Path First (OSPF) algorithm primarily relies on Dijkstra's algorithm. Both serve as fundamental algorithms for shortest path calculations; however, they do not account for network energy efficiency. Considering dynamic routing with multiple links between nodes is essential. Combining dynamic routing with a control plane configuration forms the basis of Software-Defined Networking. In this approach, adapting to dynamic traffic patterns requires fewer active nodes than previous methods, thereby reducing power usage by prioritizing higher traffic efficiency within the network. In Software-Defined Networking, the number of hops is minimized by employing an algorithm that defines connection capacities to handle varying throughput demands across different network flows. Energy-efficient shortest paths are impacted by the quantity and positioning of controllers throughout the network infrastructure. As number of controllers rises, total power usage also increases, as a higher count of controllers necessitates additional switches to connect two network entities. The average processing time of a switch in its queue buffer is considered as an indicator of the switch's energy consumption.

High-speed Software-Defined Networks may require managing numerous controllers and servers across various data centers. In cases involving a substantial number of resources and connections, the resulting high power consumption of these data centers becomes a significant concern.

- Operating costs increase when switches and nodes are deployed inefficiently. Traditional routing algorithms lack mechanisms to limit network power consumption by finding optimal flow paths. They do not support dynamic routing, which can adapt flow routes based on changing QoS requirements.

- Furthermore, energy-conscious routing algorithms emphasize efficient network monitoring and seek to minimize connections linking switches with controllers. However, there remains a gap in energy-aware routing approaches to fully address energy efficiency, resource utilization, load balancing, and similar factors.

### 4. Proposed Shortest Path Algorithm For SDN

In this approach, the route length is measured by the hop count connecting the origin and destination [30]. The methodology includes restricting active link usage to lower power consumption. Consequently, the physical distance between nodes is proportional to the shortest path, the connection weight, and inversely related to link capacity. To meet our objectives, an initial set of routes is generated, and optimal controller positioning is determined. The Capacity-based CPP strategy [31] is employed to allocate controllers and their associated switches. The management path linking each forwarding node with its assigned controller is established as a preliminary step. This process begins by generating a connected graph with equal weights.

The path finding process in Algorithm utilizes an adapted form of Dijkstra's algorithm [30], which is commonly employed to determine the best route between two nodes in a network. This adaptation is refined to execute as quickly as possible while preserving high-quality results. Our approach seeks to attain an almost optimal outcome with minimal processing complexity, recognizing that runtime significantly impacts the reaction time of SDN control units in meeting system demands. After variable link costs are calculated in the initial step, algorithm selects the shortest or least-cost path based on link costs determined during topology discovery. The evaluation metric applied is the utilized bandwidth of the path, represented as  $E = f_{ij}(GNF) - U_{e,f}$ . As a result, the shortest path minimizes power consumption across nodes and links. Furthermore, Utilizing multi-path routing helps decrease overall delay for data flow across different service infrastructures. For routing solutions, the feasible disjoint paths must ensure both Resilience. Moreover, the network layout enforces maximum response time limits, restricting unauthorized network services and managing management data over these connections. Nonetheless, management data solely passes via the coordinator linked to the switch, aside from the source and target coordinators.. If a node loses connection with its controller, an additional control message is dispatched by the controller to the next one, notifying it of the forwarding rule to apply to the node within its jurisdiction. The SDN controller continuously gathers information on the network's framework and conducts path scans for every control stream within its environment.

Algorithm: To select shortest path

**Input:**

- G: Graph defined by [N, E]
- N: Set of network nodes
- E: Set of weighted edges and links
- fr(di): Flow requested by the data plane
- K: Number of controllers
- $C$ : Controller capacity
- $E$ : Set of edges

**Steps:****1. Calculate Initial Costs:**

- Compute the cost to neighbors on the weighted graph G based on the current traffic flow between nodes  $\lambda_{ij}$  and the link capacity  $C_E$ .

**2. Iterate Over All Nodes (N):**

- For each node i, find the minimum cost paths  $e$  and  $f$  for the flow requested by the data plane fr(di) and the overall network flow  $f_{ij}(GNF)$ .

**3. Handle Non-Neighbor Nodes (NNE):**

- For each node i in the set of non-neighbor nodes:

○ If the current distance between nodes  $i$  and  $j$  ( $D_{ij}$ ) is greater than the calculated cost of paths  $_e$  and  $_f$ , and there is a link between  $i$  and  $j$  ( $f_i, j_g = 1$ ):

▪ Update the minimum cost between nodes  $i$  and  $j$  as:  
 $\min_{ij} = f_{Eg} + f_{ij}(\text{GNF})$ .

**4. Check Link Existence:**

○ If no link exists between nodes  $i$  and  $j$  ( $f_i, j_g = 0$ ), skip this step and continue.

**5. For Each Active Node (i):**

○ Compute the previous hop ( $\text{PreviousHop}(i) = i$ ) and next hop ( $\text{NextHop}(i) = i$ ) for routing.

**6. Compute the Cost Metric:**

○ Calculate the cost metric  $C(e_{ij})$  for all edges between nodes  $i$  and  $j$  in the edge set  $E$ , based on the traffic flow  $f_{ij}(d)$  and link status  $e_{ij}^L$ .

**7. Check Path Length:**

○ If the length of the path is zero ( $\text{length}(\text{path}) = 0$ ), store the current shortest distance in a temporary array  $\text{tmp}[\text{distance}]$ .

**8. Select the Closest Node:**

○ Begin with the node closest to the source, using  $[t, u] = \min(\text{tmp})$ .

**9. Evaluate Neighbor Nodes:**

○ For each neighbor of the current node  $i$ , if the following condition is met:

$$\text{CostMatrix}(i, j) + D_j < D_i$$

(i.e., if the cost from node  $i$  to node  $j$  through its neighbor is less than the current known distance), update the shortest path distance as:

$$\text{distance}(\text{destination}) = \text{distance}(\text{source}) + \text{CostMatrix}(\text{source}, \text{destination}).$$

**10. Update Shortest Path:**

○ When a shorter path is found, update the shortest distance accordingly to optimize the path continuously.

“The shortest path algorithm”, an adaptation of Dijkstra’s algorithm [50], determines the minimum distance route between two points. In Software-Defined Networks, The controller’s processing time is directly proportional to execution duration. In the first stage, dynamic cost is calculated using the utilized bandwidth of the route as the evaluation parameter.

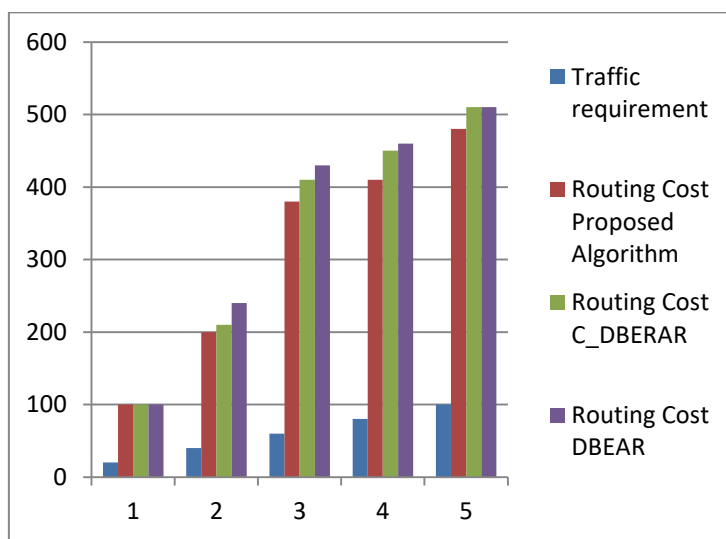
## 5. Result and Discussion

A performance analysis is conducted on a large network consisting of 100 nodes and 460 links to assess the optimization of the proposed algorithm. The efficiency of our approach reviewed against two other “Energy-Aware Routing (EAR)” methods employing shortest path routing algorithms: DBEAR, illustrating the traditional shortest path algorithm designed for distance-centric, energy-efficient routing based on the classic “Dijkstra algorithm” [32],

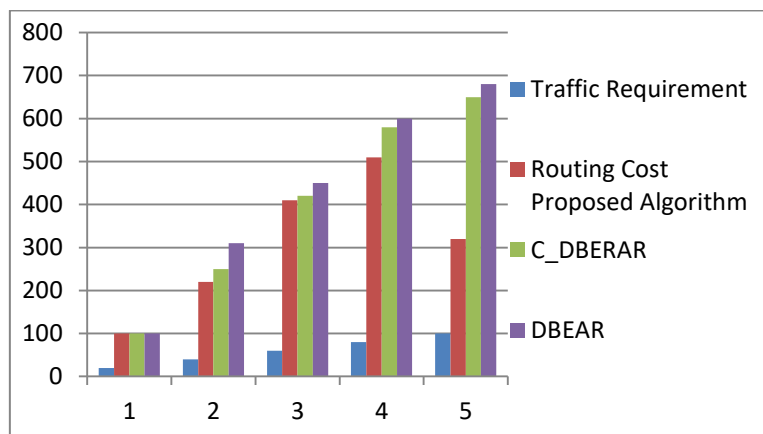
alongside C\_DBEAR, a constrained distance-based EAR incorporating capacity restrictions [33].

To analyze and determine the efficiency and reliability of the proposed algorithm against chosen algorithms within a specific network topology and load, energy-aware routing typically improves utilization and lowers costs by generating routes tailored to specific demands, thereby reducing the flow routing process between active nodes in the network.

For evaluating average energy cost, two types of networks are utilized: one containing 25 nodes and 80 links, and the other comprising 54 nodes and 170 links.



**Figure 1a.** Path cost for small scale network

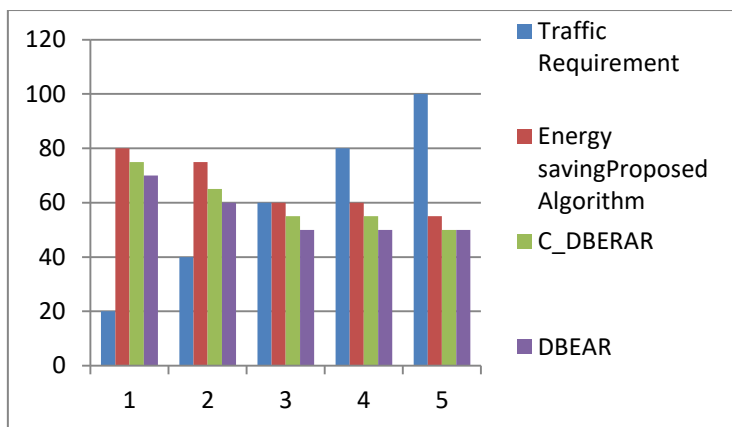


**Figure 1b.** Path cost for large scale network

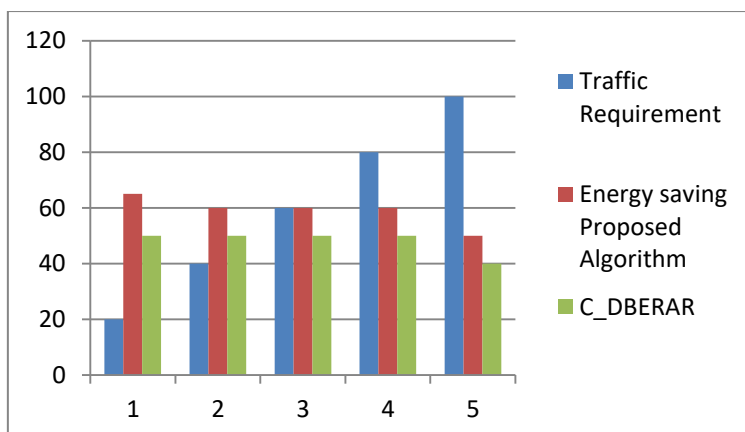
The routing cost based on demand for three algorithms is shown in Fig. 1a and 1b, representing small and large-scale networks. For the small-scale network, the shortest path is used between controllers and switches. In contrast, as network size and traffic demand increase, the overall routing cost rises. Our proposed algorithm clearly surpasses the other two algorithms in large-scale networks by minimizing the count of active nodes within the network.

**Energy Saving by Algorithm**

A key factor is the energy-saving capability of the algorithm used in SDN. Energy savings are achieved by minimizing the total cost through the reduction of active nodes, consolidating network resources at critical points.



**Figure 2a.** Energy saving for small scale network



**Figure 2b.** Energy saving for large scale network

It restricts power consumption by optimizing the placement of controllers and links across all computing nodes. 5a, 5b, and 5c named figures show the differences across two distinct network sizes, indicating that our proposed algorithm outperforms the other two algorithms, reaching up to 80% energy savings for a 25-node requirement. For a large-scale network scenario, energy savings reach 65%, with the proposed algorithm still outperforming the other algorithms.

**6. Conclusion**

This article reviews current energy-aware routing techniques in Software-Defined Networking. The shortest path and connection weight are directly related to link capacity, limited link ability, and restricted network links for traffic. The shortest path algorithm determines the optimal route from one node to another for controller-switch mapping. The proposed energy-aware routing model enhances network energy efficiency by combining

resource management and resilience. The simulation outcomes indicate that the proposed algorithm surpasses other methods across various network scales, achieving notable energy efficiency by incorporating both resilience and resource optimization.

### References

1. Wei Xiang, Ning Wang, and Yuan Zhou, "An Energy-Efficient Routing Algorithm for Software-Defined Wireless Sensor Networks", *IEEE Sensors Journal*, 16(20) 016, pp: 7373-7400
2. Khan, W. Z., Rehman, M. H., Zangoti, H. M., Afzal, M. K., Armi, N., & Salah, K. (2020). Industrial internet of things: Recent advances, enabling technologies and open challenges. *Computers & Electrical Engineering*, 81, 106522. doi:10.1016/j.compeleceng.2019.
3. K. Begum and S. Dixit, "Industrial WSN using IoT: A survey," in *International Conference on Electrical, Electronics, and Optimization Techniques*, 2016, pp. 499-504.
4. Malik, P. K., Sharma, R., Singh, R., Gehlot, A., Satapathy, S. C., Alnumay, W. S., ... Nayak, J. (2020). Industrial Internet of Things and its applications in industry 4.0: State of the art. *Computer Communications*. doi:10.1016/j.comcom.2020.11.016
5. Qiu, T., Chi, J., Zhou, X., Ning, Z., Atiquzzaman, M., & Wu, D. O. (2020). Edge Computing in Industrial Internet of Things: Architecture, Advances and Challenges. *IEEE Communications Surveys & Tutorials*, 1–1. doi:10.1109/comst.2020.3009103
6. N. Feamster, J. Rexford and E. Zegura, "The road to SDN: an intellectual history of programmable networks," *ACM Sigcomm Computer Communication Review*, vol. 44, no. 2, pp. 87-98, 2014
7. M. Banikazemi, D. Olshefski, A. Shaikh, and J. Tracey, "Meridian: an SDN platform for cloud network services," *IEEE Communications Magazine*, vol. 51, no. 2, pp. 120-127, 2013
8. F. Fernando Jurado-Lasso; Ken Clarke; Andres Navarro Cadavid; Ampalavanapillai Nirmalathas; (2021). Energy-Aware Routing for Software-Defined Multihop Wireless Sensor Networks. *IEEE Sensors Journal*, (), -. doi:10.1109/jsen.2021.3059789
9. D. M. Casas-Velasco, O. M. C. Rendon and N. L. S. da Fonseca, "Intelligent Routing Based on Reinforcement Learning for Software-Defined Networking," in *IEEE Transactions on Network and Service Management*, vol. 18, no. 1, pp. 870-881, March 2021, doi: 10.1109/TNSM.2020.3036911.
10. Y. -R. Chen, A. Rezapour, W. -G. Tzeng and S. -C. Tsai, "RL-Routing: An SDN Routing Algorithm Based on Deep Reinforcement Learning," in *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 4, pp. 3185-3199, 1 Oct.-Dec. 2020, doi: 10.1109/TNSE.2020.3017751.
11. L. Zhao, J. Li, A. Al-Dubai, A. Y. Zomaya, G. Min and A. Hawbani, "Routing Schemes in Software-Defined Vehicular Networks: Design, Open Issues and Challenges," in *IEEE Intelligent Transportation Systems Magazine*, vol. 13, no. 4, pp. 217-226, winter 2021, doi: 10.1109/MITS.2019.2953557.

12. Rafique, W., Qi, L., Yaqoob, I., Imran, M., Rasool, R. ur, & Dou, W. (2020). Complementing IoT Services through Software Defined Networking and Edge Computing: A Comprehensive Survey. *IEEE Communications Surveys & Tutorials*, 1–1. doi:10.1109/comst.2020.2997475
13. O. Ahmed, F. Ren, A. Hawbani and Y. Al-Sharabi, "Energy Optimized Congestion Control-Based Temperature Aware Routing Algorithm for Software Defined Wireless Body Area Networks," in *IEEE Access*, vol. 8, pp. 41085-41099, 2020, doi: 10.1109/ACCESS.2020.2976819.
14. J. Rischke, P. Sossalla, H. Salah, F. H. P. Fitzek and M. Reisslein, "QR-SDN: Towards Reinforcement Learning States, Actions, and Rewards for Direct Flow Routing in Software-Defined Networks," in *IEEE Access*, vol. 8, pp. 174773-174791, 2020, doi: 10.1109/ACCESS.2020.3025432.
15. Y. -R. Chen, A. Rezapour, W. -G. Tzeng and S. -C. Tsai, "RL-Routing: An SDN Routing Algorithm Based on Deep Reinforcement Learning," in *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 4, pp. 3185-3199, 1 Oct.-Dec. 2020, doi: 10.1109/TNSE.2020.3017751.
16. Rahouti, Mohamed, KaiqiXiong, and Yufeng Xin. "Secure software-defined networking communication systems for smart cities: Current status, challenges, and trends." *IEEE Access* 9 (2020): 12083-12113.ss
17. Latif, Zohaib, et al. "A comprehensive survey of interface protocols for software defined networks." *Journal of Network and Computer Applications* 156 (2020): 102563.
18. Shaghghi, Arash, et al. "Software-defined network (SDN) data plane security: issues, solutions, and future directions." *Handbook of Computer Networks and Cyber Security: Principles and Paradigms* (2020): 341-387.dd
19. Alzahrani, Abdulsalam O., and Mohammed JF Alenazi. "Designing a network intrusion detection system based on machine learning for software defined networks." *Future Internet* 13.5 (2021): 111.
20. Ahuja, Nisha, et al. "Automated DDOS attack detection in software defined networking." *Journal of Network and Computer Applications* 187 (2021): 103108.
21. Lv, Zhihan, and Neeraj Kumar. "Software defined solutions for sensors in 6G/IoE." *Computer Communications* 153 (2020): 42-47.
22. Chica, Juan Camilo Correa, Jenny CuatindioyImbachi, and Juan Felipe Botero Vega. "Security in SDN: A comprehensive survey." *Journal of Network and Computer Applications* 159 (2020): 102595.
23. Alam, Iqbal, et al. "A survey of network virtualization techniques for Internet of Things using SDN and NFV." *ACM Computing Surveys (CSUR)* 53.2 (2020): 1-40.
24. Chen, Yi-Ren; Rezapour, Amir; Tzeng, Wen-Guey; Tsai, Shi-Chun (2020). RL-Routing: An SDN Routing Algorithm Based on Deep Reinforcement Learning. *IEEE Transactions on Network Science and Engineering*, (), 1–1. doi:10.1109/TNSE.2020.3017751. doi:10.1109/access.2020.3025432
25. Sufiev, H.; Haddad, Y.; Barenboim, L.; Soler, J. Dynamic SDN controller load balancing. *Future Internet* 2019,11, 75. [CrossRef]

26. Ouhab, A., Abreu, T., Slimani, H., & Mellouk, A. (2020). Energy-efficient clustering and routing algorithm for large-scale SDN-based IoT monitoring. ICC 2020 - 2020 IEEE International Conference on Communications (ICC). doi:10.1109/icc40277.2020.9148659
27. Chou, L.D.; Yang, Y.T.; Hong, Y.M.; Hu, J.K.; Jean, B. A genetic-based load balancing algorithm in OpenFlow network. In *Advanced Technologies, Embedded and Multimedia for Human-Centric Computing*; Springer: Berlin/Heidelberg, Germany, 2014; pp. 411–417.
28. L. Zhao *et al.*, "Novel Online Sequential Learning-Based Adaptive Routing for Edge Software-Defined Vehicular Networks," in *IEEE Transactions on Wireless Communications*, vol. 20, no. 5, pp. 2991-3004, May 2021, doi: 10.1109/TWC.2020.3046275.
29. Daniela M -Velasco, *Graduate Student Member, IEEE* Oscar Mauricio CaicedoRendon , Senior Member, IEEE, and Nelson L. S. da Fonseca , Senior Member, IEEE *Intelligent Routing Based on Reinforcement Learning for Software-Defined Networking IEEE TRANSACTIONS ON NETWORK AND SERVICE MANAGEMENT, VOL. 18, NO. 1, MARCH 2021*
30. R. Arroyo-Valles, R. Alaiz-Rodriguez, A. Guerrero-Curieses, and J. Cid- Sueiro, "Q-probabilistic routing in wireless sensor networks," in *Proc. Int. Conf. Intell. Sens. Sens. Netw. Inf. (ISSNIP)*, Melbourne, VIC, Australia, 2007, pp. 1–6.
31. Forster and A. L. Murphy, "FROMS: Feedback routing for optimizing multiple sinks in WSN with reinforcement learning," in *Proc. Int. Conf. Intell. Sens. Sens. Netw. Inf. ISSNIP*, Melbourne, VIC, Australia, 2007, pp. 371–376.
32. Reddy, S.V.P.S. and Kumar, M.V. 2024. A New Hybrid Energy Storage System for Electric Vehicle Drive System. *Engineering, Technology & Applied Science Research*. 14, 6 (Dec. 2024), 18855–18861. DOI:<https://doi.org/10.48084/etasr.9010>.
33. N. Kato , "The deep learning vision for heterogeneous network traffic control: Proposal, challenges, and future perspective," *IEEE Wireless Commun.*, vol. 24, no. 3, pp. 146–153, Jun. 2017.