2024 4th International Conference of Science and Information Technology in Smart Administration (ICSINTESA) | 979-8-3503-7611-1/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICSINTESA62455.2024.10747989

Optimizing Self-Learning Forwarding Strategies in Vehicular Named Data Network through Protocol Simplification

Fitra Nur Hanif, Leanna Vidya Yovita, Istikmal
Telkom University
Bandung, Indonesia
fitrahanif02@gmail.com

***Abstract*—The Information Centric Network (ICN) concept gave rise to the Named Data Network (NDN), a content-centric computer network architecture. Vehicular ad-hoc Networks (VANETs), characterized by high mobility, have embraced NDN, employing broadcast-based forwarding strategies. This self-learning approach enables adaptive path adjustments without relying on explicit routing instructions, a valuable trait in dynamic wireless environments. However, node movements are interpreted as link failures, prompting nodes to issue Negative Acknowledgments (NACKs) to consumers. Yet, due to changes in node positions, NACKs might lose their way back to the intended consumers, leading to increased transmission overhead. To address these issues, this paper proposes a streamlined communication protocol for self-learning forwarding strategies. The suggested approach mitigates the overhead associated with negative acknowledgments, thereby reducing the delay in retransmitting failed packets. Simulations conducted in ndnSIM 2.8 compare the performance of this scheme against existing strategies (such as multicast, multicast VANET, and self-learning) across scenarios with varying numbers of nodes. Results indicate that the proposed scheme achieves a lower round-trip time (approximately 12.89%) and a higher throughput (around 16.02%) compared to the default self-learning approach as the number of nodes increases. This improvement stems from the reduction in negative acknowledgment protocol overhead within the self-learning forwarding strategy, enabling more efficient retrieval of successful packets.**

***Index Terms*—named data network, vehicular network, forwarding strategy, self learning, Negative Acknowledgments (NACKs)**

I. Introduction

The Information Centric Network (ICN) paradigm has shown resilience over the years as it evolved from location-oriented computer networks to a future layer of computer networks based on content known as the Named Data Network (NDN) [1]. The hierarchy-based architecture of the NDN system allows matching with information that has the requested name as a prefix, which in turn helps in verifying the relationship between the hierarchical information name and the corresponding information object [2]. In NDN, there are two types of packets exchanged over the network, namely interest packets and data packets [3]. These two messages are signed by the consumer as the one requesting the content and the provider in charge of delivering the content among consumers or intermediate nodes using data packets [4]. Interest packets are triggered by the consumer to request specific data, whereas data packets containing the requested information are sent in response to these requests [5]. The three main types [2] of data structures in NDN routers : include the Forwarding Information Base (FIB) which is routing information based on prefix names, the Pending Interest Table (PIT) which is a table used to manage information about interests that are waiting to be fulfilled by data packets and the Content Store (CS) which is a local cache that stores copies of data that has been requested and cached by nodes for later reuse. This approach enables caching within the network as well as replication of content, which in turn makes it easier to deliver information efficiently and in a timely manner [6]. An aspect of the communication model in Named Data Networking (NDN) is to request the remaining chunks of the network by name, which can be retrieved from intermediate nodes if a similar request has been made previously and the associated data has been cached, or directly from the producer [7]. It overcomes the routing inefficiencies that occur in current Internet Protocol (IP)-based networks with regard to NDN's goal of providing efficient data dissemination by focusing on content rather than on host identity and location [8].

Inspired by NDN, the Vehicular ad-hoc Network VANET communication architecture has been modified to adopt NDN directly [5]. Many studies have proposed integrated solutions for VANETs with NDNs involving optimized forwarding strategies [8] , [9]. Vehicular Ad-hoc Network (VANET) is a spontaneously formed network among vehicles and other connecting devices, which communicate with each other through wireless media to exchange useful information [10]. However, nodes in VANETs have higher mobility speeds, presenting significant communication challenges [5]. In vehicular ad hoc networks [8], vehicles not only act as information users, but also as information sending and receiving points. However, since vehicles have a limited transmission range, they can only communicate directly with vehicles or nodes within a certain distance, following the end-to-end communication architecture of the traditional transmission control/internet protocol (TCP/IP). Therefore, the implementation of NDN in a vehicular environment benefits from the broadcast nature for data transmission. Broadcast mode is suitable for VANETs where vehicles move at high speeds [10].

The broadcast-based self-learning method, introduced [11] is a broadcast-based adaptive approach where every node in the network receives information. This method scales up [12] to understand paths without specific instructions, beneficial in dynamic wireless networks.

Other research [13], In MANETs with self-learning, NACK delivery faces costs due to potential mobility losses, causing delays and increased transmission overhead, including message exchange protocols. To address this, this paper suggests implementing a Data Aggregation Forwarding (DAF) strategy in NDNs to reduce latency and overhead. The data-centric approach of NDN along with DAF shows advantages over IP, especially in scenarios involving mobility and multicast communication in MANETs. In the default multicast strategy, NACK messages are sent when a particular name is unreachable which leads to flooding the network with NACK packets due to constant changes in the network topology [14], this causes the data message cannot be received between neighboring nodes even though the NACK has been processed first.

Therefore, this research adopts the multicast-vanet forwarding strategy to overcome the above problems that occur in the self-learning forwarding strategy. This study evaluates the performance of Vehicular Ad Hoc Network (VANET) in Named Data Networking (NDN) architecture by considering the minimum value of Round-Trip Time by reducing the protocol complexity, so that the connection between consumers and producers can be more stable. In addition, it also considers increasing the throughput value.

II. RELATED WORK

The application of NDN in vehicular environments benefits from the broadcasting nature for data transmission. In a VANET environment, where vehicles are moving at high speed, broadcasting mode is suitable for use [8]. Nonetheless, prior research [15] has emphasized a mapping mechanism linking Pending Interest Table (PIT) entries with layer-2 addresses, facilitating the unicast delivery of data at the layer-2 level. This advancement enables the proposed scheme to ascertain the optimal timing for transmitting data packets either through unicast or broadcast methods.

A broadcast-based adaptive forwarding strategy [11], and improved [12], termed self-learning, entails that each node within the network obtains information. Self-learning involves nodes initially expressing interest to learn routes through broadcasts, and upon receiving data, they ascertain which routes carry content with suitable prefixes. In addition, self-learning allows nodes in a wired network to send a NACK (Negative Acknowledgement) back to the consumer, similar to RRER (Route Error) in Ad hoc On-Demand Distance Vector (AODV), the AODV protocol is a single-path protocol. When a link failure occurs, the routing request process starts again.

According to research results [13], NACK delivery in MANETs with a self-learning approach is considered costly as NACKs can get lost due to mobility, causing delays in reaching consumers. This can increase the transmission overhead, similar to message exchange protocols. The paper introduces the Data Aggregation Forwarding (DAF) strategy in NDN to reduce latency and transmission overhead. The data-centric approach of NDN and the DAF strategy show advantages over IP, particularly in the context of mobility and multicast communication in MANETs.

The research [14] conducted for NACK-related multicast VANET evaluates the impact of excessive NACK messages on the network due to continuous topology changes, which causes difficulties in data message reception. The framework includes multicast-vanet forwarding strategies to address these issues and optimize data delivery in VANETs.

III. SYSTEM MODELLING

In this section, we will describe the main components and their respective roles in the proposed system model. These components play an important role in carrying out the various stages described earlier.

A. Vehicular Ad-hoc Named Data Network

In the application of NDN to VANET, some source nodes can act as data providers [5]. The caching system within NDN is pivotal in minimizing both end-to-end delay and network congestion. Data bearing identical names can be stored in various cache servers (CS), providing accessibility to other nodes [16]. In a wireless VANET environment, neighboring vehicles may listen and store unsolicited information into their cache [17]. Thus, when another vehicle needs the information, it can retrieve it from the local cache without having to access the original source [18].

B. Forwarding on Named Data Network

Interest packets received by nodes in a forwarding strategy on the NDN can take action to be fulfilled, discarded, or forwarded in the order of checking the Content Store (CS), Pending Interest Table (PIT), and Forwarding Information Base (FIB) [19]. The forwarding steps in NDN are described [20], the Content Store (CS) caches data that can be directly forwarded to consumers without re-requesting the source. The Pending Interest Table (PIT) records pending and serviced interests. If an interest has no associated content, it is forwarded to another node. Pending interests can be removed from the PIT. Forwarding Information Base (FIB) contains routing information based on prefix name to determine the outgoing face in one-hop forwarding. If there is no matching FIB entry, the interest is forwarded through all exit faces of the node.

C. On Broadcast-based self-learning forwarding strategy

Broadcast-based self-learning is an approach described [11] that when a data packet is first sent on an unknown path, it is broadcast to the entire network. If there is a response from the destination, a new entry will be created in the forwarding table to route subsequent packets directly to the destination using unicast.

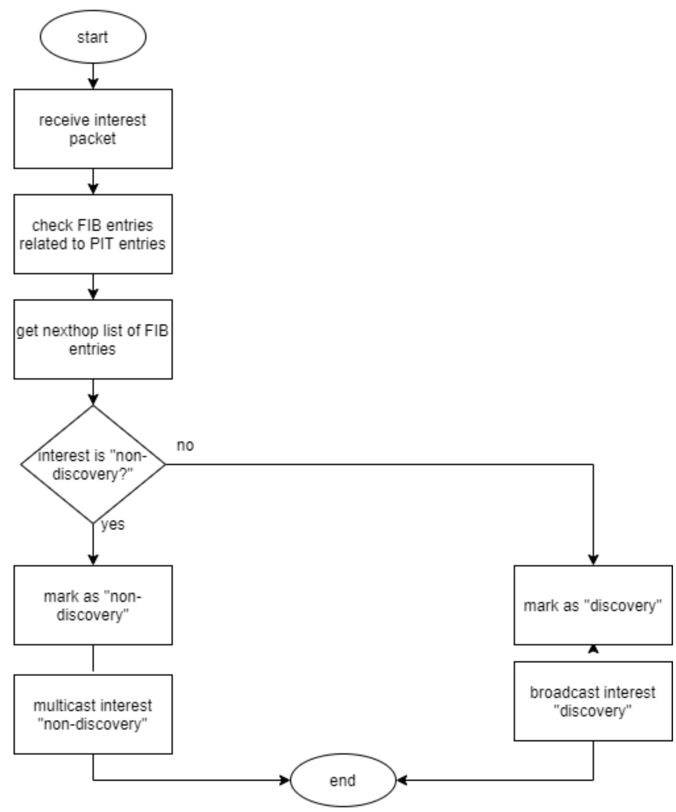

Fig. 1. Flowchart After Receive Interest on Broadcast Self-Learning Forwarding Modification

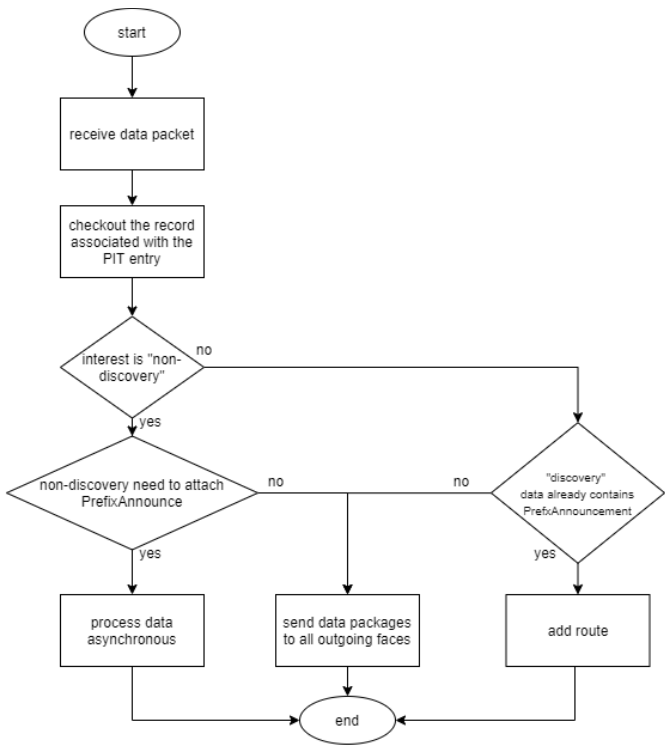

Fig. 2. Flowchart After Receive Data on Broadcast Self-Learning Forwarding Modification

The design introduced in citation [12] implements self-learning NDN in wired LAN environments utilizing Ethernet. Divided into two stages, namely 1). receive interest packets and 2). receive data packets. When a consumer requests content with a prefixed name, the consumer starts sending its first interest. After successfully sending interest, the interest packet is received by the router. The FIB is then updated by adding new entries. This entry provides information about how data packets associated with that prefix should be routed. The process of adding entries to the FIB can be done using the parameters specified by the pitEntry pointer. By adding this new entry to the FIB, NFD ensures that data packets arriving with this prefix are properly forwarded to the appropriate destination on the network. Next, store the destination list or next list received from the FIB entry in get nexthop. Next, assign a tag to the Interest to indicate that it is a “non-discovery” or “discovery” type. This tag is used to label the Interests. Interest “non-discovery” on matching FIB entry, the Interest will be multicast to the appropriate destination. Interest ”discovery”, will be broadcast to the entire network.

Then the next step can be seen in the Fig. 2 , 2). When the packet data has been received, then check the record related to the PIT entry to ensure that the incoming data matches the PIT entry. Next check the interest, when the interest is “non-discovered” and needs to attach a prefix announcement then process the data asynchronously with the asyncProcessData function. when the interest is “non-discovered” and does not need to attach a prefix announcement then send data to all outgoing faces. When the interest is “discover” and the data contains an announcement prefix then, the addition of a route is represented by the addRoute function. When there is no prefix announcement in the data, it is carried out to all outgoing faces.

In this research, when not receiving a packet, the next consumer can retransmit for the next process. This will reduce the complexity of the protocol and reduce the messages that must be processed. This reduces the delay associated with retransmitting unsuccessful packets.

D. Test Parameter

- Round-Trip Time (RTT), indicates the average time it takes to retrieve all data packets in a network through all nodes. As a key metric for network latency, changes in RTT values not only impact the user’s Quality of Experience, but can also indicate potential performance or security issues within the network [18]. Through the maintenance of the Pending Interest Table (PIT), the forwarding plane has the ability to measure RTT performance at each data capture step. When there are multiple next steps in the Forwarding Information Base (FIB) entry for an Interest, a forwarding strategy corresponding to the Interest name will determine the next step that will be used to forward the Interest based on route ranking, forwarding plane measurements, and local policy [21]. This allows a node to measure the time it takes to send an i-th request (Interest) and receive a reply or response relating to the i-th data requested, as follows [22] :

$$RTT_i = T(D_i) - T(R_i) \quad (1)$$

Description :
$T(R_i) = Interesttransmission(Request)$,
$T(D) = satisfaction(onData)$

- Throughput, recent research [23], highlights that reducing the frequency of sending ACK (Acknowledgment) signals or packets in the TCP/IP protocol can result in increased access to the communication channel for the transmission of Data packets thus potentially increasing the throughput or overall data transfer rate in applications that use the protocol. The same concept is applied in the NDN architecture (Named Data Network) [22], where the approach of taking multiple Data packets with a single Interest can also be used. The formula used to calculate the throughput value is as follows [24] :

$$Throughput = \frac{sizeofdatareceived}{datasendingtime} \quad (2)$$

IV. RESULT AND ANALYSIS

This system uses the NDN-based NDN4IVC Framework by combining the ndnSIM (NS-3-based NDN Simulator) and SUMO (Simulation of Urban Mobility) simulators for visualization of the vehicle node environment. it is assumed that each node represents each vehicle. Simulation environment design an urban environment is selected The so-called Urban area is characterized by a lot of hustle and bustle of complex activities, one of which is about transportation and high mobility of the city [25]. When configuring mobility, it is necessary to pay attention to the scenario of increasing the number of nodes because it affects the performance of the VANET network. Employing the 802.11p MAC protocol standard, a customized iteration of the IEEE 802.11 (Wi-Fi) standard, serves as the Medium Access Control (MAC) and Physical Layer (PHY) specification for Dedicated Short-Range Communication (DSRC) standards. This adaptation finds particular application in aerial networks, notably in vehicular ad hoc networks (VANETs) [26]. MAC 802.11p has a frequency range of 5.9 GHz (within the range of 5.85-5.925 GHz), this frequency range is set specifically for communication in vehicular environments and enables stable transmission isolated from interference with other Wi-Fi networks [27]. The scenario was conducted for 300 seconds in the ndn simulator using the NDN4IVC framework. Utilize a traffic management system (TMS) for exchanging data on traffic congestion via V2X (Vehicle-to-Everything) communication. The type of protocol used is UDP which supports situations where speed and efficiency are more important than reliability, and the movement of nodes is designed randomly which is assumed to be like a real vehicle. The CS size used is 1000. A factor that

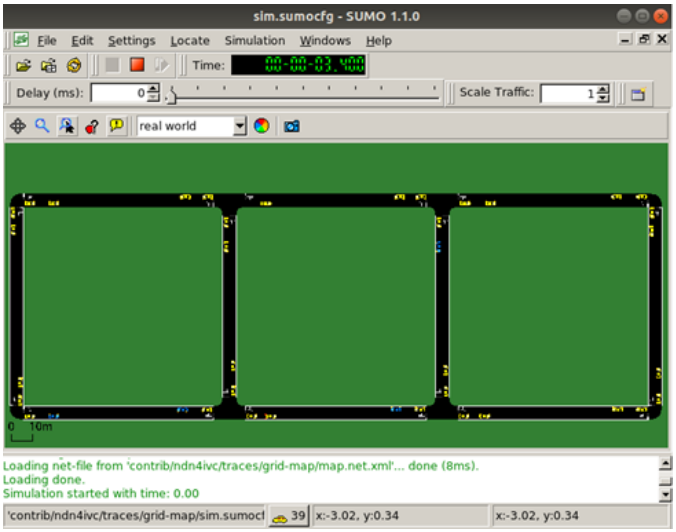

Fig. 3. Urban Simulation Area

TABLE I
SIMULATION PARAMETERS

Parameter	Value
Number of node	60; 70; 90; 100 unit vehicle
Number of RSU	2 unit
Strategy forwarding	Multicast, multicast-vanet, self-learning, self-learning modification
Interest interval	1000 ms
Simulation time	300 s
Simulation area	Urban
Mac Type	IEEE 802.11p
Node movement	Random Way Point
Traffic type	UDP/Traffic Management System (TMS)

can affect performance is the number of nodes. For the change scenario, the number of nodes tested were 60, 70, 90 and 100 nodes. Three extra forwarding strategies were employed for comparison purposes: (a) The default multicast scheme utilizes the NFD multicast forwarding strategy, which disseminates Interests to every neighboring node through flooding. (b) Multicast-vanet is implemented according to [14]. Disabling the NACK process to overcome the network interest flooding with NACK packets. (c) default self-learning, implementing the self-learning algorithm to NDN. (d) modified self-learning, adopting NACK removal in the self-learning algorithm to Vehicular Named Data Network. Next is to produce output in the form of the required parameters, namely Round Trip Time and throughput. Data obtained from the simulation results in the form of logs, then converted into .xlsx form and calculated based on the formula listed. The process of running the simulation is carried out three times to get an accurate value.

In VANET environments, where vehicles move dynamically and change paths frequently, low RTT is key to support responsive and reliable communication between vehicles and network infrastructure. The main benefit of simplifying the

TABLE II
ROUND TRIP TIME COMPARISON OF STRATEGY FORWARDING BASED ON NUMBER OF NODE

	Parameter (Round Trip Time (ms))			
Number of node	multicast	Multicast-vanet	Self-learning default	Self-learning modification
60	44.35	44.35	44.50	38.83
70	34.07	34.06	32.98	32.60
90	22.92	22.92	23.24	23.16
100	21.10	21.10	21.10	18.38

TABLE III
THROUGHPUT COMPARISON OF STRATEGY FORWARDING BASED ON NUMBER OF NODE

	Parameter (Round Trip Time (ms))			
Number of node	multicast	Multicast-vanet	Self-learning default	Self-learning modification
60	8.71	8.71	8.71	9.26
70	11.48	11.48	11.48	13.67
90	23.44	23.44	23.47	24.44
100	26.15	26.15	26.14	28.63

protocol communication complexity by reducing the negative acknowledgment protocol overhead is to reduce the round trip time in VANETs with dynamically moving vehicles.

In the Fig.4, it can be seen that the average RTT value at a node count of 60 self-learning modification is 12.7% lower than self-learning and 12.4% lower than multicast and multicast-vanet. Then at the number of nodes 70 self-learning modification 5.9% is lower than multicast and multicast-vanet and 1.95% lower than self-learning. Then at the number of nodes 90 self-learning modification 1% is higher than multicast and multicast-vanet and 0.34% lower than self-learning. At node count 100 self-learning modification is 12.89% lower than self-learning, multicast and multicast-vanet.

Based on this, an increase in the number of nodes indicates that more data packets are stored in the node cache. By reducing the negative acknowledgment protocol overhead, there can be an increase in efficiency as no time is spent on sending and processing NACK messages. This can help in reducing the overall RTT as the message does not have to wait for a negative response before attempting retransmission. When the increase in the number of nodes set is small (e.g. 10 nodes from the previous simulation), the switching that occurs in the experiment is not significant, so there is no performance change. A threshold of increasing the number of nodes set large (e.g. 20 nodes from the previous simulation) tends to yield greater benefits, as long as the threshold is set at the content store capacity of 1000.

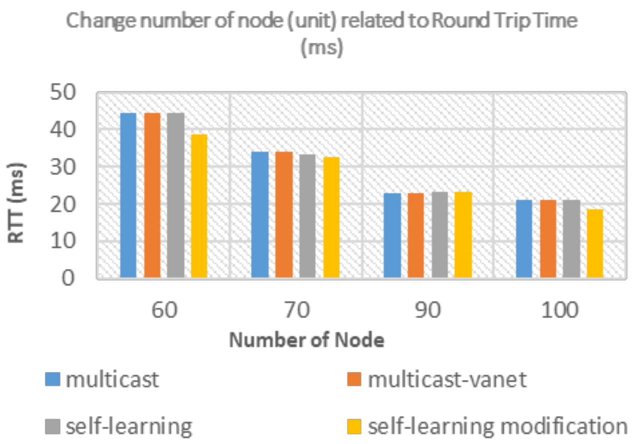

Fig. 4. Comparison graph of round trip time results

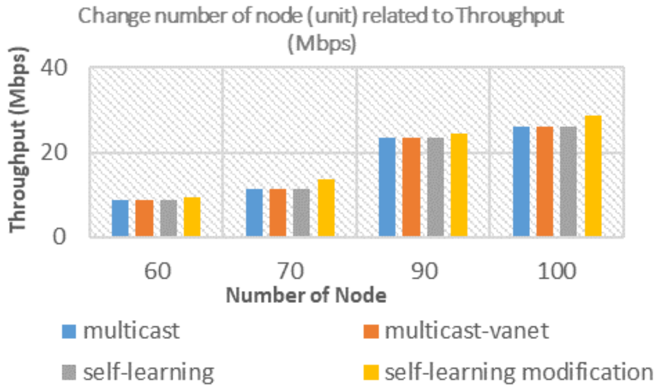

Fig. 5. Comparison graph of Throughput results

In the Fig.5, it can be seen that the average throughput value at the number of nodes 60 self-learning modification is 6.31% higher than self-learning, multicast and multicast-vanet. Then at the number of nodes 70 self-learning modification is 16.02% higher than self-learning, multicast and multicast-vanet. Then at a node count of 90 self-learning modification is 4.09% higher than multicast and multicast-vanet and 4.13% higher than self-learning. At a node count of 100 self-learning modification is 8.66% higher than multicast and multicast-vanet and 8.69% higher than self-learning.

Throughput can be said to be good, seen from the high performance of sending data from the source node to the destination node. Based on all the trials that have been carried out, the addition of the number causes the throughput value of the self-learning modification to increase. This is because the more nodes there are in the simulation environment, the denser the simulation environment will be so that the less likely the occurrence of link failure on the communication path and causes the durability of the link to be longer. The self-learning forwarding strategy without NACK is able to optimize data delivery automatically. By increasing the number of nodes, the system can have more data and experience that can be used to make better forwarding decisions, which in turn can increase throughput.

V. CONCLUSION

This paper discusses simplifying protocol complexity by reducing of negative acknowledgement protocols. The method used, namely modified self-learning, using the scenario of adding variations in the number of nodes in parameter testing. It also compares the proposed scheme with several other forwarding strategies namely multicast, multicast-vanet, default self-learning. Based on the test results, the self-learning forwaring strategy in the VANET-NDN network, it appears that in the scenario using urban vehicle route design, as well as the variation in the number of nodes shows that the more vehicle nodes, the smaller the round trip time. Reflecting that the proposed mechanism has a low round trip time (i.e. the number of nodes 60 self-learning modification is 12.7% lower than self-learning and 12.4% lower than multicast and multicast-vanet, the number of nodes 70 self-learning modification 5.9% lower value compared to multicast and multicast-vanet and 1.95% lower than self-learning, the number of nodes 90 self-learning modification 1% higher value compared to multicast and multicast- vanet and 0.34% lower than self-learning and the number of nodes is 100 self-learning modification is 12.89% lower than self-learning, multicast and multicast-vanet) and a high throughput value (namely the number of 60 self-learning modification nodes is 6.31% higher than self-learning, multicast and multicast-vanet, the number of nodes is 70 self-learning modification is worth 16.02% higher than self-learning, multicast and multicast-vanet, the number of nodes 90 self-learning modification is worth 4.09% higher than multicast and multicast-vanet and 4.13% higher than self-learning and the number of nodes 100 self-learning modification is 8.66% higher than multicast and multicast-vanet and 8.69% higher than self-learning), it is because the default self-learning forwarding strategy uses negative acknowledgment delivery so that more time is required to retrieve all successful packets. This protocol complexity simplification mechanism can be further evaluated in the future on a simulation basis in real scenarios.

REFERENCES

[1] D. Dhakal, A. Gautam, S. Dey, and K. Sharma, "A Review on Forwarding Strategies in NDN based Vehicular Networks," *EMITTER International Journal of Engineering Technology*, vol. 9, pp. 339–356, dec 2021.

[2] G. Xylomenos, C. N. Ververidis, V. A. Siris, N. Fotiou, C. Tsilopoulos, X. Vasilakos, K. V. Katsaros, and G. C. Polyzos, "A Survey of information-centric networking research," *IEEE Communications Surveys and Tutorials*, vol. 16, no. 2, pp. 1024–1049, 2014.

[3] C. A. Sarros, V. Demiroglou, and V. Tsaoussidis, "Intermittently-connected IoT devices: Experiments with an NDN-DTN architecture," *2021 IEEE 18th Annual Consumer Communications and Networking Conference, CCNC 2021*, 2021.

[4] Z. Ali, M. A. Shah, A. Almogren, I. Ud Din, C. Maple, and H. A. Khattak, "Named data networking for efficient IoT-based disaster management in a smart campus," *Sustainability (Switzerland)*, vol. 12, no. 8, pp. 1–21, 2020.

[5] H. Khelifi, S. Luo, B. Nour, H. Moungla, Y. Faheem, R. Hussain, and A. Ksentini, "Named Data Networking in Vehicular Ad Hoc Networks: State-of-the-Art and Challenges," *IEEE Communications Surveys and Tutorials*, vol. 22, no. 1, pp. 320–351, 2020.

[6] D. Hernandez, L. Gameiro, C. Senna, M. Luis, and S. Sargento, "Handling Producer and Consumer Mobility in IoT Publish-Subscribe Named Data Networks," *IEEE Internet of Things Journal*, vol. 9, no. 2, pp. 868–884, 2022.

[7] S. Fayyaz, M. A. U. Rehman, M. S. ud Din, M. I. Biswas, A. K. Bashir, and B. S. Kim, "Information-Centric Mobile Networks: A Survey, Discussion, and Future Research Directions," *IEEE Access*, vol. 11, no. April, pp. 40328–40372, 2023.

[8] K. Ahed, M. Benamar, A. A. Lahcen, and R. E. Ouazzani, "Forwarding strategies in vehicular named data networks: A survey," *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 5, pp. 1819–1835, 2022.

[9] J. Wan, X. Gu, J. Wang, and L. Chen, "A Trust Scheme Based on Vehicles Reports of Events in VANETs," *Wireless Personal Communications*, vol. 105, no. 1, pp. 121–143, 2019.

[10] Sandeep N. Kugali, "Vehicular ADHOC Network (VANET):-A Brief Knowledge," *International Journal of Engineering Research and*, vol. V9, no. 06, pp. 1026–1029, 2020.

[11] J. Shi, E. Newberry, and B. Zhang, "On broadcast-based self-learning in named data networking," *2017 IFIP Networking Conference, IFIP Networking 2017 and Workshops*, vol. 2018-Janua, pp. 1–9, 2017.

[12] T. Liang, J. Pan, M. A. Rahman, J. Shi, D. Pesavento, A. Afanasyev, and B. Zhang, "Enabling named data networking forwarder to work out-of-the-box at edge networks," *2020 IEEE International Conference on Communications Workshops, ICC Workshops 2020 - Proceedings*, 2020.

[13] M. A. Rahman and B. Zhang, "On Data-centric Forwarding in Mobile Ad-hoc Networks: Baseline Design and Simulation Analysis," *Proceedings - International Conference on Computer Communications and Networks, ICCCN*, vol. 2021-July, no. Icccn, 2021.

[14] G. Araujo, M. Peixoto, and L. Sampaio, "NDN4IVC: A Framework for Simulations of Realistic VANETs Applications through NDN," *ICN 2022 - Proceedings of the 2022 9th ACM Conference on Information-Centric Networking*, no. ii, pp. 162–164, 2022.

[15] F. Wu, W. Yang, Z. Fan, and K. Tian, "Multicast rate adaptation in WLAN via NDN," *Proceedings - International Conference on Computer Communications and Networks, ICCCN*, vol. 2018-July, 2018.

[16] M. Chowdhury, J. A. Khan, and L. Wang, "Smart forwarding in NDN VaNET," *ICN 2019 - Proceedings of the 2019 Conference on Information-Centric Networking*, pp. 153–154, 2019.

[17] A. Boudelaa, Z. Abdelhafidi, N. Lagraa, C. A. Kerrache, M. Bilal, D. Kwak, and M. B. Yagoubi, "SAFT-VNDN: A Socially-Aware Forwarding Technique in Vehicular Named Data Detworking," *Computers, Materials and Continua*, vol. 73, no. 2, pp. 2495–2512, 2022.

[18] M. Chowdhury, J. A. Khan, and L. Wang, "Leveraging Content Connectivity and Location Awareness for Adaptive Forwarding in NDN-based Mobile Ad Hoc Networks," *ICN 2020 - Proceedings of the 7th ACM Conference on Information-Centric Networking*, pp. 59–69, 2020.

[19] D. Saxena, V. Raychoudhury, N. Suri, C. Becker, and J. Cao, "Named Data Networking: A survey," *Computer Science Review*, vol. 19, pp. 15–55, 2016.

[20] E. T. da Silva, J. M. H. de Macedo, and A. L. D. Costa, "NDN Content Store and Caching Policies: Performance Evaluation," *Computers*, vol. 11, no. 3, pp. 1–17, 2022.

[21] L. Wang, V. Lehman, A. K. Mahmudul Hoque, B. Zhang, Y. Yu, and L. Zhang, "A Secure Link State Routing Protocol for NDN," *IEEE Access*, vol. 6, pp. 10470–10482, 2018.

[22] D. O. F. Philosophy, "DATA-CENTRIC WIRELESS NETWORKS," 2022.

[23] T. Li, K. Zheng, K. Xu, R. A. Jadhav, T. Xiong, K. Winstein, and K. Tan, "TACK: Improving Wireless Transport Performance by Taming Acknowledgments," *SIGCOMM 2020 - Proceedings of the 2020 Annual Conference of the ACM Special Interest Group on Data Communication on the Applications, Technologies, Architectures, and Protocols for Computer Communication*, pp. 15–30, 2020.

[24] M. N. D. Satria, F. H. Ilma, and N. R. Syambas, "Performance comparison of named data networking and IP-based networking in palapa ring network," *Proceedings - ICWT 2017: 3rd International Conference on Wireless and Telematics 2017*, vol. 2017-July, pp. 43–48, 2017.

[25] E. Suryani, R. A. Hendrawan, P. F. Adipraja, and R. Indraswari, "System dynamics simulation model for urban transportation planning: A case study," *International Journal of Simulation Modelling*, vol. 19, no. 1, pp. 5–16, 2020.

[26] F. Arena, G. Pau, and A. Severino, "A review on IEEE 802.11p for intelligent transportation systems," *Journal of Sensor and Actuator Networks*, vol. 9, no. 2, pp. 1–11, 2020.

[27] D. Korzun, A. Vdovenko, and O. Bogoiavlenskaia, *Distributed Computer and Communication Networks: Control, Computation, Communications*, vol. 1337. 2016.