electronics

MDPI

Article

QWLCPM: A Method for QoS-Aware Forwarding and Caching Using Simple Weighted Linear Combination and Proximity for Named Data Vehicular Sensor Network

Dependra Dhakal * and **Kalpana Sharma**

Sikkim Manipal Institute of Technology, Sikkim Manipal University, Majhitar, Rangpo 737136, India; kalpana.s@smit.smu.edu.in
* Correspondence: dependra.d@smit.smu.edu.in

Abstract: The named data vehicular sensor network (NDVSN) has become an increasingly important area of research because of the increasing demand for data transmission in vehicular networks. In such networks, ensuring the quality of service (QoS) of data transmission is essential. The NDVSN is a mobile ad hoc network that uses vehicles equipped with sensors to collect and disseminate data. QoS is critical in vehicular networks, as the data transmission must be reliable, efficient, and timely to support various applications. This paper proposes a QoS-aware forwarding and caching algorithm for NDVSNs, called QWLCPM (QoS-aware Forwarding and Caching using Weighted Linear Combination and Proximity Method). QWLCPM utilizes a weighted linear combination and proximity method to determine stable nodes and the best next-hop forwarding path based on various metrics, including priority, signal strength, vehicle speed, global positioning system data, and vehicle ID. Additionally, it incorporates a weighted linear combination method for the caching mechanism to store frequently accessed data based on zone ID, stability, and priority. The performance of QWLCPM is evaluated through simulations and compared with other forwarding and caching algorithms. QWLCPM's efficacy stems from its holistic decision-making process, incorporating spatial and temporal elements for efficient cache management. Zone-based caching, showcased in different scenarios, enhances content delivery by utilizing stable nodes. QWLCPM's proximity considerations significantly improve cache hits, reduce delay, and optimize hop count, especially in scenarios with sparse traffic. Additionally, its priority-based caching mechanism enhances hit ratios and content diversity, emphasizing QWLCPM's substantial network-improvement potential in vehicular environments. These findings suggest that QWLCPM has the potential to greatly enhance QoS in NDVSNs and serve as a promising solution for future vehicular sensor networks. Future research could focus on refining the details of its implementation, scalability in larger networks, and conducting real-world trials to validate its performance under dynamic conditions.

Keywords: named data networking; VSN; forwarding; caching; weighted; proximity

Citation: Dhakal, D.; Sharma, K. QWLCPM: A Method for QoS-Aware Forwarding and Caching Using Simple Weighted Linear Combination and Proximity for Named Data Vehicular Sensor Network. *Electronics* **2024**, *13*, 1183. https://doi.org/10.3390/electronics13071183

Academic Editors: Dongkyun Kim, Apostolis Xenakis, Costas Chaikalis, Ilias K. Savvas and Kosmanos Dimitrios

Received: 18 January 2024
Revised: 9 March 2024
Accepted: 21 March 2024
Published: 23 March 2024

1. Introduction

Named data networking (NDN) is an emerging network architecture that has attracted significant attention from researchers and practitioners because of its ability to address some of the challenges faced by traditional Internet Protocol (IP) networking. One key advantage of NDN is its inherent support for content-oriented communication [1], making it well-suited for use in vehicular sensor networks (VSNs), focusing on accessing and sharing sensor data. A significant surge of interest in VSNs has occurred from both industry and academia. As the automotive industry moves rapidly toward autonomous vehicles, we see a critical need for technologies that can provide reliable communication and quality of service (QoS) in VSNs [2,3].

In NDN, routers use data names to route interest packets toward potential data sources. When the content label of an interest packet matches the content name of a data packet, the data packet is considered to fulfill the interest. NDN names are organized hierarchically to represent individual bits of information carried by a packet. Key data structures in NDN include the content store (CS), pending interest table (PIT), and forwarding information base (FIB). The FIB contains data identifier prefixes that guide interest packets to appropriate data sources, enabling faster response to future interests by temporarily storing data packets in the CS or cache at each node. The PIT tracks pending interests, which are requests waiting for data packets to be delivered, ensuring the efficient delivery of data to requesters [4]. NDN employs two planes [5] for packet transmission: the routing plane and the forwarding plane. The forwarding plane, also known as the control plane, provides stateful and intelligent control. It uses a forwarding strategy module to make forwarding decisions, relieving the router from the responsibility of making all decisions. In addition, the forwarding plane can detect and recover from network faults without assistance from the routing plane. The routing plane assists the forwarding plane by processing the FIB. Unlike IP networks—which transmit packets through a single interface at a time—NDN supports multi-path routing, such that packets can be routed across multiple interfaces for the same purpose [6].

Named data vehicular sensor networks (NDVSNs) support both safety and non-safety applications in vehicular networks [7]. Safety applications aim to enhance road safety and reduce accidents by enabling vehicles to communicate and share information, including collision avoidance, intersection safety, and emergency vehicle notification systems. Non-safety applications focus on improving the driving experience, providing entertainment, and optimizing transportation systems, such as traffic management, infotainment, and location-based services. Safety applications take precedence in NDVSNs, as they are critical for saving lives and preventing accidents, and non-safety applications should not interfere with the functioning of safety applications. Therefore, designing NDVSNs with prioritization of safety over non-safety applications in forwarding and caching is crucial [8].

To achieve these objectives, NDVSNs require a well-defined QoS mechanism. Researchers have addressed QoS requirements in NDVSNs, particularly focusing on forwarding, caching, and routing [9–13]; however, existing forwarding strategies often fail to consider QoS factors, such as priority and stability, required for high-speed vehicles where frequent path breaks occur. Additionally, the default caching strategy in NDN architectures stores all received packets, regardless of cache size, prioritization, or stable nodes, leading to excessive overhead and delays. To address these challenges, the proposed approach, called QWLCPM (QoS-aware Forwarding and Caching using Weighted Linear Combination and Proximity Method), introduces a weighted linear combination (WLC) method for forwarding and caching decisions in NDVSNs. This method utilizes a set of weights to determine the importance of each metric used for forwarding and caching. Stable nodes are selected based on the WLC method as intermediaries for data forwarding. The caching probability is determined to identify data items that need to be replaced from the cache. QWLCPM offers a simple yet effective solution for managing data in NDVSNs, aiming to improve the network's QoS. The WLC method provides a promising approach to address challenges and optimize the caching probability and stable node determination in VSNs. The QWLCPM method optimizes vehicular networks through advanced forwarding and caching, considering node stability, data priority, and node proximity, alongside a spatial-temporal strategy for timely data eviction. By employing a WLC and proximity method (PM), it identifies forwarding nodes and calculates caching probabilities to ensure optimal QoS. Stable nodes, chosen for their strategic locations, act as efficient data carriers, reducing transmission distances and network load. The network is segmented into zones defined by RSU transmission ranges to streamline communication and prioritize data delivery effectively. This zonal approach enhances network resource utilization, minimizes congestion, and ensures priority data reach their intended recipients swiftly and reliably, significantly improving network performance and service quality.

QWLCPM stands out for its holistic approach towards enhancing network performance in vehicular environments through a multi-metric evaluation, which scrutinizes node stability, signal strength, vehicle speed, and proximity to pinpoint the most stable nodes for data forwarding, ensuring consistent communication in dynamic conditions. The application of a weighted linear combination method allows for the prioritization of metrics tailored to current network conditions and application demands, enabling the algorithm to flexibly modify its decision-making criteria to optimize performance across different scenarios, such as urban or highway settings. Furthermore, its dynamic caching strategy, which aligns with the quality of service requirements and the network's present state, prioritizes caching and forwarding decisions based on node stability and QoS needs, ensuring high-priority data are more likely to be managed and propagated by reliable nodes. The algorithm's capacity to adapt to network changes, including variations in vehicle density, speed, and direction, guarantees its effectiveness, even with swiftly shifting network conditions. By intelligently deciding on the nodes for data forwarding and caching locations, QWLCPM enhances the efficient utilization of network resources, reducing redundant data transmissions and boosting network efficiency, a critical aspect in bandwidth- and resource-constrained environments. Ultimately, QWLCPM's goal is to elevate QoS in vehicular networks, aiming for reduced latency, enhanced data delivery rates, and sustained communication reliability and efficiency amidst high mobility and evolving network topologies.

The remainder of this paper is organized as follows: Section 2 provides an overview of existing literature and prior research relevant to the field. The core of the proposed strategy, described in Section 3, intricately details the comprehensive approach, explaining how it factors in node stability, data priority, and spatial–temporal considerations for cache management and forwarding decisions. Section 4 includes the forwarding metrics—such as zone separation, geolocation (GPS), vehicle speed, transmission signal strength, and vehicle ID assignment—to optimize QoS. Section 5 delves into the intricacies of the forwarding process, elucidating Hello Packet exchange, communication disruption, network partitions, beacon nodes, priority queue processing, and the packet type and naming scheme employed in the NDVSN architecture. Section 6 outlines the caching methodology utilized while also discussing content diversity and hit ratio as performance indicators. To facilitate a better understanding of the experimental setup, Section 7 elaborates on the tools and methods used for performance evaluation. Lastly, Section 8 summarizes the research findings, emphasizing the efficacy of the proposed QWLCMP approach in urban vehicular environments and outlining future research directions aimed at enhancing its performance in highway scenarios.

2. Related Work

Numerous caching and forwarding methods have been proposed to enhance QoS and overall performance in vehicular networks. These methods aim to optimize the storage and delivery of data packets, considering factors such as traffic classification, content popularity, mobility patterns, and network conditions. This section provides an overview of key caching and forwarding strategies proposed in the literature.

In one approach, the authors of [6] introduced a QoS-linked privileged content caching (QLPCC) system to manage the PIT and content store in a QoS traffic scenario. It utilizes a flow table to classify traffic and assigns eviction scores to content based on flow IDs. The system records content expiry times, flow IDs, and paths to determine whether to cache or forward content. An eviction algorithm was proposed to prioritize content based on its relevance and reliability, effectively freeing up memory resources. In another paper, to address the diverse needs of different applications, the authors' pNDN scheme [7] utilized name prefixes to indicate globally recognized priorities for vehicular data traffic. This approach was built upon standard NDN forwarding algorithms and allowed for the differentiated treatment of high- and low-priority data, reducing latency. Similarly, the liteNDN [14] strategy optimized packet-forwarding decisions and data retrieval by sharing knowledge among routers. It considers routing costs and the significance of data to make informed decisions about caching packets, resulting in reduced latency and unnecessary traffic and improving the caching efficiency when compared to conventional NDN strategies.

In terms of forwarding algorithms, researchers in [15] presented a priority-based approach that assigns data packets to high-, medium-, and low-priority levels. The algorithm utilizes push- or pull-based forwarding methods based on the packet's priority level, aiming to improve the hit ratio and reduce network overhead. For cooperative caching, a cluster-based scheme [16] was proposed that takes vehicle mobility patterns into account. The network is divided into clusters, and data are cooperatively cached among vehicles within the same cluster. A weight is assigned to each vehicle to determine the optimal cluster head, considering its suitability, which is refreshed at regular intervals. However, cluster head selection in high-speed vehicular networks may introduce overhead.

Additionally, a dynamic cooperative cache management scheme [17] was suggested, relying on popular and social data and involving a master node that operates hierarchically with nearby nodes to retain frequently accessed contents. However, in this scheme, the master node may experience a bottleneck during high network activity. In another approach, distributed probabilistic caching [18] allows each node to make caching decisions independently based on user demands, vehicle importance, and relative motion. A clustering algorithm is utilized to extract information on user demands and preferences, but this may introduce additional overhead to the network.

To address connectivity disruptions and topology changes, Navigo [19] fetches data chunks from multiple potential providers instead of relying on a single producer. It has a self-learning feature to identify the locations of content providers, facilitating faster data retrieval. Futhermore, hybrid protocols such as HVNDN [20] integrate interest and data packets with acknowledgment and re-transmission mechanisms. They employ probabilistic and opportunistic forwarding strategies for both location-independent and -dependent content.

Different forwarding strategies tackle specific challenges; for example, LISIC [21] prioritizes interest packet transmission based on link stability to avoid broadcasting storms, while RUFS [22] reduces the broadcast storm problem by selecting only one neighbor node to forward interest packets based on satisfaction rates. The LDE [23] forwarding strategy combines local decision-making and global coordination to minimize overhead and increase the packet-delivery ratio. It considers distance-based decision-making and employs a feedback mechanism to detect bottlenecks and congestion points. The mobility-prediction-based forwarding strategy [24] reduces interest broadcasting storms by validating information in the neighbor table (NBT) and selecting the next-hop forwarder based on distance along the road, stable link, and link expiry time. The authors of [9] proposed a cache replacement policy, dividing the cache store into sub-cache stores based on traffic classification and content size. Each content item is assigned a popularity-density value, and the highest-value content is cached while the lowest is removed. Simulation experiments confirmed its effectiveness. The authors also introduced a content popularity-diversity replacement policy based on the knapsack problem, which proved to be effective and scalable. Finally, a geographic opportunistic forwarding strategy [25] was devised, combining Delay-Tolerant Network (DTN) and NDN models to enhance data-delivery quality using multi-source routing and router caching and selecting reliable relay nodes based on geographical information.

Most existing strategies, such as QLPCC, pNDN, liteNDN, and others, focus on either improving caching efficiency or refining forwarding algorithms but often lack a holistic approach that considers the rapid changes in network topology, the high mobility of vehicles, and varying data priorities in real time.

Furthermore, while some methods, like the dynamic cooperative cache management scheme and distributed probabilistic caching, offer innovative solutions to data storage and dissemination, they may suffer from scalability issues or introduce additional overhead due to the reliance on master nodes or complex clustering algorithms. This indicates a need for more scalable, decentralized solutions that can adapt without significant overhead or bottlenecks, especially in high-density vehicular environments.

Another gap exists in addressing the seamless integration of QoS parameters such as content popularity, user demand, and vehicle mobility patterns with network conditions to ensure reliable and efficient data forwarding and caching. Strategies like Navigo, HVNDN, and geographic opportunistic forwarding explore parts of this challenge but do not fully exploit the potential of predictive analytics and real-time data to enhance network performance comprehensively. Additionally, while efforts like LISIC and RUFS aim to minimize broadcasting storms and improve packet delivery ratios through localized decision-making, there remains a significant opportunity to develop a more cohesive strategy that leverages global network insights alongside local data to optimize decision-making processes. This would involve integrating feedback mechanisms that can dynamically adjust caching and forwarding criteria based on current network conditions, content requirements, and vehicle dynamics.

In conclusion, the existing research literature has emphasized the need for sophisticated caching and forwarding strategies in vehicular networks. These strategies consider factors such as traffic classification, content popularity, mobility patterns, network conditions, and user demands to optimize data storage, retrieval, and dissemination. By effectively integrating caching and forwarding mechanisms, we can improve the overall performance and QoS in vehicular NDN architectures.

3. QWLCPM Strategy

QWLCPM is a comprehensive approach for forwarding and cache management in vehicular networks, considering the stability of nodes, the priority of data, and the proximity of nodes to their request, as well as incorporating temporal and spatial components such that data are evicted on time, as detailed in Table 1.

Table 1. Application types and validity parameters. Adapted with permission from [26]. Copyright 2020 IEEE.

Application Type	Service Type	Space Validity	Time Validity
Safety applications (high priority)	Work zone warning	500 m to 1 km	Until Work Finished
	Vehicle accident warning	500 m to 1 km	30 s
	Dangerous road warning	500 m to 1 km	20 s
Traffic applications (medium priority)	Road congestion	5 km	20 min
	Traffic map	10 km	20 min
Comfort applications (low priority)	Multimedia file sharing	1 km	15 min
	Commercial advertisement	1–5 km	1–5 days

This technique uses a WLC and a proximity-determination method to find the forwarding nodes and caching probability and determine which data should be cached to achieve the desired QoS. The WLC approach uses a set of weights to assess the importance of each data item in the cache and calculates the caching probability to define the likelihood of each data item being replaced. The stable nodes are selected based on proximity determination using the weighted linear approach, and they play an important role in forwarding the data from the source to the destination. These nodes act as intermediaries, reducing the need for data to be transmitted over longer distances and reducing the overall network load. The combination of the WLC and proximity-determination approaches provides a promising solution for improved QoS in VSNs. The WLC approach achieves efficient cache management, while the proximity determination optimizes the performance of the network by allowing nodes to communicate with other nodes in the network. QWLCPM operates as a hybrid model, leveraging both distributed decision-making based on local node metrics and a form of centralized organization provided by RSUs.

4. Forwarding Metrics

The QWLCPM method, which is discussed in the following subsection, utilizes various metrics to identify a stable node within one-hop neighbors and forwards the packet all the way to the intended producer node.

4.1. Zone Separation

The vehicular network is divided into zones based on the transmission range of roadside units (RSUs), as shown in Figure 1. RSUs can only communicate with vehicles and other RSUs within a certain distance. This division of the road network into zones allows for the more effective management of communication within each zone, optimizing network resources and improving overall efficiency.

The deployment of RSUs is becoming increasingly common, with transportation agencies and municipalities recognizing the value of RSUs in managing traffic, enhancing safety, and supporting intelligent transportation systems. With the advent of 5G and beyond, the integration of RSUs in vehicular networks is expected to accelerate, offering high-speed, low-latency communication capabilities essential for the QWLCPM scheme. RSUs play a pivotal role in the QWLCPM scheme by providing stable, high-capacity nodes that can manage data forwarding and caching with greater reliability than moving vehicles. This stability is crucial for implementing QWLCPM advanced QoS-prioritization mechanisms, ensuring that critical data are efficiently distributed across the network. The QWLCPM scheme is positioned to take full advantage of emerging infrastructure enhancements, ensuring that it remains relevant and effective as vehicular networks evolve. This forward-looking approach ensures the scalability and adaptability of the scheme to future technological advancements [27].

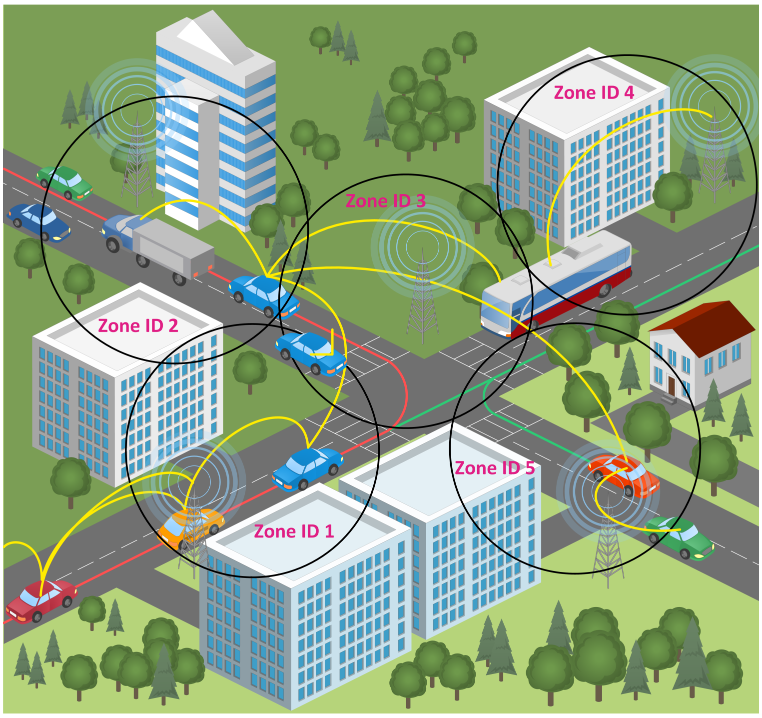


Figure 1. Zone separation.

A zone-based division facilitates the efficient and effective delivery of priority-based data. RSUs in high-priority zones prioritize the delivery of critical information, ensuring quick and reliable transmission of emergency notifications and other important data. This approach provides QoS and reduces network congestion. Priority data are transmitted within the same zone, making better use of network resources and minimizing congestion risks.

When a vehicle enters the range of an RSU or its neighboring RSU, it is assigned a zone. In the case of overlapping zones, the RSU with the larger transmission range is assigned the zone. If a node moves out of its designated zone, it transfers high-priority packets to the RSUs to increase the cache hit ratio. Meanwhile, vehicles continue sending measurement data, including their transmission range, to the RSU. When the transmission range falls below a threshold, vehicles offload their high-priority tasks to the RSU before moving to the next RSU range or zone, as shown in Algorithm 1.

Algorithm 1: Zone Assignment and Priority Task Offloading

```
Data: Set of all vehicles V and all RSUs R in the network
Result: Zones assigned to each vehicle and RSU
for each RSU i ∈ R do
    Set Z_i = {v ∈ V | d(v, i) ≤ Tx(i)};
    Assign P(v) = Z_i for all v ∈ Z_i;
end
for each vehicle v ∈ V do
    Let N_i be the set of all one-hop neighbors of v within Tx(v);
    If there exists an RSU i such that N_i is a subset of Z_j for some Z_j in R:
        Assign P(v) = Z_j
    Otherwise, assign P(v) = null. Let U be the set of indices of overlapping zones
    for v;
    if U is not empty then
        Let m = arg max_{n∈U} (Tx(n))
            Assign P(v) = Z_m
    end
    if v moves out of its designated zone P(v) then
            Offload high-priority packets to the RSU in P(v)
    end
    Send measurement data to the RSU in P(v), including the transmission range
    of v if the transmission range of v falls below a threshold then
            Offload high-priority tasks to the RSU in P(v) before moving on to the
            next RSU range or zone
    end
end
```

In the process of zone assignment and priority task offloading, the algorithm undertakes three main steps. Firstly, it identifies all the overlapping zones that the vehicle is in, forming the set U containing the indices of overlapping zones. Secondly, it iterates over the indices in U to find the RSU with the largest transmission range among the overlapping zones, computed as m, where $Tx(n)$ represents the transmission range of RSU n. Once the RSU with the largest transmission range m is determined, the algorithm assigns the corresponding zone Z_m to the vehicle.

4.2. Geolocation

In an NDVSN, vehicles can act as both consumers and producers of data, requesting and disseminating information based on user interests or vehicular conditions. Geolocation information can be used to determine the location of vehicles in the network and their proximity to each other. The QWLCPM assigns a geolocation, or global position, to every vehicle or node in the network, which can then be utilized to identify stable nodes in close proximity for content distribution. Let V be the set of all nodes or vehicles in the network.

For each $n\ V$, let GPS(n) be the geolocation value of node n. The proximity of two nodes n and m can be determined by the distance between their GPS values:

$$\text{distance}(n, m) = \sqrt{(\text{GPS}(n) - \text{GPS}(m))^2}. \tag{1}$$

4.3. Vehicle Speed

Using the average speeds (or velocities) of vehicles to determine stable nodes within a certain proximity in an NDVN can improve the efficiency of content dissemination. By finding the speed of one-hop neighbor vehicles, we can identify vehicles that are more likely to remain in the same location for a longer time, increasing their stability as nodes for content dissemination. For example, vehicles stopped at a traffic signal or parked in a parking lot are more stable nodes for content dissemination than those moving at high speeds on a highway. Similarly, we can identify vehicles that are more likely to be in proximity to each other. Vehicles moving at similar speeds in the same direction on a road are more likely to be near each other than vehicles that are moving at different speeds or in opposite directions. Thus, one critical factor used in QWLCPM to determine stable nodes is vehicle speed, which is shared with their one-hop neighbors.

4.4. Transmission Signal Strength

Transmission signal strength (Tss) refers to the intensity of the signal transmitted from one vehicle to another. By measuring the Tss between vehicles, vehicles in close proximity to each other can be identified and used to select stable nodes for content dissemination. For example, vehicles close to each other with a strong Tss between them are more likely to be stable nodes for content dissemination than those further apart or with weak Tss. The closer the vehicles and the stronger the Tss, the more likely it is that the vehicles can be stable nodes for content dissemination.

4.5. Vehicle ID Assignment

Using the same issued vehicle ID is helpful in determining stable nodes. By using the same vehicle ID in consecutive communication, we can identify vehicles that are more likely to remain in the same location for a longer period, increasing their stability as nodes for content dissemination. Multiple vehicles communicating using the same vehicle ID indicates that these vehicles are in proximity and can serve as potential nodes for content dissemination. This approach can be particularly useful in scenarios where a group of vehicles is traveling together in a convoy or a platoon or in a city scenario where the speed of the vehicles is similar because of traffic congestion.

5. Forwarding

Forwarding plays a vital role in facilitating the name-based data-retrieval process. When a node receives a data request, it checks its local cache to determine whether it has a copy of the requested data. If so, it sends the data back to the requester. If not, it forwards the request to its neighboring nodes in the hopes that one of them has the desired data. This section covers the different processes involved in packet forwarding by identifying a stable node as the next-hop forwarder.

5.1. Hello Packet

Hello packets are used to create a one-hop stable node neighbor list. They contain the following fields: vehicle ID, geoLocations, speed, and zone ID. When a node receives an interest or data packet that piggybacks neighbor information, it can check if it already has the latest neighbor information. If so, it can defer forwarding of the next scheduled hello packet for a short period. During this period, if the node receives an interest or data packet that contains updated neighbor information, it can update its neighbor information and cancel the deferred hello packet. If the node does not receive updated neighbor information during this period, it can forward the deferred hello packet to its neighbors. Deferring hello packets when neighbor information is already available in piggybacked interest or data packets can help reduce network traffic and improve network efficiency, as nodes do not need to periodically send hello packets if they already have the latest neighbor information. The algorithm for hello packet generation is given as Algorithm 2.

Algorithm 2: Hello Packet

Input : Node's neighbor list, Hello packet interval T_{hello}, Defer interval T_{defer}
Output: Updated neighbor list after each hello packet interval

```
Set the node's neighbor list to an empty set;
Start a timer for the hello packet interval, T_hello;
while true do
    Wait for the next hello packet interval;
    if node has received any interest or data packet with piggybacked neighbor information
      in the last T_hello interval then
        Update node's neighbor list with piggybacked neighbor information
          (vehicle ID, geoLocations, speed, and zone ID);
    else
        Defer the hello packet transmission for a short period, T_defer;
        while waiting for T_defer do
            if new interest or data packet arrives with piggybacked neighbor information
              then
                Update node's neighbor list with piggybacked neighbor information
                  (vehicle ID, geoLocations, speed, and zone ID);
                Cancel the deferred hello packet transmission;
                Break from the loop;
            end
        end
        if no new neighbor information arrived then
            Send the deferred hello packet to all neighbors;
        end
    end
    Reset the hello packet interval timer;
end
```

5.2. Communication Disruption

If a path break occurs during data forwarding, then the data are fetched from the intermediate nodes that had cached data earlier or from the node where the link break occurred. The disruption may occur in the communication link between neighboring vehicles forwarding data messages to the content requester because of vehicle speed and distance, transmission range, or an unreliable communication medium. In such cases, alternate save paths and routes are used to deliver the data. Alternately, RSUs also play an important role in delivering these messages, as vehicles leaving the zone or high-priority data are uploaded into RSUs. Vehicles unload packets in order of priority, such that high-priority data are unloaded into RSUs as soon as they move out of range or zone.

5.3. Network Partitions

In crowded urban settings, acknowledgment is unnecessary as vehicles are typically very close together. A sender's interest need not be acknowledged, as the next hop can hear the transfer in the shared channel. Data packets can be re-sent from the intermediary nodes where a disconnect occurs if a consumer has not received them after a certain period.

5.4. Beacon Nodes

Fast-moving nodes with a speed greater than the average speed are referred to as "beacon nodes", which are responsible for broadcasting high-priority and other information to RSUs and nodes in the range as they move forward. As these nodes, i.e., vehicles, cross other vehicles in the path, they act as beacon nodes, or carriers and distributors of information in the network, passing several zones in between. Beacon nodes are more useful in highway networks with sparse and fast vehicles. Algorithm 3 demonstrates how beacon nodes broadcast priority information within their range.

Algorithm 3: Broadcasting Information Using Beacon Nodes in a Vehicular Network

Input : N, V, S, A, H
Output: Broadcasting of high-priority information to nodes in the range of beacon nodes

B ← v ∈ V | speed(v) > A; Assign high priority to all nodes in B; **foreach** b ∈ B **do**
    Broadcast all information in H to nodes within range of b; Move b forward in the network;
if *new information becomes available* **then**
    Update H; Update the information being broadcast by all nodes in B;

Nodes with speeds exceeding the threshold (A) are selected as beacon nodes (B). These nodes are assigned a high priority and are responsible for disseminating information as they move through the network. Beacon nodes in subset B broadcast high-priority information (H) to other nodes within their range as they progress forward in the network. This enables the distribution of information to neighboring nodes and RSUs within the range of these beacon nodes. If new information becomes available, the algorithm aims to update the stored information (H) and broadcast the updated information by all beacon nodes in subset B. This step allows for the dissemination of the latest information across the network. S (the subset of beacon nodes) is not directly used in the algorithm provided, but it is conceptually relevant as it can represent the beacon nodes formed by selecting nodes meeting the speed criteria.

5.5. Priority Queue Processing

A packet at a router is classified and assigned to a priority queue. The router processes packets based on priority, giving precedence to higher-priority packets. Each node has a primary queue that distributes packets in three levels: high, medium, and low. High-priority tasks move to medium priority after all high-priority packets are processed, and the same occurs for medium to low priority. As high-priority tasks may starve medium- and low-priority queues, a multi-level feedback queue is used. When traffic increases, the average packet count in each queue is calculated, adjusting the feedback queue to prevent starvation. Figure 2 shows the flow chart for packet processing prioritization at the router.

Let N be the total number of packets in the system, and P_h, P_m, and P_l represent the numbers of packets in the high-priority queue, medium-priority queue, and low-priority queue, respectively. The process's main queue function categorizes a packet based on its priority and adds it to the corresponding priority queue. This function is called when a packet is added to the main queue. The process's high queue function handles the highest-priority packets in the high-priority queue. If the high queue is empty, it moves to the medium queue. A packet that is present in the high queue is processed, and the average packet processing time is updated accordingly. The process medium queue function handles packets in the medium queue. If the medium queue is empty, it moves to the low queue. If the average processing time multiplied by the length of the medium queue exceeds the length of the high queue, a packet is moved from the medium queue to the high queue. The process's low queue function processes the lowest-priority packets in the low queue. If both the high queue and the medium queue are empty, packets are processed from the low queue. If the average processing time multiplied by the length of the medium queue exceeds the length of the low queue, a packet is moved from the low queue to the medium queue.

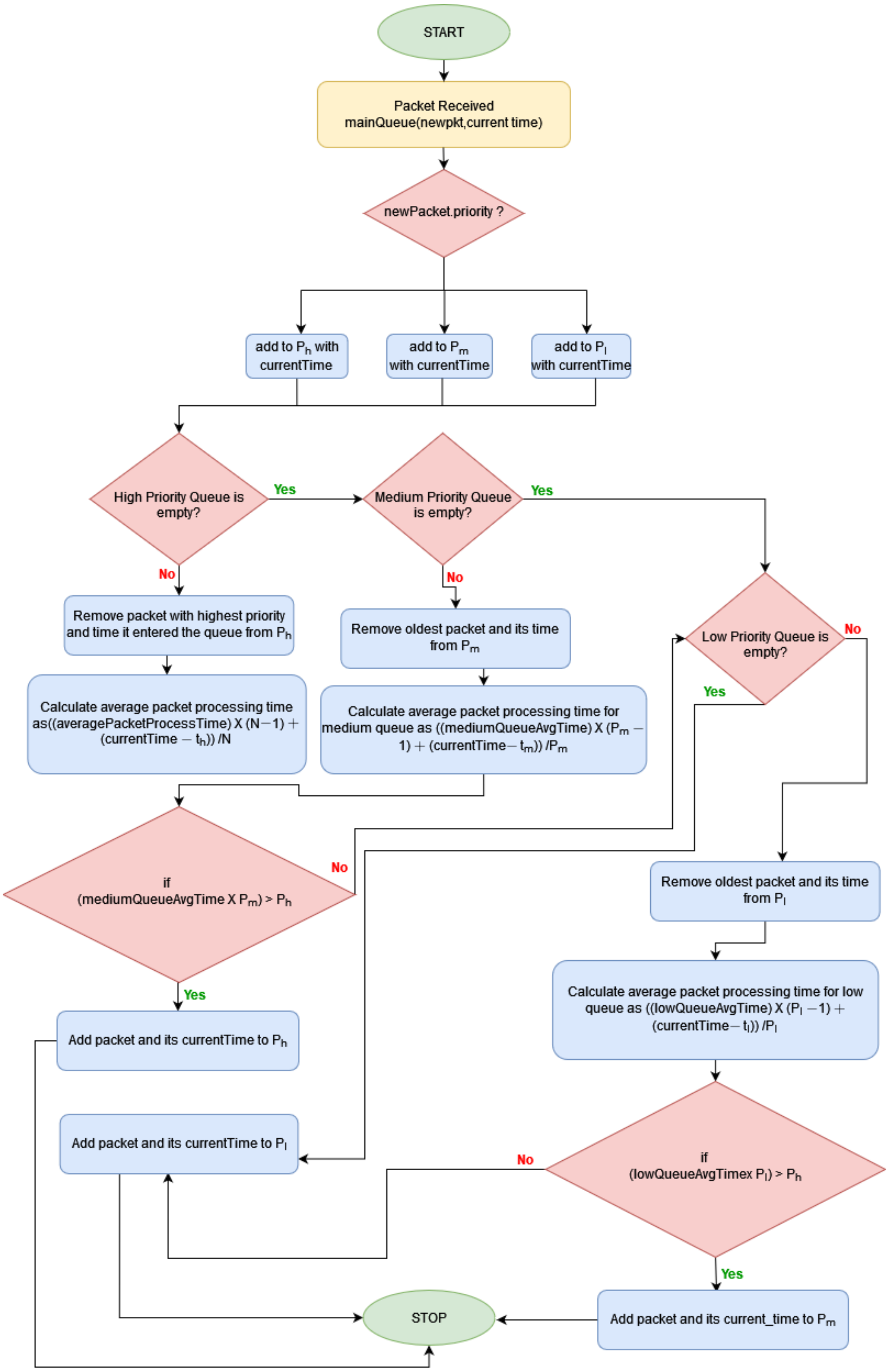


Figure 2. Queue processing of priority packets.

5.6. Packet Type and Naming Scheme

The three types of packets are interest, data, and hello packets. Interest and data packets are modified to piggyback the zone ID, geolocations, vehicle speed, and vehicle ID during the movement of packets between nodes. A hierarchical namespace with a universally recognized prioritization set of values is assumed to be agreed upon by vehicular devices and other entities exchanging data with them (e.g., RSUs and nodes). For simplification, vehicular applications are divided into three main priority categories—high, medium, and low—based on their latency requirements and assigned temporal and spatial validity using the techniques developed in our previous paper [28]. However, this approach can be expanded to include additional service classes. Similarly to the method used in [7], three prefixes (high, medium, and low) are used to indicate the class membership of a given packet. The mapping of various types of vehicular application content to the three priority classes is depicted in our paper. Road traffic information is critical for vehicles in congested areas and must be constrained by strict temporal and spatial requirements. Hence, accident data with time and spatial validity can be identified globally by utilizing the prefixes accident-data, high, crash-type, 3 min, and 1 km, where the last two parameters specify the time and spatial validity, respectively.

5.7. Finding Stable Nodes

Using a combination of zone ID, Tss, speed, geolocation, and vehicle ID can significantly improve the determination of stable nodes for next-hop forwarding of packets using close-proximity technology in an NDVN. Signal strength can indicate the proximity of vehicles to each other, while using the same vehicle ID can indicate the stability of the node. Average speed can help to identify vehicles that are moving at a slower pace, which can be an indication of a stable node, while geolocation can provide accurate information about the physical locations of vehicles. When combined, these metrics help identify stable nodes in close proximity to each other, improving the efficiency of content dissemination in the network. For example, if multiple vehicles are identified with the same zone ID, similar geolocation, slow average speed, and high Tss between them, they are more likely to be stable nodes in close proximity and, thus, can be used for content dissemination. Moreover, by considering multiple metrics, the determination of stable nodes can be made more robust and accurate. For instance, if one metric is affected by a specific factor, such as the Tss being influenced by environmental noise, the other metrics can still provide useful information to identify stable nodes. Thus, a combination of metrics can yield a more reliable and comprehensive approach to identifying stable nodes using close-proximity technology. Every node in the network shares its zone ID, geolocation, average speed, Tss, and vehicle ID with its one-hop neighbors. The nodes share these metrics through a piggybacking technique using interest and data packets, which are modified to include this stability information.

Let n be the number of nodes in the network and $w = (w1, w2, w3, w4)$ be the weights associated with the Tss, vehicle speed, GPS values, and vehicle ID, respectively. Let $s = (s1, s2, \ldots, sn)$, $v = (v1, v2, \ldots, vn)$, $p = (p1, p2, \ldots, pn)$, and $vid = (vid1, vid2, \ldots, vidn)$ be the vectors representing the values of these factors, respectively, for each node in the network.

The stability of node i can be defined according to the following function:

$$p(stability_i) = w1 \times si + w2 \times vi + w3 \times pi + w4 \times vidi. \tag{2}$$

For each pair of nodes i and j in the network, the difference in their stability values can be calculated as

$$d = p(stability_i) - p(stability_j). \tag{3}$$

If $d > 0$, node i is more stable than node j. If $d < 0$, node j is more stable than node i. If $d = 0$, the two nodes have similar stability values.

5.8. Weight Determination

The determination of the weights [29] to be given to each metric (i.e., Tss, average speed, geolocation, and vehicle ID) in the weighted linear method depends on the specific application and network conditions. Tss is often considered one of the most important metrics in determining stable nodes using close-proximity technology, as it indicates the strength of the wireless signal between two or more vehicles. A higher Tss can indicate that the vehicles are in close proximity to each other and can communicate more efficiently. Therefore, Tss is usually given a higher weight in the weighted linear proximity approach. Vehicle ID is another important metric for identifying stable nodes, as it can indicate whether two or more vehicles are part of the same platoon or convoy. Vehicles that have the same ID are likely to be in close proximity to each other and can communicate more efficiently. Therefore, vehicle ID is usually also given significant weight in the weighted linear proximity approach. Geolocation is another essential metric, as it provides accurate information about the physical locations of the vehicles, indicating two or more vehicles are in close proximity to each other. However, due to potential technical constraints or related challenges, such as signal inaccuracies, device limitations, or environmental factors, there may be instances where obtaining precise geolocation data becomes difficult. As a result, the weight assigned to geolocation is moderated to reflect the potential uncertainties or limitations associated with its reliable acquisition. Finally, average speed can help identify vehicles that are moving at a slower pace, which can be an indication of a stable node. A slower average speed may also indicate that the vehicle is moving in a platoon or convoy. Average speed is usually given a lower weight in the weighted linear proximity approach since it is just an indication of possible traffic behavior. In a city scenario, the weights assigned to each metric for the weighted linear proximity approach may differ from those assigned in a highway scenario because the characteristics of the environment, traffic flow, and communication conditions can differ significantly between these scenarios.

In a city scenario, there are typically more obstacles, such as buildings and other infrastructure, that can interfere with the wireless signal. This interference can affect the accuracy of the geolocation and signal Tss metrics. Therefore, in this scenario, vehicle ID may be given a higher weight, as it can help identify vehicles that are in close proximity to each other and can communicate more efficiently, even in the presence of obstacles. In a highway scenario, fewer obstacles are present, and vehicles typically travel at higher speeds. This can affect the accuracy of the average speed metric, as calculating the average speed of a vehicle that is traveling at high speed is more difficult. As a result, in a highway scenario, Tss and geolocation may be given higher weights, as they can provide accurate information about the proximity and location of the vehicles.

The weights $w1, w2, w3, w4$ for both city and highway scenarios are determined using the Analytic Hierarchy Process (AHP) method [30]. These steps need to be followed for each scenario:

1. Define the criteria: In this case, the criteria are *Tss* , Vehicle Speed, GPS values, and vehicle ID.
2. Construct pairwise comparison matrices: For each scenario (city and highway), we need to construct a pairwise comparison matrix based on the importance of each criterion relative to the others.
3. Calculate priority vectors (weights) for each criterion.
4. Perform consistency check to ensure the comparisons make logical sense.

Following the previous analysis regarding the significance of various parameters for both city and highway scenarios, the following conclusions have been drawn:

5.8.1. (a) City Scenario Assumptions

- *Tss* is very important due to close proximity communication but might be slightly affected by obstacles.
- Vehicle speed is less critical because of slower average speeds in city traffic.

- GPS values are essential but are affected by obstacles like buildings.
- Vehicle ID is very important for identifying vehicles in platoons or convoys, especially amidst city obstacles.

5.8.2. (b) Highway Scenario Assumptions

- *Tss* and GPS values are crucial due to fewer obstacles and the need for precise location and proximity data.
- Vehicle Speed is somewhat less reliable due to higher speeds, making speed less indicative of stability.
- Vehicle ID remains important for identifying vehicles traveling together but might be slightly less critical than in city scenarios due to the open environment.

From the above assumption, the pairwise comparison matrix for city and highway scenarios is as follows:

City Scenario:

- *Tss* vs. Vehicle speed: *Tss* is more important.
- *Tss* vs. GPS: Comparable importance.
- *Tss* vs. vehicle ID: Slightly less important than vehicle ID.
- Vehicle speed vs. GPS: Less important.
- Vehicle speed vs. vehicle ID: Less important.
- GPS vs. vehicle ID: Slightly less important than vehicle ID.

Highway Scenario:

- *Tss* vs. Vehicle speed: *Tss* is more important.
- *Tss* vs. GPS: Comparable importance, but both are highly important.
- *Tss* vs. vehicle ID: Comparable to vehicle ID.
- Vehicle speed vs. GPS: Less important.
- Vehicle speed vs. vehicle ID: Less important.
- GPS vs. vehicle ID: Slightly less important than GPS.

$$\text{Comparison Matrix (City)} = \begin{bmatrix} 1 & 3 & 2 & \frac{1}{2} \\ \frac{1}{3} & 1 & \frac{1}{2} & \frac{1}{4} \\ \frac{1}{2} & 2 & 1 & \frac{1}{3} \\ 2 & 4 & 3 & 1 \end{bmatrix} \tag{4}$$

This matrix compares Tss, Vehicle Speed, GPS, and vehicle ID against each other for the city scenario. The normalization process adjusts the comparison matrix so that each column sums to 1. The detailed normalized matrix is given by

$$\text{Normalized Matrix} = \begin{bmatrix} 0.261 & 0.3 & 0.308 & 0.24 \\ 0.087 & 0.1 & 0.077 & 0.12 \\ 0.130 & 0.2 & 0.154 & 0.16 \\ 0.522 & 0.4 & 0.462 & 0.48 \end{bmatrix} \tag{5}$$

$$\begin{aligned} \text{Tss} &= \frac{0.26086957 + 0.3 + 0.30769231 + 0.24}{4} = 0.277, \\ \text{Vehicle Speed} &= \frac{0.08695652 + 0.1 + 0.07692308 + 0.12}{4} = 0.096, \\ \text{GPS} &= \frac{0.13043478 + 0.2 + 0.15384615 + 0.16}{4} = 0.161, \\ \text{vehicle ID} &= \frac{0.52173913 + 0.4 + 0.46153846 + 0.48}{4} = 0.466. \end{aligned} \tag{6}$$

The weight for the highway is calculated similarly. Table 2 provides the various weights in both highway and city scenarios used in the QWLCPM approach.

Table 2. Weight Metric Table for Forwarding.

Metrics	City	Highway
Speed	0.096	0.101
Geolocation	0.161	0.351
vehicle ID	0.466	0.360
Tss	0.277	0.188

6. Caching

Caching aims to enhance system performance by temporarily storing frequently or recently accessed data in a high-speed location. These copied data can be accessed quickly without having to retrieve them from the original source every time they are requested.

Priority-based data categorization is crucial for organizing information based on importance and requirements. Dynamic cache partitioning is employed to store data in the cache according to priority, enhancing the hit ratio and content diversity. Each priority section's limit in the partition is adjusted based on the received data and available space. Temporal-based data eviction aims to achieve the timely removal of invalid data from the cache. When the time validity expires for data, they become invalid and should be promptly removed. For instance, accident data may expire in 30 s, while work zone warnings can last all day, as listed for the various applications listed in Table 1. As a result, the fundamental concept is to correlate time with each form of data.

Safety and non-safety applications have certain spatial validity, as indicated in Table 1. For instance, car accident warnings are valid for up to 1 km, and emergency warnings have a validity of 500 m. When the spatial validity expires, the priority of the data decreases, becoming potential replacement candidates. They are not immediately removed from the cache, as they may still have use beyond their spatial validity. Spatial validity is significant in NDN, as it enables content providers to control data dissemination and consumption. It allows for restricted access to specific geographic regions, reduced network traffic, and conserved resources by disseminating data only where relevant.

To determine which data to cache, we use a hybrid approach, which considers factors such as stability, priority, and zone ID. Packets from stable nodes are given more weight than those from less stable ones, higher-priority data are given more weight, and packets from the same zone are also considered when determining the cache replacement probability.

For each node i in the network, its stability value is obtained:

1. Calculate the cache replacement priority $P(\text{Priority}_i)$ as follows:
   (a) If the priority of node i is high,
   $P(\text{Priority}_i) = 0.9$
   (b) If the priority of node i is medium,
   $P(\text{Priority}_i) = 0.5$
   (c) If the priority of node i is low,
   $P(\text{Priority}_i) = 0.1$
2. If the node i is in the same zone as its request,
   $w(\text{zone}) = 1$
3. If the node i is in a different zone from its request,
   $w(\text{zone}) = 0.5$.

The different weights assigned to the priority of nodes aim to keep the most appropriate data in the cache and evict the least important data first. The zone ID is used to determine the cache priority of nodes in different zones from their request, such that nodes closer to their request have higher cache priority. For each datum in the network, a temporal value is associated. When this temporal value expires, the datum is evicted from the cache immediately, regardless of its cache priority or any other factor. $P(Stable\_Cache\_i)$,

$P(Priority\_i)$, and w_zone are combined into a WLC equation to calculate the final cache replacement probability for each datum in the network as follows:

$$Cache\_Factor = w\_zone \times w\_priority \times w\_stability \times p\_stable\_cache\_i \times p\_priority\_i, \quad (7)$$

$$Caching\_Probability(i) = Cache\_Factor \times (1 - e^{-(t - t\_expire\_i)/T}). \quad (8)$$

Equation (5) maps to a caching probability between 0 and 1, where a high caching probability corresponds to a high probability of being cached and a low final score corresponds to a low probability of being cached. This can be used as a caching probability function to determine which data should be replaced from the cache. The weighting factors for stability, priority, and zone vary depending on the specific caching strategy. However, generally speaking, stability and priority tend to be more influential than zone in caching decisions. Therefore, stability and priority are given weights of 0.8 and 0.7, respectively, while zone is given a weight of 0.4.

7. Content Diversity and Hit Ratio

Content diversity is an important aspect of vehicular networks, as it can improve the overall performance of the network and enhance the user experience. Content diversity refers to the variety of content available to the users, which can be improved by various means, such as content caching, content replication, and content discovery. Temporal diversity refers to the variety of content that is cached over time. A high temporal diversity indicates that the content diversity is high, as the nodes cache different content at different times. To calculate temporal diversity, we can analyze the content caching patterns over time and measure how often the nodes cache new and diverse content. In summary, content diversity in priority caching in vehicular networks can be calculated by measuring the variety of content stored in the nodes and how well the content serves the needs of different users and applications. Temporal diversity TD in priority caching can be measured by calculating the variation in the content that is cached over time:

$$TD = 1 - \frac{S}{N}, \quad (9)$$

where S is the number of content items that are cached multiple times over a given time period, and N is the total number of content items cached during that time period. This equation calculates the percentage of content items that are cached only once during the time period, indicating how diverse the content is over time. If all the content items are cached exactly once during the time period, then the temporal diversity is maximal (i.e., TD = 1). If some of the content items are cached multiple times, then the temporal diversity decreases, as the content diversity over time is lower.

The often complex relationship between hit ratio and content diversity in a caching system depends on several factors, such as the cache-replacement policy, the popularity distribution of the content, and the diversity metric used. On the one hand, increasing content diversity can reduce the hit ratio of a caching system, especially if the cache-replacement policy is not designed to prioritize popular or frequently accessed content items. This is because caching more diverse content items can lead to a higher likelihood of cache misses, as each vehicle may request a different set of items not present in the cache. On the other hand, content diversity can improve the hit ratio of a caching system if it is combined with an effective cache-replacement policy that considers the priority and access patterns of the content. By caching a diverse set of priority and frequently accessed items, a caching system can reduce the overall number of cache misses and improve the hit ratio. Therefore, the effect of content diversity on the hit ratio depends on the specific caching strategy used and the nature of the content being cached. To evaluate the impact of content diversity on hit ratio, the following assumptions are considered:

- The cache has a fixed size and can store a limited number of content items.

- The content items have different priority levels, represented by a probability distribution function.
- Vehicles request content items according to a probability-distribution function that depends on the priority of the items and their diversity.

Under these assumptions, we can define the following parameters:

- C—the cache size in terms of the number of content items;
- $P(p)$—the popularity distribution function of the content items, where p is the popularity level;
- $F(d, p)$—the probability distribution function of content requests for each vehicle, where d is the diversity level and p is the priority level;
- H—the hit ratio, defined as the fraction of content requests that are satisfied by the cache.

To model the impact of content diversity on the hit ratio, we can use the following equation:

$$H(d) = \sum_{p} \min(C, N(p, d)) \times F(d, p), \tag{10}$$

where $N(p, d)$ is the number of content items with priority level p that are unique to the set of content items requested by vehicles with diversity level d, and the function $\min(C, N(p, d))$ represents the number of content items that can be cached in the cache given its size constraint, which is employed to limit the number of content items considered for caching based on the cache size constraint (C), i.e., the capacity of the cache. If $N(p, d)$ exceeds the cache size C, the min function will result in C to ensure that the calculation remains within the cache's storage limit. We can then compare the hit ratio for different diversity levels to evaluate the impact of content diversity on caching performance. Specifically, we can calculate the change in hit ratio ΔH as a function of the change in diversity Δd using the following equation:

$$\Delta H / \Delta d = \frac{H(d_2) - H(d_1)}{d_2 - d_1}, \tag{11}$$

where d_1 and d_2 are two different diversity levels.

Spatial validity refers to the idea that the relevance of content items may depend on the vehicle location and the context of the request. For example, a warning message about road conditions may be relevant only to vehicles within a certain range of the issue. To incorporate spatial validity into the hit ratio equation, we can modify the probability distribution function of content requests to include a spatial component. Specifically, we can define $F(d, p, x)$ as the probability distribution function of content requests for each vehicle with diversity level d, popularity level p, and spatial context x. We can then modify the equation for hit ratio to include a spatial constraint as follows:

$$H(d, x) = \sum_{p} \min(C, N(p, d, x)) \times F(d, p, x), \tag{12}$$

where $N(p, d, x)$ is the number of content items with priority level p and diversity level d that are unique to the set of content items requested by vehicles within a certain spatial context x. The specific definition of $N(p, d, x)$ may depend on the spatial constraint being considered. For example, if the spatial validity of a warning message is 1 km, $N(p, d, 1)$ as the number of content items with priority level p and diversity level d that are unique to the set of content items requested by vehicles within a 1 km radius of the location where the warning was issued.

To modify the hit ratio equation based on zones, we can incorporate the zone separation factor Z_j, which indicates whether nodes i and j are in the same zone, into the original hit ratio equation:

$$H(d,x) = \frac{1}{N}\sum_{i=1}^{N}[(1-S_i)(1-D_i)Z_j H(d_i,x)], \tag{13}$$

where $H(d,x)$ is the hit ratio of content with descriptor d and spatial context x, N is the number of nodes in the network, S_i is the spatial validity of node i, D_i is the temporal diversity of node i, Z_j is the zone separation factor of zone j, and $H(d_i,x)$ is the hit ratio of content with descriptor d_i and spatial context x.

In this modified equation, Z_j is multiplied by $H(d_i,x)$ to obtain the effective hit ratio contribution of node i based on the zone separation factor. If nodes i and j are in the same zone (i.e., their distances are less than or equal to the RSU range, and $Z_j = 1$), then the hit ratio contribution of node i is fully considered; however, if nodes i and j are in different zones (i.e., their distances are greater than the RSU range, and $Z_j = 0$), the hit ratio contribution of node i is considered zero, indicating that the content provided by node i is not relevant to the current zone. The zone-separation factor can be calculated using the same method as described in the modified content diversity equation. By incorporating the zone separation factor into the hit ratio equation, we can account for the impact of zone separation on the relevance of content in vehicular networks.

To modify content diversity based on zone separation using the RSU range to define zones, we can add another parameter to calculate content diversity CD that employs the RSU range as the zone:

$$CD = \frac{1}{N}\sum_{i=1}^{N}[(1-S_i)(1-D_i)Z_j]. \tag{14}$$

8. Simulation Environment

8.1. Methodology

We carried out a simulation on vehicular networks using the simulation tools ndnSIM and SUMO [31] to evaluate the performance of QWLCPM in city and highway scenarios with 20 simulation repetitions for each scenario. We simulated Manhattan models, which consisted of five vertical and horizontal bidirectional streets with an area of 2 × 2 km for the city and a 4 km long stretch for the highway case.

As the vehicular network is divided into zones with respect to the RSU transmission range, RSUs were deployed along the streets at 300 m intervals with network sizes ranging from 100 to 1000 vehicles in the city scenario and 100–300 vehicles per square kilometer in the highway scenario. The QWLCPM performance was compared with the least recently used (LRU), dynamic cooperative cache management scheme based on social and popular data (DCCMS) and least frequently used (LFU) approaches in terms of cache hit ratio, content delivery ratio, and interest satisfaction ratio (ISR). LRU is a caching algorithm that removes the least recently used items from the cache first. The basic idea is to keep track of the order in which elements are accessed. When the cache reaches its capacity and a new item needs to be stored, the algorithm identifies the least recently used item and replaces it with the new one. LRU is based on the assumption that items that have been accessed recently are more likely to be accessed again in the near future. DCCMS introduces an innovative approach to a cache-management technique that prioritizes content based on popularity and social interactions among nodes. It incorporates a master node concept for hierarchical collaboration and content distribution, focusing on maximizing the use of cache resources and minimizing content delivery latency [17]. LFU is a caching algorithm that removes the least frequently used items from the cache first. It keeps track of how often each item in the cache is accessed. When the cache reaches its capacity and a new item needs to be stored, the algorithm identifies the least frequently used item and replaces it with the new one. LFU assumes that items that have been accessed less frequently are less likely to be accessed again in the future [32]. For delay and hop count, QWLCPM was assessed based on the best route strategy of the basic NDN [33] (bNDN, i.e., the default forwarding strategy of the named data network); in particular, bNDN forwards an interest to the upstream with the lowest routing cost. The simulation parameters for both city and highway scenarios are listed in Table 3.

We utilized the following performance measures:

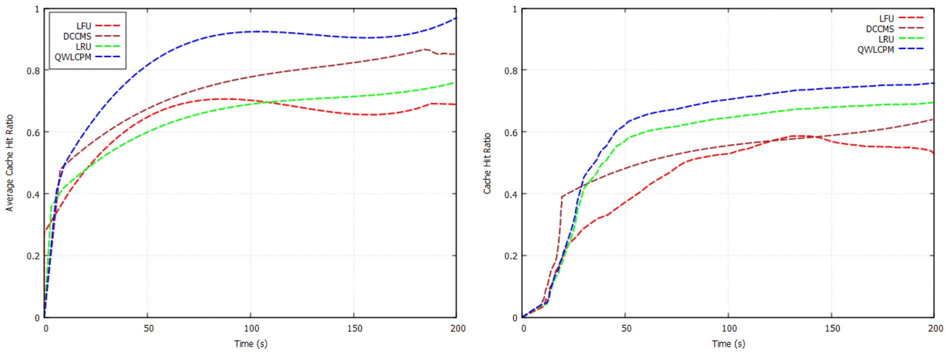
- Cache hit ratio. The cache hit ratio is a measure of how well a cache performs in a computer system, indicating whether requested data are already stored in the cache.
- Interest satisfaction ratio. In an NDVSN, the ISR is a performance metric used to measure the efficiency of content retrieval in the network. It is calculated as the ratio of the number of interests that are successfully satisfied with a corresponding data packet to the total number of interests sent. A higher ISR means that a larger percentage of interests were successfully satisfied, indicating better performance of the network in terms of content retrieval.
- Hop count. Hop count is an essential metric used in NDN, which measures the number of intermediate network nodes through which a data packet has to pass to reach its destination. The hop count affects the overall performance and efficiency of the network. A higher hop count indicates that data packets have to travel through more intermediate nodes, which can increase the latency and delay in delivering the data. In addition, a higher hop count can increase the likelihood of packet loss or congestion, which can further degrade the network performance.
- Delay. Delay in NDN refers to how long it takes for a data packet to traverse the network from the time it is requested until the time it is received by the requester. Delay is an important performance metric in NDN, as it affects the QoS experienced by users and applications that rely on timely access to data.
- Content diversity. Content diversity in NDN refers to the variety of unique content items that are available in the network. In NDN, data are requested and retrieved based on their names rather than locations or addresses. This means that the network can support a wide range of content types and formats, and users can request and receive specific content items directly from the network.

Table 3. Simulation Parameters.

Parameter	City	Highway
Vehicle density	Dense	Sparse
No. of vehicles per km^2	100–1000	100–300
Simulation area	2 km^2	4 km highway stretch
Vehicle speed (km/h)	0–50	0–100
Hello packet gen time (s)	1	1
Simulation time (s)	180	180
RSU transmission range (m)	250	250
Vehicle transmission range (m)	50	50
Content size (kb)	1024	1024
Priority	High, medium, and low	High, medium, and low
Interest frequency (packets/s)	5/10	5/10
Packet priority distribution	Random	Random
Simulation Repetition	20	20
Mobility model	Manhattan	Manhattan

8.2. Performance Evaluation

The cache hit ratio is illustrated in Figure 3, and QWLCPM was found to have a higher hit ratio than LRU and LFU in both highway and city scenarios. This result is because QWLCPM considers various factors simultaneously and makes informed caching decisions. It also employs a spatial and temporal component to remove data from the cache in a timely manner, keeping the cache effective and enabling new data to be cached as needed. In Figure 3, LRU and LFU initially outperformed the other one at various times, but as vehicle density increased with time, LRU eventually outperformed LFU in both scenarios. The DCCMS quickly shows significant improvement and continues to outperform other traditional algorithms throughout. This indicates that DCCMS effectively identifies and caches content that is in high demand in the city environment, which is likely more dynamic. Its superior performance suggests that the factors it considers, possibly including speed, density, and social data, are highly relevant for a highway scenario. From both environments, QWLCPM shows the most consistent and highest performance in terms of cache hit ratio. This implies that QWLCPM is well suited to environments with either relatively stable or highly dynamic content requests.

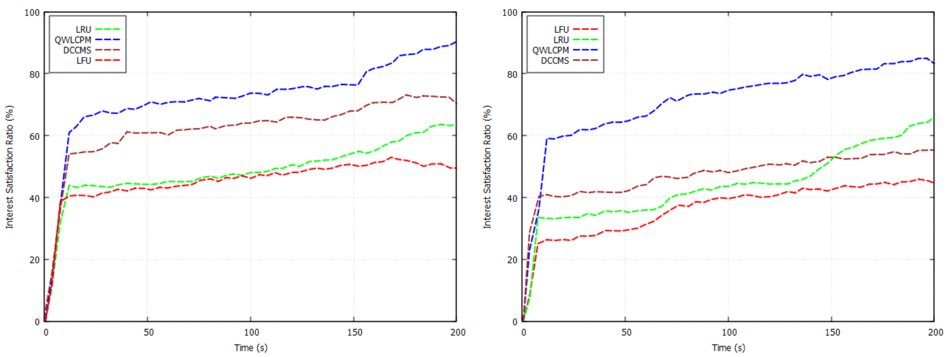


(**a**) (**b**)

Figure 3. Cache hit ratio: (**a**) city scenario; (**b**) highway scenario.

The zone-based caching approach utilized by QWLCPM allows caching decisions to be made based on the location of the requesting node, which can improve the content-delivery ratio. QWLCPM identifies stable nodes based on their proximity using a weighted linear approach, and these nodes are essential in forwarding data from the source to the destination. By using stable nodes as intermediaries, QWLCPM can improve ISR, reduce delay, and decrease the hop count, as depicted in Figures 4–6. However, the performances of QWLCPM, LFU, and LRU were worse under the highway scenario compared to the city scenario. DCCMS is highly effective in city environments where social interactions and content popularity are significant factors. However, on the highway, its performance becomes more comparable to other algorithms, possibly because the factors influencing content popularity may shift more rapidly due to the fast-moving nature of vehicles. Overall, the interest satisfaction ratios reflect the adaptability and effectiveness of DCCMS and QWLCPM in dynamic environments, with DCCMS particularly excelling in the city context.

Additionally, QWLCPM takes into account the proximity of the data source and the requesting vehicle when making caching decisions, such that data are more likely to be cached in nearby nodes. This improved the cache hit ratio, delay, and hop count, as shown in Figure 5.

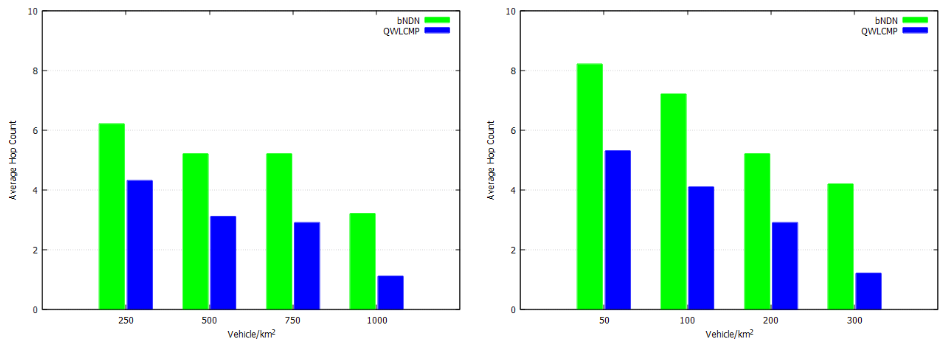


(**a**) (**b**)

Figure 4. Interest satisfaction ratio: (**a**) city scenario; (**b**) highway scenario.

In the highway scenario, characterized by sparse traffic and high vehicle speeds, the delay and ISR tended to increase, resulting in a low cache hit ratio and a larger hop count. However, QWLCPM outperformed both LRU and LFU in terms of all performance metrics. By utilizing an offloader and priority partitioning with RSU, the hit ratio increased since high-speed vehicles could offload high-priority content before leaving the zone. QWLCPM utilizes the priority-wise partitioning of the cache, enhancing the likelihood of high-priority data being cached while minimizing the chances of replacement, thereby increasing the hit ratio and reducing the delay, as shown in Figure 6. Another benefit is that the cache is not dominated by a few frequently accessed items, leading to better content diversity, as shown in Figure 7.

QWLCPM uses RSUs as zone separators, dividing the vehicular network into smaller zones. As a result, QWLCPM reduces the number of nodes that need to be searched for cached content, which decreases the delay in content delivery and improves the overall hop count, as shown in Figure 6.



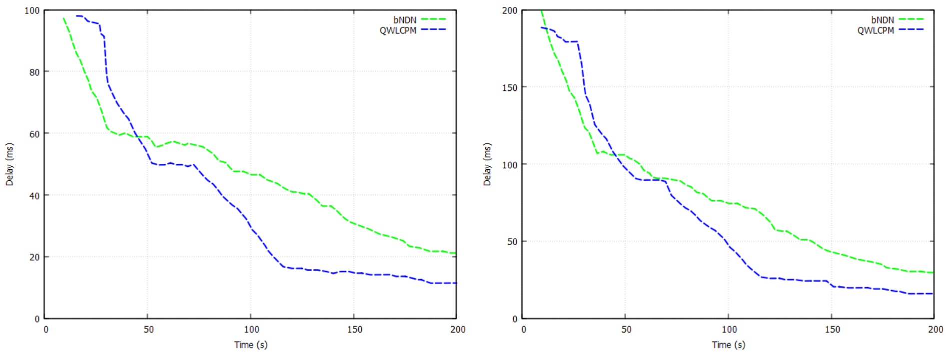
(**a**) (**b**)

Figure 5. Hop count: (**a**) city scenario; (**b**) highway scenario.

Figure 6 shows that the content diversity for the LFU cache-replacement policy is likely to increase with vehicle density since LFU replaces the least frequently used item in the cache with a new item. Over time, the cache retains recently accessed items while replacing less frequently accessed items with new ones. Therefore, the cache has a higher probability of containing less frequently accessed items and a lower probability of containing frequently accessed items. As a result, the content diversity of the cache increases over time, leading to a lower cache hit rate for frequently accessed items and a lower hit ratio, as shown in Figure 7. However, QWLCPM uses priority-based partitioning to increase the hit ratio, leading to an increase in content diversity. QWLCPM appears to achieve a balance between maintaining a high hit ratio and preserving content diversity. It starts off on par with other strategies but then overtakes them, suggesting that QWLCPM is effective at managing cache space to include a diverse set of contents without significantly sacrificing the hit ratio. DCCMS follows a similar trend to QWLCPM but with slightly less performance in both metrics, indicating that while it is effective, it might prioritize popular content slightly over diversity compared to QWLCPM. LFU and LRU both show lower performance in maintaining a balance between hit ratio and content diversity compared to QWLCPM, with LFU, in particular, falling behind as diversity increases. This outcome is consistent with LFU's strategy of favoring frequently accessed content, which can result in lower diversity.

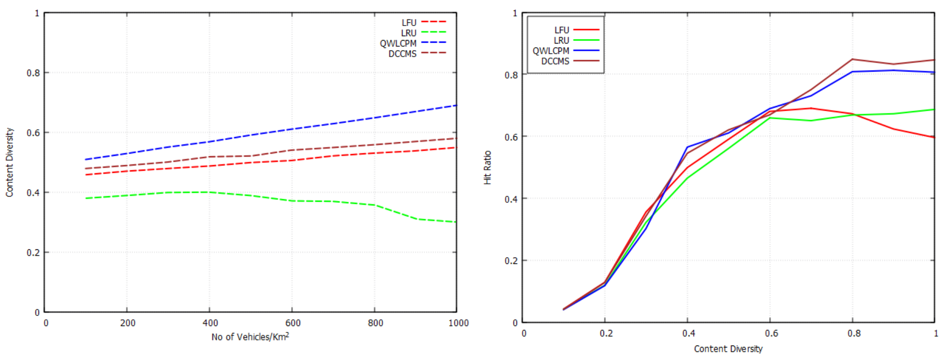
The increased delay in the highway scenario was due to network partitions and communication disruption resulting from the larger average distances between vehicles traveling at high speeds, and the increase in hello packets was due to the sparsity of the traffic.

Overall, the use of RSUs as zone separators, priority-wise partitioning of the cache, and proximity-based caching in QWLCPM worked together to improve the cache hit ratio, delay, hop count, ISR, and content delivery ratio in vehicular networks, thereby enhancing QoS. By optimizing caching decisions and improving overall network efficiency, QWLCPM can achieve better network performance compared to traditional caching algorithms, such as LRU, LFU, and bNDN.



(**a**) (**b**)

Figure 6. Delay/latency. (**a**) City scenario; (**b**) highway scenario.



(**a**) (**b**)

Figure 7. Content diversity and hit ratio for the city scenario. (**a**) content diversity; (**b**) hit ratio vs. content diversity.

The 95% confidence interval for the mean cache hit ratio for QWLCPM across all simulation runs is approximately from 0.456 to 0.555. This means we can be 95% confident that the true mean cache hit ratio of the population from which these simulations were drawn falls within this range. The Kernel Density Estimation (KDE) graph shown in Figure 8 suggests that the average cache hit ratios from the simulations have a distribution with a clear central tendency and low variability, following an approximately normal distribution. A narrow peak, as seen here, suggests low variance, meaning that the average cache hit ratios do not vary widely across the different simulations.

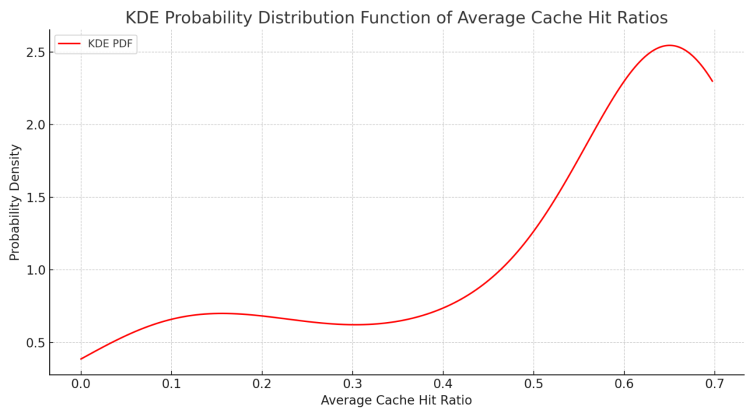


Figure 8. KDE probability distribution function.

Figure 9 displays a visual representation of packet loss percentages in relation to various RSU transmission ranges (100 m to 500 m) and vehicle transmission ranges (10 m to 50 m). As the RSU transmission range increases from 100 m to 500 m, there is an evident trend where packet loss initially decreases but then begins to increase. At 300 m, the packet loss appears to be the lowest for all vehicle transmission ranges, which may indicate an optimal balance between coverage area and signal reliability in an urban environment. For shorter vehicle transmission ranges (10 m and 20 m), packet loss is generally lower across all RSU transmission ranges. As the vehicle transmission range increases to 30 m and above, there is an upward trend in packet loss, indicating that the longer the signal needs to travel between vehicles, the higher the likelihood of packet loss. The highest packet loss is observed at the 500 m RSU transmission range across all vehicle transmission ranges, which could be due to signal degradation over longer distances or increased interference in a city environment with numerous obstacles. The lowest packet loss occurs in the mid-range RSU transmission ranges (200 m to 400 m), suggesting that these ranges may offer a more stable communication link in urban areas where buildings and other structures can impact signal strength. These trends suggest that in a city environment, where obstacles such as buildings can interfere with wireless signals, there is an optimal RSU transmission range that minimizes packet loss while still providing adequate coverage. It also indicates that vehicle transmission ranges need to be managed carefully to prevent increased packet loss, especially as vehicles become more dispersed or when they face more complex urban layouts.

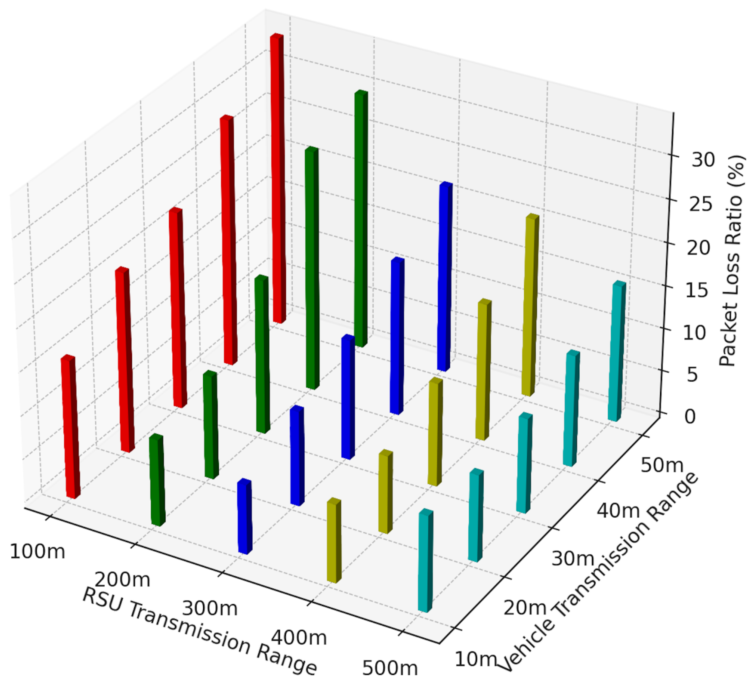


Figure 9. Packet loss ratio across RSU and vehicle transmission ranges (city).

The graph shown in Figure 10 represents packet loss ratios at increasing RSU transmission ranges (from 300 m to 700 m) for various vehicle transmission ranges (from 10 m to 50 m) and vehicle densities (from 250 Veh/km^2 to 1000 Veh/km^2). As the RSU transmission range increases, the packet loss ratio tends to increase. This trend may suggest that while higher RSU ranges can cover more distance, they might also lead to higher packet loss, potentially due to signal degradation over longer distances or increased chances of communication interruptions at higher speeds, which are typical of highway driving. Within each RSU transmission range, as the vehicle transmission range increases, packet loss also increases slightly. This could be due to the larger area over which signals must be disseminated, which can be more challenging at high vehicle speeds on highways. Based on this graph, the RSU transmission range of 300 m shows the lowest packet loss across all vehicle densities, suggesting it may be an optimal setting for highway environments, balancing coverage and communication reliability. The higher RSU ranges (600 m and 700 m), while covering more distance, might not be as effective due to increased packet loss. A moderate RSU transmission range, perhaps around 300 m to 500 m, could be ideal for highway scenarios to ensure effective DSRC communication.

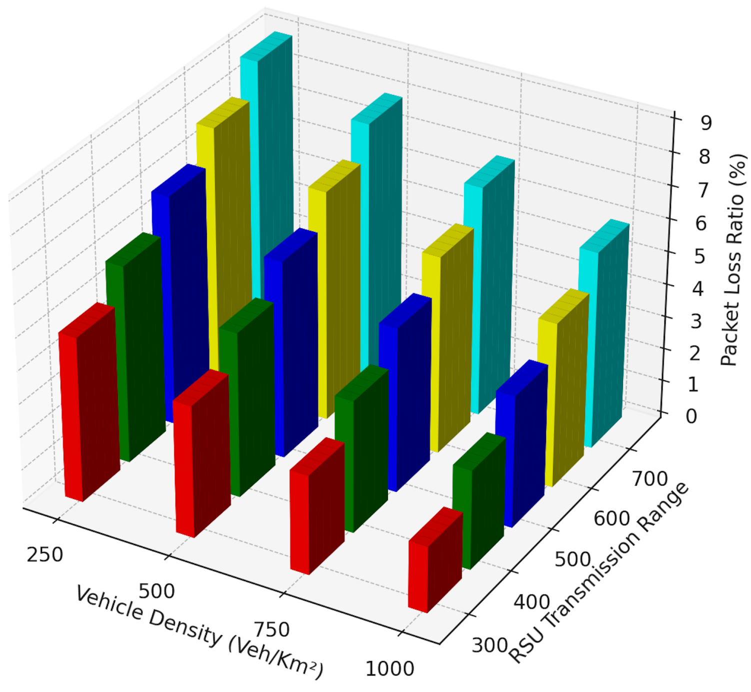


Figure 10. Packet loss ratio across RSU and vehicle transmission ranges (highway).

Metrics evaluation, weighted linear combination, decision-making (forwarding and caching), and cache management form the core components of the algorithm's approach to optimizing data forwarding and cache management in vehicular networks. Initially, the algorithm evaluates the essential metrics of signal strength, vehicle speed, position, and vehicle ID to determine node stability and optimal data forwarding paths, with an evaluation time complexity of O(n), where n signifies the count of nodes within the transmission range. By applying a weighted linear combination method, it computes a stability score for each node through linear operations on evaluated metrics, ensuring the computation complexity remains O(n) due to the fixed number of metrics and individual node processing. The decision-making phase, concerning packet forwarding and caching, is based on these computed stability scores alongside QoS requirements. This phase is approximated to have a complexity of O(n) for forwarding decisions and O(c) for caching, where c represents available cache slots. Cache management, involving new content insertion and outdated content eviction guided by QoS priorities and popularity, ranges in complexity from O(c) to O(clogc), depending on the employed strategy, from simple replacements to complex criteria-based sorting. Thus, the overall time complexity per decision-making cycle of the QWLCPM algorithm simplifies to O(n + clogc), highlighting its scalability with the number of nodes and logarithmic complexity concerning cache size, indicating efficiency in diverse vehicle environments and scalability in cache management. The cache management in DCCMS can be more sophisticated, involving not only QoS priorities but also the popularity and social dynamics of content. This could lead to a more involved process with complexity varying from O(c) for simple insertions to O(clogc) for sorting based on complex criteria, similar to QWLCPM. the overall time complexity per decision-making cycle for DCCMS could be on par with or slightly higher than QWLCPM, considering additional factors in the decision-making process. The complexity would remain scalable with the number of nodes and logarithmically concerning cache size.

Comparison and conclusion meld into a singular analysis emphasizing QWLCPM's nuanced approach towards real-time performance, computational complexity, and suitability for vehicular networks. Unlike LRU and LFU methods, which prioritize efficiency with a computational complexity of O(1) under ideal conditions, QWLCPM demonstrates a higher computational overhead, primarily due to its sophisticated decision-making process, with an overall complexity of O(n + clogc). This complexity underlines QWLCPM's capacity to adaptively manage the cache based on QoS requirements and fluctuating network conditions, potentially offering better performance in scenarios marked by dynamic and heterogeneous conditions. The algorithm's design explicitly caters to vehicular networks, where decisions must consider multiple dynamic factors such as node stability and QoS priorities, rendering it more effective in these contexts despite its higher computational demand. Therefore, QWLCPM stands out for specialized applications that demand consideration of various dynamic factors, aiming to enhance QoS in vehicular networks, whereas LRU and LFU are optimized for general caching scenarios with minimal computational overhead. The selection between these caching strategies depends on the network environment's specific requirements and the primary objectives—optimizing computational efficiency or enhancing real-time network performance and QoS. DCCMS is likely to have a higher computational overhead than LRU and LFU, which have a complexity of O(1) under ideal conditions. This is due to the multi-faceted approach DCCMS takes, considering a broader range of metrics for decision-making.

9. Conclusions

This paper presents a method for caching and forwarding data in an NDVSN. The proposed approach uses a simple WLC and proximity method to optimize QoS. Through simulations in both highway and city scenarios, the proposed QWLCMP strategy demonstrates superior performance compared to traditional caching strategies like LRU, LFU, DCCMS, and bNDN, showcasing improved hit ratios, ISR, and hop counts. The zoning based on the RSU transmission range further refines the approach, proving its efficacy in urban environments. However, the study acknowledges the need for further enhancement in highway scenarios and emphasizes the importance of a comparative analysis with alternative caching and forwarding strategies in future research. QWLCPM's proximity consideration significantly improves cache performance metrics, reducing delays and optimizing hop counts, especially in scenarios with low traffic density. Its priority-based caching mechanism notably enhances hit ratios and content diversity, showcasing its substantial potential for network performance enhancement in vehicular environments. Looking ahead, we suggest further exploring and implementing QWLCPM in real-world vehicular networks due to its adaptability and demonstrated effectiveness. Future research avenues include refining implementation details, assessing scalability in larger networks, security and conducting real-world trials to validate performance under dynamic conditions. The potential integration of QWLCPM into existing vehicular network infrastructures holds promise for significant improvements, warranting comprehensive exploration and potential adoption in future network architectures.

Author Contributions: Conceptualization, D.D.; methodology, D.D. and K.S.; software, D.D.; validation, D.D. and K.S.; formal analysis, D.D.; investigation, D.D.; resources, D.D. and K.S.; data curation, D.D.; writing—original draft preparation, D.D.; writing—review and editing, D.D. and K.S.; visualization, D.D. and K.S; supervision, K.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Jacobson, V.; Smetters, D.K.; Thornton, J.D.; Plass, M.F.; Briggs, N.H.; Braynard, R.L. Networking named content. In Proceedings of the 5th International Conference on Emerging Networking Experiments and Technologies, Rome, Italy, 1–4 December 2009; pp. 1–12.
2. Liu, X.; Li, Z.; Yang, P.; Dong, Y. Information-centric mobile ad hoc networks and content routing: A survey. *Ad Hoc Netw.* **2017**, *58*, 255–268. [CrossRef]
3. Rashid, S.A.; Hamdi, M.M.; Alani, S. An overview on quality of service and data dissemination in VANETs. In Proceedings of the 2020 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA), Ankara, Turkey, 26–27 June 2020; pp. 1–5.
4. Zhang, L.; Afanasyev, A.; Burke, J.; Jacobson, V.; Claffy, K.C.; Crowley, P.; Papadopoulos, C.; Wang, L.; Zhang, B. Named data networking. *ACM SIGCOMM Comput. Commun. Rev.* **2014**, *44*, 66–73. [CrossRef]
5. Al-Omaisi, H.; Sundararajan, E.A.; Alsaqour, R.; Abdullah, N.F.; Abdelhaq, M. A survey of data dissemination schemes in vehicular named data networking. *Veh. Commun.* **2021**, *30*, 100353. [CrossRef]
6. Saxena, D.; Raychoudhury, V.; Suri, N.; Becker, C.; Cao, J. Named data networking: A survey. *Comput. Sci. Rev.* **2016**, *19*, 15–55. [CrossRef]
7. Amadeo, M.; Campolo, C.; Molinaro, A. Named data networking for priority-based content dissemination in VANETs. In Proceedings of the 2016 IEEE 27th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC), Valencia, Spain, 4–8 September 2016; pp. 1–6.
8. Englund, C.; Chen, L.; Vinel, A.; Lin, S.Y. Future applications of VANETs. In *Vehicular Ad Hoc Networks: Standards, Solutions, and Research*; Springer: Berlin/Heidelberg, Germany, 2015; pp. 525–544.
9. Khelifi, H.; Luo, S.; Nour, B.; Moungla, H. A QoS-aware cache replacement policy for vehicular named data networks. In Proceedings of the 2019 IEEE Global Communications Conference (GLOBECOM), Big Island, HI, USA, 9–13 December 2019; pp. 1–6.
10. Singh, P.K.; Nandi, S.K.; Nandi, S. A tutorial survey on vehicular communication state of the art, and future research directions. *Veh. Commun.* **2019**, *18*, 100164. [CrossRef]
11. Ahed, K.; Benamar, M.; Lahcen, A.A.; El Ouazzani, R. Forwarding strategies in vehicular named data networks: A survey. *J. King Saud Univ. Comput. Inf. Sci.* **2022**, *34*, 1819–1835. [CrossRef]
12. Chen, C.; Wang, C.; Qiu, T.; Atiquzzaman, M.; Wu, D.O. Caching in vehicular named data networking: Architecture, schemes and future directions. *IEEE Commun. Surv. Tutor.* **2020**, *22*, 2378–2407. [CrossRef]
13. Tariq, A.; Rehman, R.A.; Kim, B.S. Forwarding strategies in NDN-based wireless networks: A survey. *IEEE Commun. Surv. Tutor.* **2019**, *22*, 68–95. [CrossRef]
14. Abdelaal, M.; Karadeniz, M.; Dürr, F.; Rothermel, K. liteNDN: QoS-aware packet forwarding and caching for named data networks. In Proceedings of the 2020 IEEE 17th Annual Consumer Communications & Networking Conference (CCNC), Las Vegas, NV, USA, 10–13 January 2020; pp. 1–9.
15. Wang, S.; Wu, W.; Zhi, Z. Varied priority-based data forwarding via NDN in VANET. In Proceedings of the 2019 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI), Leicester, UK, 19–23 August 2019; pp. 574–581.
16. Huang, W.; Song, T.; Yang, Y.; Zhang, Y. Cluster-based cooperative caching with mobility prediction in vehicular named data networking. *IEEE Access* **2019**, *7*, 23442–23458. [CrossRef]
17. Ashraf, M.; Huang, C.; Raza, K.A.; Huang, S.; Yin, Y.; Wu, D.F. Dynamic cooperative cache management scheme based on social and popular data in vehicular named data network. *Wirel. Commun. Mob. Comput.* **2022**, *2022*, 8374181. [CrossRef]
18. Deng, G.; Wang, L.; Li, F.; Li, R. Distributed probabilistic caching strategy in VANETs through named data networking. In Proceedings of the 2016 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), San Francisco, CA, USA, 10–14 April 2016; pp. 314–319.
19. Grassi, G.; Pesavento, D.; Pau, G.; Zhang, L.; Fdida, S. Navigo: Interest forwarding by geolocations in vehicular named data networking. In Proceedings of the 2015 IEEE 16th International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM), Boston, MA, USA, 14–17 June 2015; pp. 1–10.
20. Deng, G.; Xie, X.; Shi, L.; Li, R. Hybrid information forwarding in VANETs through named data networking. In Proceedings of the 2015 IEEE 26th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC), Hong Kong, China, 30 August–2 September 2015; pp. 1940–1944.
21. Boukerche, A.; Coutinho, R.W.; Yu, X. LISIC: A link stability-based protocol for vehicular information-centric networks. In Proceedings of the 2017 IEEE 14th International Conference on Mobile Ad Hoc and Sensor Systems (MASS), Orlando, FL, USA, 22–25 October 2017; pp. 233–240.
22. Ahmed, S.H.; Bouk, S.H.; Kim, D. RUFS: RobUst forwarder selection in vehicular content-centric networks. *IEEE Commun. Lett.* **2015**, *19*, 1616–1619. [CrossRef]
23. Wang, X.; Liu, W.; Yang, L.; Zhang, W.; Peng, C. A new content-centric routing protocol for vehicular ad hoc networks. In Proceedings of the 2016 22nd Asia-Pacific Conference on Communications (APCC), Yogyakarta, Indonesia, 25–27 August 2016; pp. 552–558.

24. Wang, J.; Luo, J.; Zhou, J.; Ran, Y. A Mobility-Predict-based Forwarding Strategy in Vehicular Named Data Networks. In Proceedings of the GLOBECOM 2020–2020 IEEE Global Communications Conference, Taipei, Taiwan, 7–11 December 2020; pp. 1–6.
25. Liu, X.; Nicolau, M.J.; Costa, A.; Macedo, J.; Santos, A. A geographic opportunistic forwarding strategy for vehicular named data networking. In Proceedings of the Intelligent Distributed Computing IX, Paris, France, 10–12 October 2016; Springer: Cham, Switzerland, 2016; pp. 509–521.
26. Khelifi, H.; Luo, S.; Nour, B.; Moungla, H.; Faheem, Y.; Hussain, R.; Ksentini, A. Named data networking in vehicular ad hoc networks: State-of-the-art and challenges. *IEEE Commun. Surv. Tutor.* **2019**, *22*, 320–351. [CrossRef]
27. Kouonchie, P.K.; Oduol, V.; Nyakoe, G.N. Roadside Units for Vehicle-to-Infrastructure Communication: An Overview. In Proceedings of the Sustainable Research and Innovation Conference, Nairobi, Kenya, 5–7 April 2022; pp. 69–72.
28. Dhakal, D.; Rathor, M.; Dey, S.; Dey, P.; Sharma, K. ASCTWNDN: A Simple Caching Tool for Wireless Named Data Networking. In *Contemporary Issues in Communication, Cloud and Big Data Analytics*; Springer: Singapore, 2022; pp. 139–147.
29. Zhong, H.; Geng, Y.; Cui, J.; Xu, Y.; Liu, L. A weight-based conditional privacy-preserving authentication scheme in software-defined vehicular network. *J. Cloud Comput.* **2020**, *9*, 54. [CrossRef]
30. Saaty, T.L. Decision making—The analytic hierarchy and network processes (AHP/ANP). *J. Syst. Sci. Syst. Eng.* **2004**, *13*, 1–35. [CrossRef]
31. Araujo, G.; Peixoto, M.; Sampaio, L. NDN4IVC: A framework for simulations of realistic VANETs applications through NDN. In Proceedings of the 9th ACM Conference on Information-Centric Networking, Osaka, Japan, 19–21 September 2022; pp. 162–164.
32. Silva, E.T.d.; Macedo, J.M.H.d.; Costa, A.L.D. NDN content store and caching policies: Performance evaluation. *Computers* **2022**, *11*, 37. [CrossRef]
33. Afanasyev, A.; Shi, J.; Zhang, B.; Zhang, L.; Moiseenko, I.; Yu, Y.; Shang, W.; Huang, Y.; Abraham, J.P.; DiBenedetto, S.; et al. *NFD Developer's Guide*; Technical Report NDN-0021; Department of Computer Science, University of California: Los Angeles, CA, USA, 2014; Volume 29, p. 31.