smart cities

MDPI

Article

Enhancing Smart City Connectivity: A Multi-Metric CNN-LSTM Beamforming Based Approach to Optimize Dynamic Source Routing in 6G Networks for MANETs and VANETs

Vincenzo Inzillo [1,*,†], David Garompolo [1,†] and Carlo Giglio [2,*]

[1] Department of Computer Science, Istituto di Istruzione Superiore Tecnico Geometra, Industriale ed Economico, Via G. Fortunato, 89900 Vibo Valentia, VV, Italy; david.garompolo@tecnologicovibo.edu.it

[2] Department of Mechanical, Energy and Management Engineering, University of Calabria, Via P. Bucci, 87036 Arcavacata, CS, Italy

* Correspondence: vincenzo.inzillo@tecnologicovibo.edu.it (V.I.); carlo.giglio@unical.it (C.G.)

† These authors contributed equally to this work.

Highlights:

What are the main findings?

- MMS-DSR integrates advanced machine learning techniques and beamforming to optimize routing in 6G-enabled MANETs and VANETs.
- The protocol demonstrates significant improvements in throughput, latency, and routing overhead compared to traditional routing protocols.

What is the implication of the main finding?

- MMS-DSR's ability to adapt to real-time network conditions makes it highly suitable for dynamic urban environments.
- The reduction in routing overhead and enhanced scalability positions MMS-DSR as a robust solution for future smart city applications.

Abstract: The advent of Sixth Generation (6G) wireless technologies introduces challenges and opportunities for Mobile Ad Hoc Networks (MANETs) and Vehicular Ad Hoc Networks (VANETs), necessitating a reevaluation of traditional routing protocols. This paper introduces the Multi-Metric Scoring Dynamic Source Routing (MMS-DSR), a novel enhancement of the Dynamic Source Routing (DSR) protocol, designed to meet the demands of 6G-enabled MANETs and the dynamic environments of VANETs. MMS-DSR integrates advanced technologies and methodologies to enhance routing performance in dynamic scenarios. Key among these is the use of a CNN-LSTM-based beamforming algorithm, which optimizes beamforming vectors dynamically, exploiting spatial-temporal variations characteristic of 6G channels. This enables MMS-DSR to adapt beam directions in real time based on evolving network conditions, improving link reliability and throughput. Furthermore, MMS-DSR incorporates a multi-metric scoring mechanism that evaluates routes based on multiple QoS parameters, including latency, bandwidth, and reliability, enhanced by the capabilities of Massive MIMO and the IEEE 802.11ax standard. This ensures route selection is context-aware and adaptive to changing dynamics, making it effective in urban settings where vehicular and mobile nodes coexist. Additionally, the protocol uses machine learning techniques to predict future route performance, enabling proactive adjustments in routing decisions. The integration of dynamic beamforming and machine learning allows MMS-DSR to effectively handle the high mobility and variability of 6G networks, offering a robust solution for future wireless communications, particularly in smart cities.

Keywords: CNN-LSTM; MANET; VANET; beamforming; MU-MIMO; DSR; 802.11ax

check for updates

Citation: Inzillo, V.; Garompolo, D.; Giglio, C. Enhancing Smart City Connectivity: A Multi-Metric CNN-LSTM Beamforming Based Approach to Optimize Dynamic Source Routing in 6G Networks for MANETs and VANETs. *Smart Cities* **2024**, *7*, 3022–3054. https://doi.org/10.3390/smartcities7050118

Academic Editors: Pierluigi Siano and Gianluigi Ferrari

Received: 9 July 2024
Revised: 13 September 2024
Accepted: 11 October 2024
Published: 17 October 2024

1. Introduction

Mobile Ad Hoc Networks (MANETs) have evolved from a specialized area of study into a ubiquitous element of modern wireless communications, playing a crucial role in the development of smart cities. Initially engineered for tactical military communications, their adaptability and resilience have expanded their applications to include disaster recovery, remote sensing, and complex multi-layered Internet of Things (IoT) ecosystems—key components of urban infrastructure.

As the urban environment becomes increasingly connected, Vehicular Ad Hoc Networks (VANETs) emerge as a critical subset of MANETs, specially designed for the fast-moving nature of vehicular networks [1–3]. VANETs enable vehicles to communicate with each other and with roadside infrastructure, facilitating not only improved traffic management but also enhancing safety and supporting a range of services from navigation to automated driving in smart cities [4–6].

The impending arrival of Sixth Generation (6G) wireless technologies is poised to revolutionize the context of both MANETs and VANETs, particularly in urban settings. Unlike its predecessors, 6G represents a fundamental shift in wireless communication capabilities rather than a mere incremental upgrade. This upcoming technological leap is expected to dramatically reshape the extent and complexity of these network types within city environments. It promises ultra-low latency levels, down to the sub-millisecond range, unprecedented data rates reaching terabits per second, and near-perfect reliability [7,8]. These features are expected to support futuristic applications crucial for smart cities, such as real-time augmented and virtual reality, machine-to-machine communications, ultra-reliable low-latency services, and the tactile Internet, which have extreme demands on network performance.

Moreover, resource allocation, a challenge already significant in 5G, continues to be a key issue in 6G networks. Recent studies have leveraged Particle Swarm Optimization (PSO) to address non-convex optimization problems in resource allocation, demonstrating that PSO can outperform traditional algorithms like Genetic Algorithms in achieving convergence and cost minimization [9]. Advancements in MIMO technology, such as circularly polarized (CP) high-isolation MIMO antennas, have been demonstrated to improve S-parameters, mutual coupling, and channel capacity, making them well-suited for WLAN applications in the 5 GHz spectrum, which aligns with the goals of future VANETs and MANETs [10].

However, the transition to 6G introduces significant challenges, particularly for the inherently complex and dynamic networks like MANETs and VANETs in urban settings [11–13]. Traditional routing protocols, such as Dynamic Source Routing (DSR), are proving inadequate in this context. Designed in an era where network demands were less stringent, these protocols often rely on simplistic metrics like hop count for route selection, which is insufficient in the multi-dimensional Quality of Service (QoS) requirements of 6G, where latency, bandwidth, reliability, and energy efficiency are critical factors that must be simultaneously optimized [14–17].

To address these challenges, we introduce MMS-DSR (Multi-Metric Scoring Dynamic Source Routing), a novel modification and enhancement of the traditional DSR protocol, specifically designed for the complex requirements of 6G-enabled MANETs and optimized for the high-velocity, highly mobile environments of VANETs in smart cities. MMS-DSR incorporates several innovative features, including the utilization of Massive MIMO technologies and the IEEE 802.11ax standard [18]. Massive MIMO, with its capacity to manage multiple transmit and receive antennas, offers significant improvements in data rates and link reliability. The IEEE 802.11ax standard, conversely, provides advancements in network efficiency and capacity, particularly in environments with a high density of connected devices.

In addition to these advancements, we enhance MMS-DSR by integrating a CNN-LSTM based beamforming algorithm. This enhancement is designed to optimize the beamforming vectors dynamically, a critical aspect in 6G networks to adapt to rapid spatial-temporal variations in the channel. By taking advantage of the capabilities of Convolutional Neural Networks (CNNs) to extract spatial features and Long Short-Term Memory (LSTM) networks to account for temporal dependencies, our approach adapts the beamforming vectors in real-time based on the evolving network conditions.

This adaptive beamforming capability is particularly important in the context of 6G's use of Massive MIMO systems, where the ability to dynamically direct beams can significantly enhance signal quality and network performance. The integration of this beamforming algorithm into MMS-DSR means that the protocol is not only making intelligent decisions based on multi-metric scores but also considering the optimal beamforming directions to improve link reliability and data rates in the complex urban features of smart cities. The main contributions of the present proposal are synthesized as follows:

1. **Development of an Advanced Multi-Metric Scoring Mechanism:** we introduce a novel scoring algorithm that uses the properties of Massive MIMO and the 802.11ax standard. This algorithm evaluates routes based on a comprehensive set of QoS metrics, providing a nuanced and context-aware route selection process optimized for urban environments.
2. **Introduction of Machine Learning Techniques:** by incorporating machine learning algorithms, MMS-DSR is able to predict future route qualities based on historical data, enabling proactive route optimization. This is particularly useful in smart cities where predicting and adapting to changing conditions can significantly improve network performance.
3. **Integration of a CNN-LSTM Based Beamforming Algorithm:** we enhance MMS-DSR with a beamforming optimization algorithm that dynamically adjusts beamforming vectors. This integration helps in managing the spatial-temporal variations in the channel, supporting the complex beamforming needs of Massive MIMO systems in 6G.
4. **Empirical Validation in a Smart City Simulation Environment:** we validate the performance of MMS-DSR through extensive simulations using the *inet* framework along with the OSG Earth visualizer in OMNeT++. This simulation environment provides a realistic backdrop that highlights the protocol's potential benefits for smart city applications, demonstrating its robustness and efficiency in a variety of challenging urban 6G scenarios.

The remainder of this paper is structured as follows: Section 2 discusses the most relevant related works illustrating the conceptual gaps that our research aims to fill. Sections 3 and 4 present the MMS-DSR architectural overview and mathematical proof, respectively. Section 5 presents CNN-LSTM model architecture. Section 6 explains how the proposed approach could be adapted for VANET while Section 7 showcases and discusses the main findings obtained by numerical evaluation. Finally, Section 8 concludes the paper and outlines the roadmap for future research. Table 1 includes the list of main symbols used in the paper.

Table 1. List of main Symbols and Acronyms used in the paper.

Symbol	Description
α	Weighting factor for the summation real-time term
λ	Weighting factor for the beamforming score
δ	Weighting factor integrating the predictive reliability score
$w_i(t)$	Time-dependent weights
$m_i(r)$	Route metrics for MANET
$v_i(r)$	Route metrics for VANET
$P_{prediction}(r,t)$	Predictive reliability score
$BF(r,t)$	Beamforming score
$L(r,t)$	Latency
$B(r,t)$	Bandwidth

Table 1. *Cont.*

Symbol	Description
$R(r,t)$	Reliability
$B(r,t)$	Bandwidth
$BE(r,t)$	Beamforming efficacy
$De(r,t)$	Vehicle density
$Di(r,t)$	Vehicle distance
$T(r,t)$	Vehicle trajectory
$S(r,t)$	Scoring function
$TNVA$	Total Number of Vehicles in the Area
AS	Area Size
d_{i}	Euclidean distance of vehicle from the destination
d_{ref}	Reference distance
ζ	Attenuation factor
$d(t)^{\zeta}$	Future distance of the vehicle in the route to the destination at time t
$d(0)^{\zeta}$	Current distance to the destination
$J(w,t)$	Cost function
$\hat{m}_i(r,t)$	Predicted value of the metric
S	Real-time measured data matrix in input into the CNN
X	High-level spatial feature map output by the CNN
w_t	filter with $K-L+1$ coefficients
b_t	bias term
f_l	l-th activation function
$\circledast$	convolution operation
F	High-level spatial feature map in input to LSTM
H	Spatio-temporal features output by LSTM
M	Adjustable input window size
T	Adjustable ouput window size
$W_{*,F}, W_{*,H}$	Weighting matrices for current input high-level spatial feature matrix F_n and previous spatio-temporal feature matrix H_{n-1}
DSR	Dynamic Source Routing
MMS-DSR	Multi-Metric Scoring Dynamic Source Routing
SOL-DSR	Self-Organizing Learning Dynamic Source Routing
OLSR	Optimized Link State Routing
RSU	Road-Side Unit
MU-MIMO	Multi-User Multiple Input Multiple Output
OMNeT++	Objective Modular Network Testbed in C++
SUMO	Simulation of Urban MObility

2. Related Works

The rapidly evolving field of Mobile Ad Hoc Networks has witnessed the development of numerous routing protocols, each with its distinct set of characteristics and performance optimizations. However, few have addressed the challenges and opportunities presented by emerging 6G technologies as comprehensively as the MMS-DSR protocol introduced in this paper.

The study [19] enhances the DSR protocol using a hybrid optimization approach called MET-MFO, which combines Minimum Execution Time (MET) scheduling and Moth Flame Optimization (MFO). MET reduces latency by selecting routes that minimize packet processing time, while MFO ensures global route optimization, balancing exploration and exploitation to find stable paths that minimize energy consumption and extend network lifetime. Together, MET-MFO improves route discovery by reducing delays, optimizing energy use, and maintaining stable connections in dynamic MANET environments. Our MMS-DSR protocol extends this by integrating machine learning to predict and adapt to network changes, optimizing not just execution time but also key metrics like latency, bandwidth, and reliability, offering a more adaptive and comprehensive performance boost in VANET scenarios.

The article [20] presents a routing protocol that evaluates routes based on multiple metrics, including latency, bandwidth, and reliability. This multi-metric approach aims to provide a more comprehensive and adaptive routing decision-making process, ensuring higher quality of service (QoS) in diverse network conditions. The MMS-DSR protocol, however, builds on this foundation by dynamically adjusting the importance of each metric based on current network conditions using machine learning techniques, ensuring better adaptability and incorporating 6G technologies and advanced beamforming techniques, making it more suitable for the high-speed, high-density urban environments typical of smart cities.

In work [21], a new method called Select Optimal Link in Dynamic Source Routing (SOL-DSR) is proposed to specify the ideal path for routing messages between the source node and target node. In this method, the ideal path is selected according to three metrics: node energy level, number of nodes' neighbors and distance between any pair of the nodes that are used in the routing path. Our MMS-DSR protocol, however, advances this by considering a multi-dimensional metric space consisting of latency, bandwidth, and reliability, thereby providing a nuanced approach to route selection. This is particularly important in smart cities where diverse traffic conditions and varying user demands necessitate more sophisticated routing decisions.

When it comes to security considerations, ref. [22] has made significant strides by incorporating a trust-based mechanism to counter black hole attacks in DSR-based MANETs. While their work is seminal in securing DSR protocols, MMS-DSR has the architectural flexibility to integrate such security measures, thus potentially offering a more comprehensive solution that addresses both performance and security concerns, which are crucial in urban environments where network threats could disrupt critical city functions.

In the work [23] the author proposes a dynamic source routing protocol based on path reliability and monitoring repair mechanism (DSR-PM). The model performs data transmission by filtering the best reliability path. The link state information is monitored during transmission and broken links are repaired in time to ensure the communication stability and reliability of the links and improve the data transmission efficiency. While DSR-PM introduces important enhancements to traditional DSR by emphasizing path reliability and link repair mechanisms, our MMS-DSR protocol offers several significant improvements and advanced features that make it more suitable for dynamic urban environments and smart city applications.

In [24], authors adopted the Analytic Hierarchy Process (AHP) for multi-metric route evaluation, focusing on metrics like bandwidth and hop count. Although AHP is a robust and mathematically rigorous method for multi-criteria decision-making, it is static in nature. MMS-DSR, however, employs machine learning-based dynamic weight adjustment to evaluate routes, thus offering a more adaptable and real-time solution suitable for the dynamic and heterogeneous networks of smart cities.

In [25] the role of machine learning in MANET routing has been diversely implemented and notably employed Support Vector Machines (SVMs) for adaptive route selection based on network state classification. MMS-DSR takes this a step further by utilizing Long Short-Term Memory networks that offer the advantage of temporal sequence learning, thus enabling more reliable route prediction over time. This feature is particularly advantageous in urban settings where past network performance can inform future routing decisions to optimize traffic flow and resource allocation.

Laanaoui and Raghay [26] introduced an enhancement to the Optimized Link State Routing (OLSR) protocol by incorporating an Advanced Greedy Forwarding (AGF) mechanism specifically designed for Vehicular Ad Hoc Networks (VANETs) in smart cities. Their approach improved the classic OLSR protocol by introducing a scoring mechanism that considers the position and speed of vehicles to select the best forwarding path, which reduces end-to-end delay and improves the packet delivery ratio. This is especially crucial in smart cities where the dynamic and fast-changing vehicular environment requires routing protocols to quickly adapt to changes. Their work demonstrates how adapting routing protocols for VANETs can significantly enhance communication efficiency in urban environments, aligning with the goals of MMS-DSR to optimize MANET routing in the context of smart cities.

Finally, the anticipation of 6G technologies in the evolution of MANETs has been well-articulated by [27]. This study is more of a foresight into the infrastructural changes that 6G will bring to MANETs. MMS-DSR is designed keeping these technological advancements in mind, aiming to provide a routing protocol that is not just optimized for today's networks but is future-proof and adaptable for the next generation of mobile communication.

The work [28] attempts to optimize the bandwidth in the DSR routing protocol during data communication in MANET. The paper purposes a modified Dynamic Source routing protocol which is a Systematic Analysis Dynamic Source Routing protocol (SA-DSR). While SA-DSR introduces important enhancements to bandwidth optimization in traditional DSR it does not incorporate advanced communication technologies like beamforming or MU-MIMO which implies that this approach is not very suitable for high-mobility environments.

In light of the above, MMS-DSR presents a transformative approach that synergistically integrates machine learning, advanced optimization, and multi-metric evaluation. More importantly, it does so while natively considering the capabilities and potential of 6G technologies, thereby distinguishing itself as a pioneering solution in the field of MANET routing protocols optimized for smart cities.

Finally, in order to understand the research gap in the existing literature and highlight the contributions of the present proposal, Table 2 depicts a table with checkboxes locating the most relevant references analyzed in this work and our research direction. The table shown in Table 2 also categorizes each of the analyzed and discussed works based on their topic by evidencing that our work attempts to handle different topics simultaneously.

Table 2. Comparison of Approaches Across Key Dimensions.

Approach	Classic Enhanced-DSR	Multiple Metric	Machine Learning	Security Handling	Specifically Designed for VANET
Work [19]	✓				
Work [20]		✓			
Work [21]	✓				
Work [22]				✓	
Work [23]	✓				
Work [24]		✓			
Work [25]			✓		
Work [26]					✓
Work [27]			✓		
Work [28]	✓				
Proposed	✓	✓	✓		✓

3. Architectural Overview

MMS-DSR's architecture is designed to optimize routing decisions dynamically in 6G MANETs by integrating advanced machine learning models and beamforming techniques, specifically tailored for the dynamic and complex environments of smart cities. This protocol builds upon a modular structure that enhances traditional DSR components with modern technological advancements to improve adaptability and performance. Below is a detailed breakdown of each component within this architecture:

- Route Discovery Unit (RDU):initiates the discovery process when a route to a destination is required. Utilizing a combination of traditional flooding methods and predictive models, it efficiently identifies potential routes. In smart cities, this process is further enhanced by employing machine learning to predict and avoid congested or unreliable paths. This proactive route discovery is essential for maintaining high communication efficiency in urban settings.
- Multi-metric Scoring Engine (MMSE): evaluates discovered routes based on multiple metrics such as latency, bandwidth, reliability, and beamforming efficacy. This engine employs dynamic weighting to adjust the importance of each metric based on current network conditions, which is crucial for adapting to the varying demands of urban environments.
  - Beamforming Efficacy (BE): calculates the improvement in signal quality and directionality due to beamforming optimizations.
  - Weighted Scoring: each metric is assigned a dynamic weight. These weights are adjusted in real time using feedback from the Machine Learning-Based Prediction Unit to ensure that scoring aligns with the predicted network state.
  - Optimization: the engine utilizes a combination of heuristic and machine-learning techniques to fine-tune the weights based on ongoing network performance data. This adaptive approach ensures that the engine remains responsive to changing network dynamics, particularly in adjusting beamforming strategies for optimal route performance in smart cities.
- Machine Learning-Based Prediction Unit (MLPU): utilizes CNN-LSTM models to predict future network states, including changes in channel quality and node mobility. This unit enhances route selection by forecasting future performance and optimizing beamforming directions to maintain high-quality communication links. This is particularly valuable in smart cities where predictive capabilities can lead to more proactive and efficient network management.
  - Predictive Modeling: the unit employs CNN-LSTM models to analyze historical and current network data to forecast future conditions such as node mobility, channel quality, and potential interference sources.
  - Integration with Beamforming: the predictions include recommended adjustments to beamforming vectors. By predicting how channel conditions will evolve, the unit guides the Beamforming Optimization Unit to adjust angles and power levels proactively.
  - Feedback Loop: predictive insights are fed back into both the Route Discovery and Multi-metric Scoring Engine, allowing these modules to prioritize routes that are expected to offer optimal performance in the near future.
- Beamforming Optimization Unit (BOU) : a critical enhancement in MMS-DSR, this unit dynamically adjusts beamforming vectors utilizing the CNN-LSTM model based on real-time and historical channel state information (CSI). This approach ensures optimal signal directionality and strength, significantly enhancing link reliability and throughput, which is essential for maintaining robust communication in the high-density urban environments of smart cities.
  - Dynamic Beamforming Vector Adjustment: this unit employs the output from the Machine Learning-Based Prediction Unit to adjust beamforming vectors. It optimizes these vectors to maximize signal strength and minimize interference, taking into account both the current and predicted CSI.
  - Feedback to Scoring Engine: adjustments made by this unit are fed back into the Multi-metric Scoring Engine, enabling it to re-evaluate route scores with updated beamforming information.
- Route Cache (RC): stores the most efficient routes as determined by the scoring engine. It is periodically updated based on predictive feedback from the Machine Learning-Based Prediction Unit, which now also incorporates beamforming vector adjustments.

This adaptive caching is crucial in smart cities for reducing routing overhead and improving responsiveness.

The architectural diagram of MMS-DSR presented in Figure 1 provides a comprehensive visual representation of the various components and their interactions. The diagram is designed to reflect the complexity and integrative nature of the MMS-DSR protocol. The arrows in the diagram represent the flow of information between these components, demonstrating the modular and interconnected nature of the MMS-DSR architecture. The Route Discovery unit initiates the process by identifying potential routes. It utilizes a combination of traditional flooding methods and machine learning-based predictive models, ensuring that the discovered routes adapt dynamically to urban traffic conditions, reflecting both extensive and efficient pathfinding.

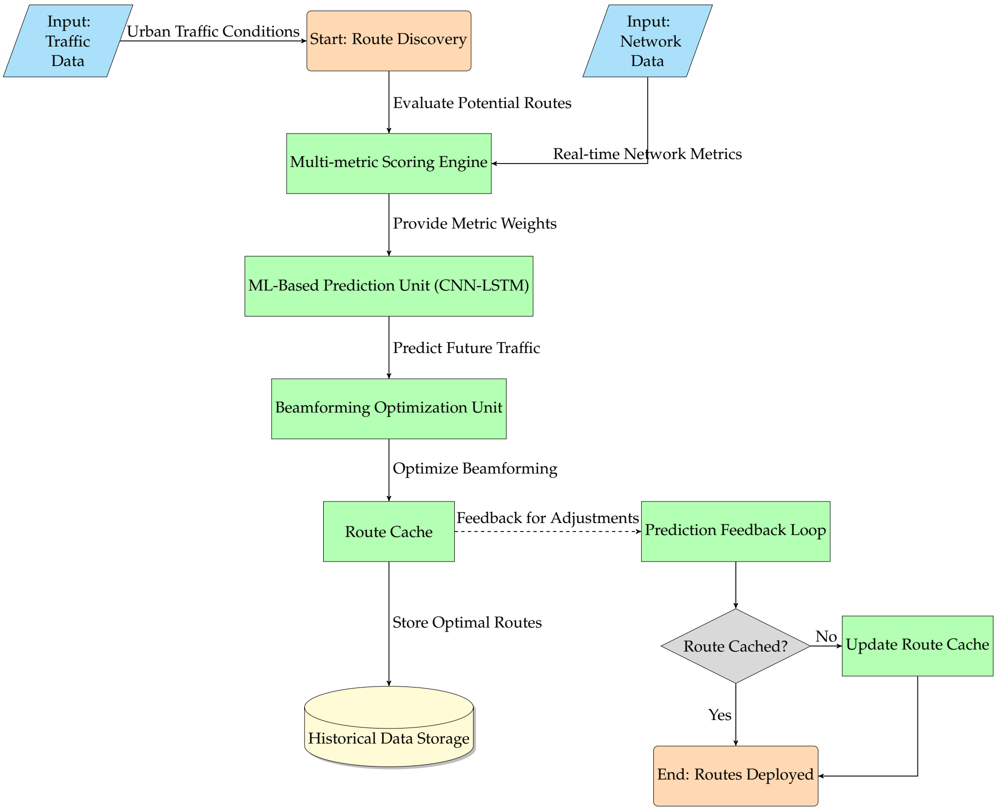

Figure 1. Flowchart for MMS-DSR Architecture.

Following route discovery, the Multi-metric Scoring Engine evaluates each identified route based on multiple metrics, such as latency, bandwidth, reliability, and beamforming efficacy. This engine adjusts the weight of each metric dynamically according to current network conditions, ensuring that the selected routes are optimized to provide balanced and efficient communication. The Machine Learning-Based Prediction Unit exploits advanced CNN-LSTM models to predict future network states, such as changes in traffic flow, node mobility, and channel quality. These predictions are crucial for proactive adjustments to routing and beamforming. The prediction unit communicates with the Multi-metric Scoring Engine and Beamforming Optimization Unit to ensure the system adapts to future conditions rather than just reacting to the current network state.

The Beamforming Optimization Unit, plays a critical role in maintaining high-quality communication links, especially in high-density urban environments. By dynamically adjusting beamforming vectors based on both real-time and historical channel state information (CSI), this unit ensures optimal signal directionality and strength, thereby enhancing link reliability and throughput. The Route Cache stores the most efficient routes as determined by the Multi-metric Scoring Engine. The Route Cache periodically updates itself based on predictive feedback from the CNN-LSTM model, ensuring it remains adaptable to network changes. Additionally, it incorporates beamforming vector adjustments to further enhance routing efficiency and reduce overhead.

Traffic Data and Network Data serve as inputs to the Route Discovery and Multi-metric Scoring Engine, respectively. These inputs provide real-time information about urban traffic conditions and network performance metrics, which are essential for determining the optimal routes. The Historical Data stored in the Route Cache and used by other units provide long-term context to enhance future routing decisions. The CNN-LSTM Model plays a central role in predicting future traffic and network conditions. Arrows indicate its influence on both the Multi-metric Scoring Engine and Beamforming Optimization Unit, where its predictive insights are applied to fine-tune routing and communication strategies.

3.1. Multi-Metric Scoring Engine in MMS-DSR

The Multi-metric Scoring Engine is the core component that evaluates and ranks the discovered routes based on multiple performance metrics. The engine operates according to the following features:

- Dynamic Metric Evaluation: the engine processes each route to compute scores based on five key metrics: latency, bandwidth, reliability, beamforming efficacy, and Channel State Information.
- Weighted Scoring: each metric is assigned a dynamic weight reflecting its current importance based on network conditions. These weights are adjusted in real time using feedback from the Machine Learning-Based Prediction Unit to ensure that scoring aligns with the predicted network state. This approach allows the system to adapt to the urban dynamics effectively, prioritizing different metrics as the urban environment evolves.
- Score Computation: the overall score for each route, S, is computed as follows:

$$S(route) = w_1 \cdot \frac{1}{L(route)} + w_2 \cdot B(route) + w_3 \cdot R(route) + w_4 \cdot BE(route) \qquad (1)$$

  where w_1, w_2, w_3, and w_4 are the weights for latency, bandwidth, reliability, and beamforming efficacy, respectively. This formula ensures that routes are evaluated comprehensively, incorporating both traditional and advanced metrics to select the most efficient path.
- Optimization: the engine uses a combination of heuristic and machine-learning techniques to fine-tune the weights based on ongoing network performance data. This adaptive approach ensures that the engine remains responsive to changing network dynamics, particularly in adjusting beamforming strategies for optimal route performance in smart cities.

This scoring process enables MMS-DSR to select the most efficient and reliable routes, considering both current network metrics and future state predictions. The inclusion of beamforming efficacy as a metric ensures that the protocol can adaptively manage and utilize advanced beamforming techniques in 6G networks for enhanced communication quality.

3.2. Machine Learning-Based Prediction Unit in MMS-DSR

The Machine Learning-Based Prediction Unit in MMS-DSR utilizes advanced CNN-LSTM models to predict future network states, enabling proactive adjustments to routing and beamforming parameters. This predictive capability aims to maintain high performance in the dynamic environments typical of 6G networks and smart cities.

- Predictive Modeling: the unit employs CNN-LSTM models to analyze historical and current network data to forecast future conditions such as node mobility, channel quality, and potential interference sources. This analysis helps predict the stability and performance of each link in a route, which is crucial for smart cities where conditions change rapidly.

- Integration with Beamforming: the predictions include recommended adjustments to beamforming vectors to optimize communication links for future network states. By predicting how channel conditions will evolve, the unit guides the Beamforming Optimization Unit to adjust angles and power levels proactively, ensuring optimal performance in the varied urban topography and infrastructure.
- Feedback Loop: predictive insights are fed back into both the Route Discovery and Multi-metric Scoring Engine, allowing these modules to prioritize routes that are expected to offer the best performance in the near future. This feedback loop ensures that routing decisions are made with an eye towards future conditions, not just current metrics, enhancing the adaptability and foresight of the routing protocol in smart cities.
- Model Training and Updating: the CNN-LSTM models are continuously trained and updated with new data to improve their accuracy. This ongoing training process allows the models to adapt to changes in network behavior and topology, ensuring that the predictions remain relevant and accurate over time. This continual learning is essential in smart cities, where urban conditions and patterns can evolve unpredictably.

3.3. Beamforming Optimization Unit in MMS-DSR

The Beamforming Optimization Unit is a key enhancement in MMS-DSR, designed to dynamically optimize beamforming vectors using CNN-LSTM models based on the predicted network conditions. This unit plays a critical role in adjusting the beamforming strategies to enhance signal quality and reliability over time, particularly in urban scenarios where buildings and other structures can cause significant signal reflection and diffraction. The Algorithm 1 explains main working operations performed by Beamforming Optimizarion unit. Main steps are synthesized as follows:

- Dynamic Beamforming Vector Adjustment: this unit uses the output from the Machine Learning-Based Prediction Unit to adjust beamforming vectors. It optimizes these vectors to maximize signal strength and minimize interference, considering both the current and predicted CSI. This real-time adjustment is crucial for urban environments where obstacles may deflect or block signals unexpectedly.
- CNN-LSTM Based Predictions: the unit employs Convolutional Neural Networks (CNNs) combined with Long Short-Term Memory networks to analyze spatial and temporal aspects of the network. This model predicts optimal beamforming directions and power levels for each node in the network.
- Real-Time Optimization: beamforming vectors are adjusted in real-time based on predictive analytics. This ensures that each node can proactively adapt to changing network conditions, maintaining high-quality communication links, which is essential in smart cities for supporting uninterrupted service delivery.
- Feedback to Scoring Engine: adjustments made by this unit are fed back into the Multi-metric Scoring Engine, allowing it to re-evaluate route scores with updated beamforming information. This feedback loop ensures that routing decisions remain optimal as network conditions evolve, particularly under the variable urban dynamics.
- Integration with Routing: information about the optimized beamforming vectors is included in the route discovery and maintenance processes, ensuring that all nodes along a chosen route adjust their beamforming strategies cohesively for uniform signal enhancement across the urban network.

By dynamically optimizing beamforming vectors, this unit significantly contributes to the robustness and efficiency of MMS-DSR, particularly in environments where directional communication can greatly enhance performance, such as in crowded urban areas.

Algorithm 1 Beamforming Vector Optimization Algorithm

```
procedure OptimizeBeamformingVectors(Routes, CNNLSTMModel)
    for each Route in Routes do
        for each Link in Route do
            LinkCSI ← GetCurrentCSI(Link)
            PredictedCSI ← CNNLSTMModel.Predict(LinkCSI)
            OptimalVectors ← CalculateOptimalBeamformingVectors(PredictedCSI)
            AdjustBeamformingVectors(Link, OptimalVectors)
        end for
    end for
end procedure
```

3.4. Route Cache in MMS-DSR

The Route Cache in MMS-DSR has been enhanced to dynamically store and manage the most efficient routes based on real-time and predictive analytics. This module now works closely with the Beamforming Optimization Unit to ensure that the cached routes are not only optimal in terms of path metrics but also in terms of communication quality and reliability. The operation principle of Route Cache is briefed by Algorithm 2 and consists of the following phases:

- Dynamic Caching: the Route Cache dynamically stores routes that are scored highest by the Multi-metric Scoring Engine. It takes into account not only the traditional route metrics but also the beamforming efficacy, ensuring that the stored routes are optimal under current and predicted network conditions. This dynamic caching is key for reducing routing overhead and improving responsiveness in the fluctuating urban environment.
- Predictive Updates: based on feedback from the Machine Learning-Based Prediction Unit, the Route Cache updates its entries to preempt potential degradations in route quality. This includes adjusting stored routes based on predicted changes in node mobility and channel quality, ensuring that the cache reflects the most current and anticipated network states, and enhancing the system's ability to handle urban dynamics.
- Beamforming Information: each route in the cache includes detailed beamforming vector information for every link in the path. This ensures that when a route is retrieved from the cache, each node along the path can quickly adjust its beamforming vectors to the optimal settings, facilitating a coherent and coordinated approach to maintaining route quality.
- Eviction and Maintenance: the cache follows an intelligent eviction policy where less optimal routes are replaced by newer, higher-quality routes. This policy considers route age, frequency of use, and predictive quality scores to maintain a balance between route freshness and historical efficacy, which is crucial for ensuring that the network can quickly adapt to changes in the urban context.
- Support for Fast Route Recovery: in case of rapid topology changes, which are common in urban environments, the Route Cache supports fast route recovery by providing alternative paths that can be quickly evaluated and deployed, minimizing downtime and packet loss.

The enhancements to the Route Cache enable MMS-DSR to maintain a high-performance routing table that is adaptive and predictive, significantly reducing the need for frequent route discoveries and improving overall network efficiency in the complex and dynamic environments of smart cities.

Algorithm 2 Dynamic Route Caching and Maintenance Algorithm

```
procedure UpdateRouteCache(NewRoute, PredictedQuality)
    if RouteCache.IsFull() then
        EvictLeastOptimalRoute(RouteCache)
    end if
    AdjustBeamformingVectors(NewRoute)
    RouteCache.Store(NewRoute, PredictedQuality)
end procedure
```

4. MMS-DSR Mathematical Formalization

MMS-DSR is designed with a unique multi-metric scoring function that adapts in real-time to the dynamic nature of Mobile Ad-Hoc Networks, incorporating beamforming strategies to enhance signal quality and reliability. This scoring function serves as the backbone for making robust and efficient routing decisions, especially in the complex and rapidly changing environments of smart cities. The scoring function $S(r,t)$ in MMS-DSR is defined as follows:

$$S(r,t) = \alpha \cdot \sum_{i=1}^{n} w_i(t) \cdot m_i(r) + \delta \cdot P_{\text{prediction}}(r,t) + \lambda \cdot BF(r,t) \tag{2}$$

In the Equation (2), $m_i(r)$ represents the metrics for each route, while $w_i(t)$ denotes weights that vary with time. The term α serves as a coefficient for the real-time summation component and δ acts as a factor that incorporates the predictive reliability score, denoted as $P_{prediction}(r,t)$. Additionally, λ functions as a weighting factor for the beamforming score expressed as $BF(r,t)$. The weights $w_i(t)$ are dynamically adjusted based on real-time network analytics and predictions from the Machine Learning-Based Prediction Unit to reflect the changing urban conditions. This adjustment is given by the following:

$$w_i(t+1) = w_i(t) - \eta \frac{\partial J(w,t)}{\partial w_i(t)} \tag{3}$$

where η represents the learning rate and $\frac{\partial J(w,t)}{\partial w_i(t)}$ is the gradient of the cost function with respect to weight $w_i(t)$. The cost function $J(w,t)$ is designed to minimize the difference between the current metric values and their predicted values, ensuring that the weights reflect the most relevant network conditions. The cost function is defined as follows:

$$J(w,t) = \sum_{i=1}^{n} \left(\frac{m_i(r,t) - \hat{m}_i(r,t)}{\hat{m}_i(r,t)} \right)^2 \tag{4}$$

The term $m_i(r,t)$ denotes the network metric for route r at time t and $\hat{m}_i(r,t)$ is the predicted value of the metric provided by the Machine Learning-Based Prediction Unit.

In order to understand how the parameters $P_{prediction}(r,t)$ and $BF(r,t)$ are obtained it is necessary to introduce Hybrid CNN-LSTM models. Hybrid Deep Learning models combining CNN and LSTM can improve the prediction accuracy [29–32]. The spatial and temporal features can be thoroughly extracted using hybrid models, where CNN is utilized to capture the spatial features of traffic data, while LSTM is employed to extract the temporal features. Let us consider traffic state data of K locations s_i $(i = 1, 2, \ldots, K)$ as inputs to predict the traffic states at times $t, t+1, \ldots, t+h$. The real-time measured data can be arranged, as explained in [32–34], into a matrix:

$$S = \begin{bmatrix} s_1 \\ s_2 \\ \vdots \\ s_K \end{bmatrix} = \begin{bmatrix} s_{1,t-N} & s_{1,t-N+1} & \cdots & s_{1,t-1} \\ s_{2,t-N} & s_{2,t-N+1} & \cdots & s_{2,t-1} \\ \vdots & \vdots & \ddots & \vdots \\ s_{K,t-N} & s_{K,t-N+1} & \cdots & s_{K,t-1} \end{bmatrix} \tag{5}$$

In $P_{\text{prediction}}(r, t)$, the matrix element $s_{k,t}$ represents a vector that includes latency (L), bandwidth (B), reliability (R), and beamforming efficacy (BE). In $BF(r, t)$, the element $s_{k,t}$ contains the Channel State Information (CSI). The real-time data matrix S is processed by a CNN, which extracts high-level spatial features. These features are then passed into LSTM models to produce predictions.

The high-level spatial feature map produced by the CNN can be expressed as follows:

$$X = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_L \end{bmatrix} = \begin{bmatrix} x_{1,t-N} & x_{1,t-N+1} & \cdots & x_{1,t-1} \\ x_{2,t-N} & x_{2,t-N+1} & \cdots & x_{2,t-1} \\ \vdots & \vdots & \ddots & \vdots \\ x_{L,t-N} & x_{L,t-N+1} & \cdots & x_{L,t-1} \end{bmatrix}$$

Each high-level feature $x_{l,t}$ is computed using the following equation:

$$x_{l,t} = f_l(w_t \otimes S_t + b_t)$$

The term w_t denotes the filter with $K - L + 1$ coefficients, while S_t represents the t-th column of the matrix S, and b_t is the bias term. The activation function f_l applies a non-linear transformation, and $\otimes$ indicates the convolution operation.

After extracting high-level spatial features, these vectors are selected as input for the LSTM network. The input to the LSTM is expressed as follows:

$$F = [F_0 \; F_1 \; \cdots \; F_{N-1}]$$

The high-level spatial feature map for each LSTM network n, denoted F_n, is formulated as follows:

$$F_n = \begin{bmatrix} x_{1,t+n-N} & x_{1,t+n-N+1} & \cdots & x_{1,t+n-N+M-1} \\ x_{2,t+n-N} & x_{2,t+n-N+1} & \cdots & x_{2,t+n-N+M-1} \\ \vdots & \vdots & \ddots & \vdots \\ x_{L,t+n-N} & x_{L,t+n-N+1} & \cdots & x_{L,t+n-N+M-1} \end{bmatrix}$$

In this formulation, M indicates the adjustable input window size. The spatio-temporal features generated by the LSTM are denoted as follows:

$$H = [H_0 \; H_1 \; \cdots \; H_{N-1}]$$

Each H_n is a matrix of size $K \times T$, with T representing the adjustable output window size, subject to the condition $M + T \leq N$.

The iterative process for generating spatio-temporal features is defined by the following equations:

$$\begin{aligned} f_n &= \sigma\Big(W_{f,F} \cdot \text{vec}(F_n) + W_{f,H} \cdot \text{vec}(H_{n-1}) + b_f\Big) \\ i_n &= \sigma(W_{i,F} \cdot \text{vec}(F_n) + W_{i,H} \cdot \text{vec}(H_{n-1}) + b_i) \\ c_n &= f_n \odot c_{n-1} + i_n \odot \tanh(W_{c,F} \cdot \text{vec}(F_n) + W_{c,H} \cdot \text{vec}(H_{n-1}) + b_c) \\ o_n &= \sigma(W_{o,F} \cdot \text{vec}(F_n) + W_{o,H} \cdot \text{vec}(H_{n-1}) + b_o) \\ H_n &= o_n \odot \tanh(c_n) \end{aligned}$$

where $W_{*,F}$ and $W_{*,H}$ are the weighting matrices associated with the current input high-level spatial feature matrix F_n and the previous spatio-temporal feature matrix H_{n-1}, respectively. The vectorization operation, denoted $\text{vec}(\cdot)$, accounts for the differing sizes of F_n and H_{n-1}. The functions σ and tanh represent the sigmoid and hyperbolic tangent functions, which are applied element-wise.

Numerical Illustration

To provide a concrete understanding of the MMS-DSR scoring mechanism, we present a numerical example with network topology, as illustrated in Figure 2. This hypothetical scenario considers several candidate routes, each with different metrics for latency, bandwidth, reliability, and beamforming efficacy (BE), typical of diverse urban paths.

The routes and their respective metrics are summarized in Table 3.

Table 3. Example routes with associated metrics.

Route	Latency (ms)	Bandwidth (Mbps)	Reliability	Beamforming Efficacy (BE)	$P_{prediction}$(r,t)	BF(r,t)
r_1	10	50	0.9	15	0.92	20
r_2	8	40	0.95	18	0.87	22.5
r_3	12	60	0.85	12	0.90	15
r_4	15	45	0.92	17	0.91	21
r_5	9	55	0.88	14	0.89	19
r_6	10	50	0.9	15	0.92	20
r_7	8	40	0.95	18	0.87	22.5
r_8	12	60	0.85	12	0.90	20
r_9	15	45	0.92	17	0.91	22.5
r_{10}	9	55	0.88	14	0.90	15

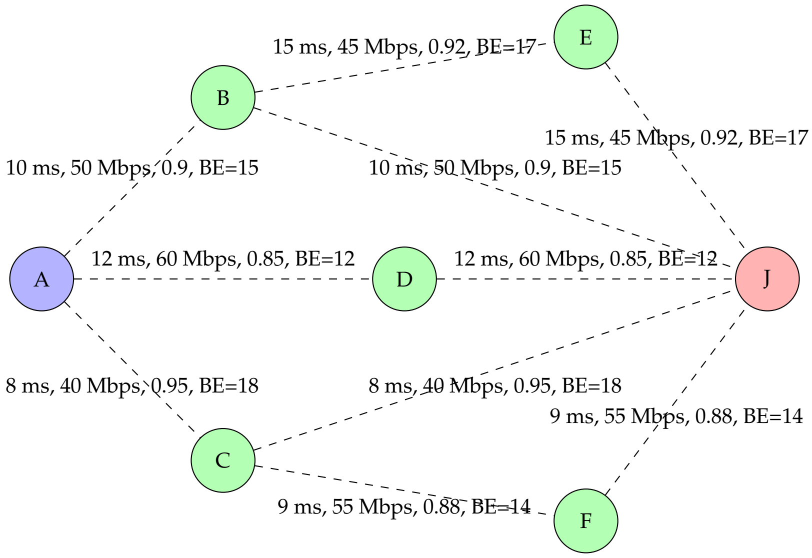
15 ms, 45 Mbps, 0.92, BE=17
E
B
15 ms, 45 Mbps, 0.92, BE=17
10 ms, 50 Mbps, 0.9, BE=15
10 ms, 50 Mbps, 0.9, BE=15
A
12 ms, 60 Mbps, 0.85, BE=12
D
12 ms, 60 Mbps, 0.85, BE=12
J
8 ms, 40 Mbps, 0.95, BE=18
8 ms, 40 Mbps, 0.95, BE=18
9 ms, 55 Mbps, 0.88, BE=14
C
9 ms, 55 Mbps, 0.88, BE=14
F

Figure 2. Network topology diagram illustrating the routes from Node A to Node J with respective metrics.

Using the scoring function we assume the following parameters: $\delta = 0.2$, $\lambda = 0.3$, $\beta = 0$, Experience $E = 10$, dynamic weights at time t: $w_1(t) = 0.35$, $w_2(t) = 0.4$, $w_3(t) = 0.2$, $w_4(t) = 0.05$.

To ensure a fair comparison, the metrics are normalized to a [0, 1] scale using the formula:

$$x_{\text{norm}} = \frac{x - x_{\text{min}}}{x_{\text{max}} - x_{\text{min}}} \tag{6}$$

The routes and their respective normalized metrics are summarized in Table 4.

The scoring function $S(r,t)$ for each route is calculated using the normalized values. For example for $r1$ we have the following:

$$S(r1,t) = 0.6125$$

Table 4. Normalized metrics for each route.

Route	Latency Norm	Bandwidth Norm	Reliability Norm	BE Norm	$P_{prediction}$ Norm(r,t)	BF Norm(r,t)
r1	0.286	0.50	0.5	0.500	1.0	0.667
r2	0.000	0.00	1.0	1.000	0.0	1.000
r3	0.571	1.00	0.0	0.000	0.6	0.000
r4	1.000	0.25	0.7	0.833	0.8	0.800
r5	0.143	0.75	0.3	0.333	0.4	0.533
r6	0.286	0.50	0.5	0.500	1.0	0.667
r7	0.000	0.00	1.0	1.000	0.0	1.000
r8	0.571	1.00	0.0	0.000	0.6	0.667
r9	1.000	0.25	0.7	0.833	0.8	1.000
r10	0.143	0.75	0.3	0.333	0.6	0.000

Repeating the same calculations for each route we will obtain the values showed in Table 5:

Table 5. Scoring Function values for each route after normalization.

Route	Scoring Function (SF)
r1	0.612500
r2	0.425000
r3	0.420000
r4	0.715833
r5	0.453333
r6	0.612500
r7	0.425000
r8	0.620000
r9	0.775833
r10	0.333333

Even if by applying the scoring function, the path *r*9 emerges as the optimal route with the highest score of 0.775833, we must remember that in the classic DSR protocol, the primary metric for selecting the optimal path is typically the number of hops. Carrying out the route analysis of the graph illustrated in Figure 3, the possible paths from source to destination and the related Scoring Function (SF) values are shown in Table 6:

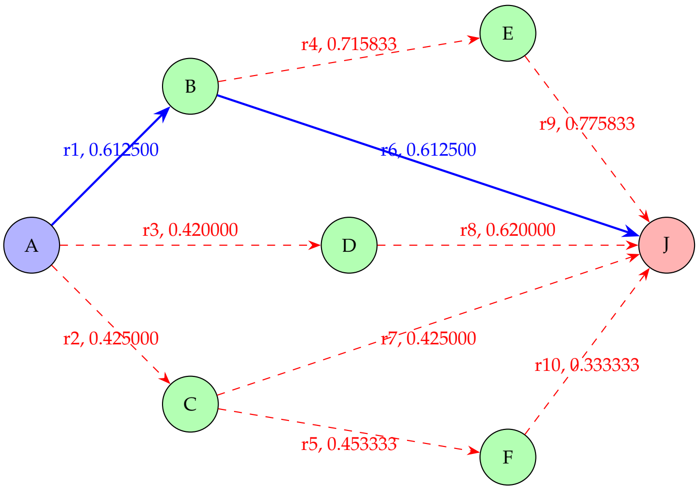

Figure 3. Network topology diagram illustrating the routes from Node A to Node J with respective metrics.

Table 6. Paths, Routes, and Total Scoring Functions (SF).

Path	Routes	Total SF
A → B → J	r1 + r6	$0.612500 + 0.612500 = 1.225$
A → C → J	r2 + r7	$0.425000 + 0.425000 = 0.850$
A → D → J	r3 + r8	$0.420000 + 0.620000 = 1.040$
A → B → E → J	r1 + r4 + r9	$0.612500 + 0.715833 + 0.775833 = 2.104166$
A → C → F → J	r2 + r5 + r10	$0.425000 + 0.453333 + 0.333333 = 1.211666$

Considering that MMS-DSR prioritizes the route with the highest cumulative scoring function (SF) and minimum number of hops, the chosen optimal path would be the following:

$A- > B- > J$ with a total SF of 1.225.

5. CNN-LSTM Model Architecture

The CNN-LSTM model architecture exploited by the MMS-DSR approach can be synthesized by the following points:

1. **Input Data Preparation:** the input to the CNN-LSTM model consists of various network metrics that are critical for routing and beamforming decisions. These basic metrics include the following:
   - **Latency (L):** measures the communication delay in the network.
   - **Bandwidth (B):** reflects the available communication capacity in the network.
   - **Reliability (R):** indicates the stability of communication links.
   - **Beamforming Efficacy (BE):** measures how effective beamforming is in improving signal quality.
   - **Channel State Information (CSI):** provides real-time information about the quality of the communication channels.

   These metrics represent both current and past network states, forming the input feature set that feeds into the CNN-LSTM model for further processing.
2. **Spatial Feature Extraction with CNN:** the CNN layers first receive normalized input data and are responsible for extracting spatial features. The input metrics, such as $\hat{L}$, $\hat{B}$, $\hat{R}$, $\hat{BE}$, and $\hat{CSI}$, are processed using convolutional filters. These filters generate feature maps that capture critical spatial patterns such as traffic congestion zones or signal interference areas, which are crucial for effective route planning in densely populated urban areas.
3. **Temporal Pattern Analysis with LSTM:** after the CNN extracts spatial features, the feature maps are fed into LSTM layers. The LSTM processes these features over time, capturing temporal dependencies in the network. For instance, the LSTM can identify recurring traffic patterns or predict future changes in network topology, which is essential for anticipating and adapting to network fluctuations.
4. **Integration of CNN and LSTM Outputs:** the CNN-derived spatial insights and the LSTM-derived temporal predictions are integrated at this stage. This combined output provides robust predictive insights, allowing for real-time, proactive adjustments to routing and beamforming strategies. The model can forecast which routes are likely to become congested and how beamforming can be adjusted to maintain signal strength.
5. **Route Optimization Outputs:** utilizing the integrated insights, the model outputs recommendations for optimizing routes and beamforming. The predictions guide the MMS-DSR protocol to adjust routes and communication strategies dynamically, particularly in scenarios with high mobility or unpredictable network conditions. The output also predicts the optimal beamforming vectors for maintaining strong communication links in complex urban environments.
6. **Model Training and Real-time Deployment:** the CNN-LSTM model is initially trained offline using a diverse dataset of simulated network scenarios, representing

varied urban traffic conditions and network configurations. The training dataset captures long-term network behaviors, enabling the model to learn both spatial and temporal patterns. Once trained, the model is integrated into the MMS-DSR protocol for real-time operation, continuously refining routing decisions based on real-time network data and its predictions.

The CNN-LSTM model is tailored for MMS-DSR in Figure 4. The architecture is designed to predict and optimize both routing and beamforming decisions, ensuring robust communication in urban environments with dynamic traffic patterns and network topologies.

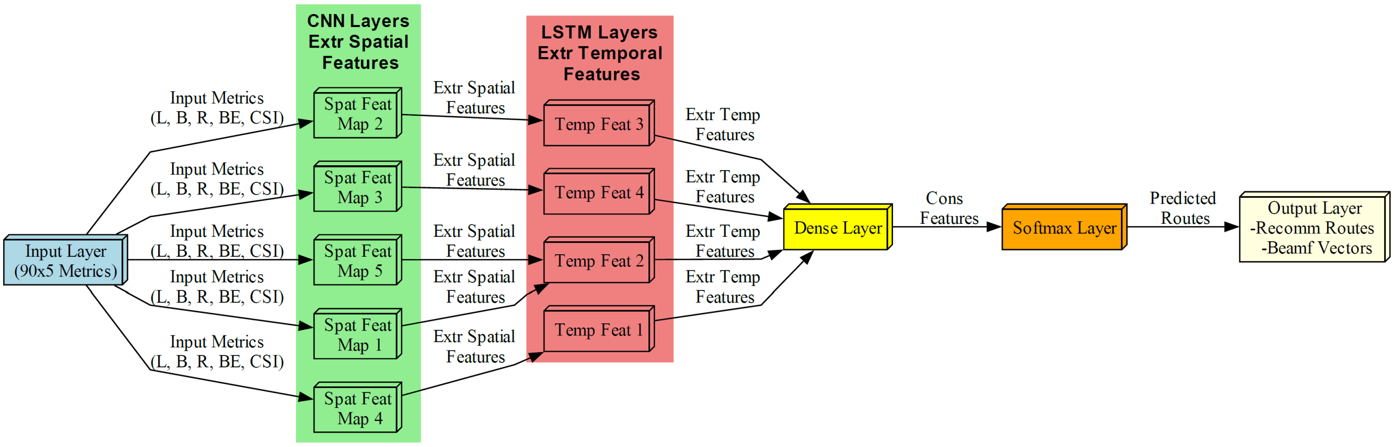

Figure 4. CNN-LSTM model architecture for MMS-DSR.

The model begins with an input layer that handles raw data with dimensions of 90 × 5, reflecting the collected metrics. This structure allows the model to process the network metrics across time, ensuring that spatial and temporal features are captured effectively.

The CNN layers apply convolutional filters, producing feature maps (90 × 32) that detect critical spatial patterns in the network, such as congestion areas or interference zones. The spatially processed data are then passed to the LSTM layers, which capture long-term temporal dependencies. These layers output 32 features each, providing a deep temporal analysis that is essential for predicting future network states.

The dense layer consolidates the spatio-temporal features learned from the CNN and LSTM layers into a final representation that summarizes the model's prediction about future network conditions. These outputs represent the decision variables or routing recommendations based on the spatial and temporal patterns captured in the previous layers. These decision variables could, for instance, predict the most efficient route, optimal beamforming vectors, or other routing-related metrics.

Following the dense layer, the model applies a softmax function to output probabilities for different routing actions or strategies. This allows the MMS-DSR protocol to make probabilistic decisions about route stability, optimal paths, and necessary beamforming adjustments. The output directly influences MMS-DSR's decision-making process, enabling it to dynamically adjust routes and beamforming strategies in real-time, based on predicted network conditions.

By predicting future network states, such as route stability and optimal beamforming vectors, the CNN-LSTM model significantly enhances the MMS-DSR protocol's ability to maintain robust communication links, especially in urban environments where network dynamics can change rapidly.

The CNN-LSTM model used in the MMS-DSR protocol is trained using historical traffic data collected from urban environments. These data include a variety of urban traffic scenarios characterized by different vehicle speeds, directions, network loads, and communication metrics. The dataset consists of traffic data collected over a span of six months, covering different traffic patterns, peak and off-peak hours and congestion scenarios.

The model training is conducted using MATLAB, which provides a robust environment for machine learning model development. The data are pre-processed to normalize features and handle missing values, ensuring consistency and accuracy in the training process. The training dataset consists of approximately 500,000 samples, and the data are split into 70% training, 15% validation, and 15% test sets.

The training process is performed on a centralized server with global knowledge of the historical traffic data. The training is performed offline, using the entire dataset to capture long-term trends and relationships. The model is then periodically updated with new data as the traffic patterns evolve. To further improve the understanding of the proposed architecture, main fundamentals theoretical concepts about CNN-LSTM are proofed in the Appendix A.

6. Adapting MMS-DSR from MANETs to VANETs for Smart Cities

To adapt the mathematical model discussed in Section 4 from MANETs to VANETs, it is necessary to consider the specific characteristics and requirements of VANETs. In particular, in mathematical formalization for VANETs we must incorporate additional parameters such as the Density of the Vehicles, the Distance of the Vehicle from the Destination, and the Trajectory of the Vehicle [35,36]. These parameters are crucial for ensuring efficient and reliable routing in highly dynamic vehicular environments.

Vehicle Density (De): vehicle density measures the concentration of vehicles within a specific area of the network. It is a critical metric for evaluating network congestion, as high vehicle density can lead to increased communication delays and packet collisions.

$$De(r,t) = \frac{\text{TNVA}}{\text{AS}} \tag{7}$$

$TNVA$ is the total number of vehicles in the area while AS is the area size. High vehicle density indicates potential congestion, which can negatively impact communication reliability and latency. Understanding vehicle density helps in optimizing routing decisions to avoid congested areas and improve overall network performance.

Vehicle Distance (Di): vehicle distance is the Euclidean distance of the vehicle in the route from the destination.

$$Di(r,t) = \exp\left(-\left(\frac{d_\text{i}}{d_\text{ref}}\right)^{\zeta}\right) \tag{8}$$

where: d_i denotes the Euclidean distance of the vehicle from the destination, d_ref is a reference distance, ζ is an attenuation factor.

Vehicle distance provides insight into the potential longevity of communication links, allowing for better route stability. Routes with shorter distances to the destination are preferred, as they are likely to offer more reliable communication.

Vehicle Trajectory (T): vehicle trajectory measures the predicted path toward the destination for the vehicle in the route, considering its current and future positions. This metric helps in assessing the alignment of vehicle movements with the desired route.

$$T(r,t) = \exp\left(-\left(\frac{\Delta d(t)^{\zeta}}{d_\text{ref}}\right)\right) \tag{9}$$

$\Delta d(t)^{\zeta} = d(t)^{\zeta} - d(0)^{\zeta}$, $d(t)$ denotes the future distance of the vehicle in the route to the destination at time t, $d(0)$ is the current distance to the destination, d_ref is a reference distance, ζ is an attenuation factor.

Vehicle trajectory helps in predicting the future alignment of vehicles with the intended route, ensuring that the selected path remains optimal over time. Routes with stable trajectories are preferred as they minimize deviations and potential disruptions in communication.

To predict the next position (x_{i+1}, y_{i+1}) of the vehicle, we use the current velocity vector (v_x, v_y) and the time interval Δt:

$$x_{i+1} = x_i + v_x \cdot \Delta t \tag{10}$$

$$y_{i+1} = y_i + v_y \cdot \Delta t \tag{11}$$

(x_i, y_i) are the current coordinates of the vehicle, v_x and v_y are the velocity components in the x and y directions, respectively, and Δt is the time interval. The overall score for each route, S, is computed as follows:

$$\begin{aligned} S(r,t) = w_1 \cdot \frac{1}{L(r,t)} + w_2 \cdot B(r,t) + w_3 \cdot R(r,t) + w_4 \cdot BE(r,t) \\ + w_5 \cdot \frac{1}{De(r,t)} + w_6 \cdot \frac{1}{Di(r,t)} + w_7 \cdot \frac{1}{T(r,t)} \end{aligned} \tag{12}$$

Weights $w_1, w_2, w_3, w_4, w_5, w_6$, and w_7 are referred to as latency, bandwidth, reliability, beamforming efficacy, density, distance, and trajectory, respectively. The scoring function $S(r,t)$ for VANETs is defined as follows:

$$S(r,t) = \alpha \cdot \sum_{i=1}^{n} w_i(t) \cdot v_i(r) + \delta \cdot P_{prediction}(r,t) + \lambda \cdot BF(r,t) \tag{13}$$

In Equation (13), $v_i(r)$ represent the route metrics, $w_i(t)$ are the time-dependent weights, α is a weighting factor for the summation real-time term, δ integrates the predictive reliability score $P_{prediction}(r,t)$, λ is a weighting factor for the beamforming score $BF(r,t)$.

An increasing of De leads to the spread of communication delays and potential packet collisions. Consequently, the score for routes through such areas would decrease, signaling less optimal paths. As Di increases, the vehicle is farther from the destination, likely leading to less reliable communication links. The score for such routes decreases because longer distances can result in higher latency and potential disconnections. An increase in T (indicating a less predictable or more erratic trajectory) implies that the vehicle's future positions are less aligned with the optimal path, leading to potential disruptions in communication. This decreases the route score.

Numerical Illustration for VANET

To provide a concrete understanding of the MMS-DSR scoring mechanism for VANETs, we present a numerical example with a network topology, as illustrated in Figure 5. This hypothetical scenario considers several candidate routes, each with different metrics for latency, bandwidth, reliability, vehicle density, vehicle distance, vehicle trajectory, and beamforming efficacy (BE), typical of diverse urban paths.

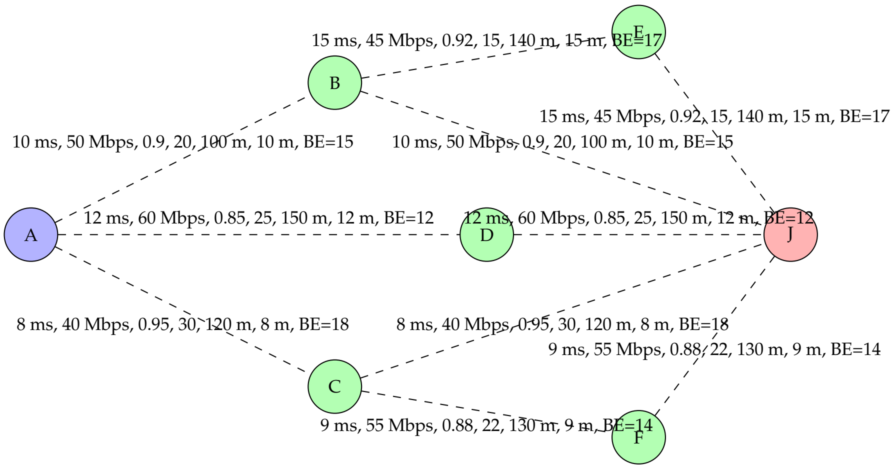
15 ms, 45 Mbps, 0.92, 15, 140 m, 15 m, BE=17
E
B
15 ms, 45 Mbps, 0.92, 15, 140 m, 15 m, BE=17
10 ms, 50 Mbps, 0.9, 20, 100 m, 10 m, BE=15
10 ms, 50 Mbps, 0.9, 20, 100 m, 10 m, BE=15
A
12 ms, 60 Mbps, 0.85, 25, 150 m, 12 m, BE=12
D
12 ms, 60 Mbps, 0.85, 25, 150 m, 12 m, BE=12
J
8 ms, 40 Mbps, 0.95, 30, 120 m, 8 m, BE=18
8 ms, 40 Mbps, 0.95, 30, 120 m, 8 m, BE=18
9 ms, 55 Mbps, 0.88, 22, 130 m, 9 m, BE=14
C
9 ms, 55 Mbps, 0.88, 22, 130 m, 9 m, BE=14
F

Figure 5. Network topology diagram illustrating the routes from Node A to Node J with respective metrics.

The routes and their respective metrics are summarized in Table 7.

Table 7. Example routes with associated metrics.

Route	L (ms)	B (Mbps)	R	De	Di (m)	T (m)	BE	P_{pr}(r,t)	BF(r,t)
r_1	10	50	0.9	20	100	10	15	0.92	20
r_2	8	40	0.95	30	120	8	18	0.87	22.5
r_3	12	60	0.85	25	150	12	12	0.90	15
r_4	15	45	0.92	15	140	15	17	0.91	21
r_5	9	55	0.88	22	130	9	14	0.89	19
r_6	10	50	0.9	20	100	10	15	0.92	20
r_7	8	40	0.95	30	120	8	18	0.87	22.5
r_8	12	60	0.85	25	150	12	12	0.90	20
r_9	15	45	0.92	15	140	15	17	0.91	22.5
r_{10}	9	55	0.88	22	130	9	14	0.90	15

Using the scoring function we assume the following parameters: $\delta = 0.2$, $\lambda = 0.3$, $\beta = 0$, Experience $E = 10$, dynamic weights at time t: $w_1(t) = 0.3$, $w_2(t) = 0.2$, $w_3(t) = 0.1$, $w_4(t) = 0.05$, $w_5(t) = 0.1$, $w_6(t) = 0.15$, $w_7(t) = 0.1$.

To ensure a fair comparison, the metrics are normalized to a [0, 1] scale using the formula shown in Equation (6).

The routes and their respective normalized metrics are summarized in Table 8.

Table 8. Normalized metrics for each route.

Route	L Norm	B Norm	R Norm	De Norm	Di Norm	T Norm	BE Norm	P_{pr} Norm(r,t)	BF Norm(r,t)
r1	0.286	0.50	0.5	0.20	0.0	0.20	0.500	1.0	0.667
r2	0.000	0.00	1.0	0.60	0.4	0.00	1.000	0.0	1.000
r3	0.571	1.00	0.0	0.40	1.0	0.40	0.000	0.6	0.000
r4	1.000	0.25	0.7	0.00	0.8	0.60	0.833	0.8	0.800
r5	0.143	0.75	0.3	0.35	0.6	0.25	0.333	0.4	0.533
r6	0.286	0.50	0.5	0.20	0.0	0.20	0.500	1.0	0.667
r7	0.000	0.00	1.0	0.60	0.4	0.00	1.000	0.0	1.000
r8	0.571	1.00	0.0	0.40	1.0	0.40	0.000	0.6	0.667
r9	1.000	0.25	0.7	0.00	0.8	0.60	0.833	0.8	1.000
r10	0.143	0.75	0.3	0.35	0.6	0.25	0.333	0.6	0.000

The scoring function $S(r,t)$ for each route is calculated using the normalized values. For example for $r1$ we have the following:

$$S(r1,t) = 0.5429$$

Repeating the same calculations for each route we will obtain the values showed in Table 9:

Table 9. Scoring Function values for each route after normalization.

Route	Scoring Function (SF)
r1	0.5429
r2	0.5500
r3	0.4700
r4	0.675833
r5	0.530000
r6	0.5429
r7	0.5500
r8	0.4700
r9	0.675833
r10	0.530000

Even if by applying the scoring function, the paths $r4$ and $r9$ emerge as the optimal route with the highest score of 0.675833, we must remember that in the classic DSR protocol, the primary metric for selecting the optimal path is typically the number of hops. Carrying out the route analysis of the graph illustrated in Figure 6, the possible paths from source to destination and the related Scoring Function (SF) values are shown in Table 10:

Table 10. Paths, Routes, and Total Scoring Functions (SF).

Path	Routes	Total SF
A → B → J	r1 + r6	$0.5429 + 0.5429 = 1.0858$
A → C → J	r2 + r7	$0.5500 + 0.5500 = 1.1000$
A → D → J	r3 + r8	$0.4700 + 0.4700 = 0.9400$
A → B → E → J	r1 + r4 + r9	$0.5429 + 0.675833 + 0.675833 = 1.894566$
A → C → F → J	r2 + r5 + r10	$0.5500 + 0.530000 + 0.530000 = 1.610000$

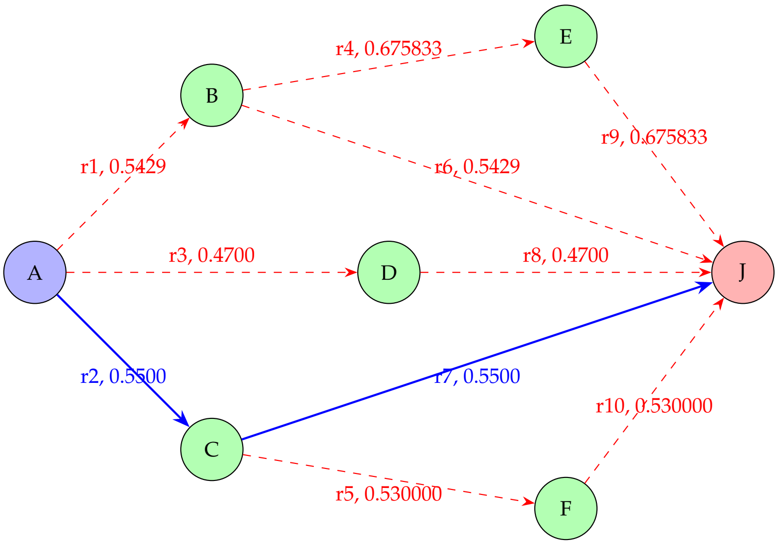

Figure 6. Network topology diagram illustrating the routes from Node A to Node J with respective metrics.

Considering that MMS-DSR prioritizes the route with the highest cumulative scoring function (SF) and minimum number of hops, the chosen optimal path would be the following:

$A \rightarrow C \rightarrow J$ with a total SF of 1.1000.

7. Simulation and Evaluation

Simulations were performed using OMNeT++ [37] integrated with SUMO [38] (Simulation of Urban MObility) and the inetframework to emulate realistic urban MANET and VANET scenarios. SUMO managed the vehicular traffic within the city, reflecting accurate vehicle movements and interactions, while inet supported advanced network simulations using the 802.11ax HE mode discussed in [39]. In order to accurately model radio signal propagation and the effect of urban environments, the RicianFading path loss model provided by the inet framework was used to account for both dominant line-of-sight signals and multiple reflected signals typical of urban settings. Additionally, the obstacles.xml configuration file was employed to emulate the presence of physical obstacles such as buildings and other structures, further enhancing the realism of the simulation. The simulations were configured to reflect the unique traffic and networking conditions of the Metropolitan area of Reggio Calabria (Italy). The global geo-coordinates for the four corners of the simulation area are x1: (38.09, 15.63), x2: (38.11, 15.64), x3: (38.94, 15.66), x4: (38.11, 15.66). Within this area, vehicles dynamically entered and exited the simulation, simulating real-world urban vehicular mobility patterns.

In this setup, the radio transmission ranges for RSUs were set to 500 m, while mobile nodes (vehicles) had a transmission range of 250 m, consistent with 802.11ax standards. Communication patterns were defined by a random communication model, in which vehicles dynamically selected another vehicle or Road-Side Unit (RSU) as their communication destination. This setup simulated vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication in an urban environment.

A centralized controller was responsible for managing data collection, processing, and updating the machine learning models used for routing decisions in the MMS-DSR protocol. The controller received real-time data from RSUs, analyzed them, and updated the multi-metric scoring and beamforming models every 10 s to reflect changes in traffic conditions and network dynamics. The overhead of the centralized control involved data processing and model updating, which was feasible for small to medium-sized cities such as Reggio Calabria.

The RSUs were strategically placed at major intersections and areas of high vehicle density to maximize MU-MIMO gains and beamforming efficiency, ensuring optimal coverage and minimal communication disruptions. The placement was determined based on traffic flow data, highlighting regions of frequent congestion and high vehicle mobility, making these locations critical for efficient traffic management.

In the considered scenario, a total of 15 RSUs were deployed. These units were crucial for the network's infrastructure, particularly for managing the high-speed vehicular communication required in urban settings. The RSUs were equipped with advanced MU-MIMO systems, enabling significant enhancements in signal directivity and strength through beamforming techniques. Table 11 includes the Simulation Parameter set.

Table 11. Simulation Parameters set.

Parameter	**Value**
Tone Configuration	26, 242, 2 × 996
Protocols	DSR, MMS-DSR, SOL-DSR, Enhanced OLSR
MIMO Configurations	10 × 10
Seed Variability	Multiple sets
Confidence Interval	95%
Simulation Time	1000 s
Number of Nodes	20 to 100
Simulation Area	2858 m × 1460 m
Data Rate	Up to 9607.5 Mbps
Network Standard	IEEE 802.11ax
Mobility Model	SUMO-based (Random Waypoint and Manhattan for additional analysis)
Traffic Type	UDP, CBR, VBR
TX Data App	UDPBasicApp
RX Data App	UDPSink
Packet Size	1500 bytes
Node Sensitivity	−120 dBm
Path Loss Model	RicianFading
Obstacle Model	Obstacles.xml (buildings, physical structures)
Simulation Tools	OMNeT++, SUMO, Veins
RSU Quantity	15 RSUs covering key traffic nodes
RSU Transmission Range	500 m
Mobile Node Transmission Range	250 m
Centralized Control Update Interval	10 s

7.1. Performance Metrics

The performance of the MMS-DSR protocol is assessed by focusing on throughput, end-to-end latency, and route discovery time under varying network loads and conditions. Throughput is a critical metric for measuring the network's capacity to deliver data successfully, reflecting the efficiency of the protocol in handling data transmission.

The comparison highlighted in Figure 7 shows that at 100 vehicles, MMS-DSR reaches a throughput of 9400 Mbps, significantly outperforming SOL-DSR's 6600 Mbps and Enhanced OLSR's 5000 Mbps. This substantial advantage can be attributed to the multi-metric scoring engine employed by MMS-DSR, which adjusts the weights of various routing metrics based on current network conditions. Combined with LSTM's predictive capabilities, this approach enhances route selection and management, enabling MMS-DSR to sustain high throughput levels even as network conditions fluctuate.

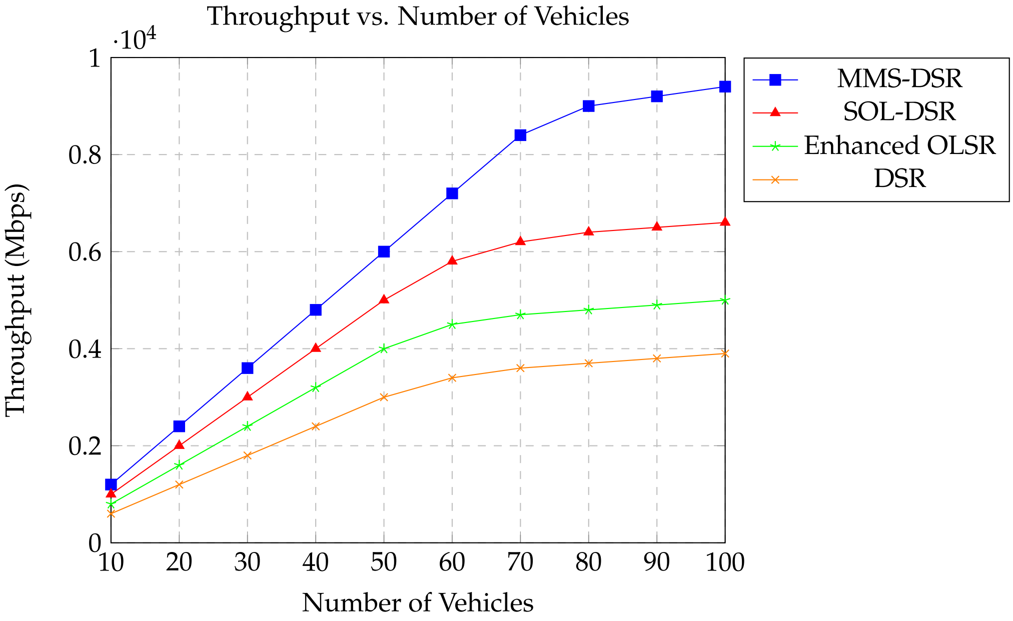

Figure 7. Throughput comparison across varying numbers of vehicles.

From Figure 7 it can be also observed that as the number of vehicles increases, MMS-DSR shows minimal efficiency loss, with the throughput increasing from 1200 Mbps at 10 vehicles to 9400 Mbps at 100 vehicles, showing less than a 1% drop in efficiency compared to initial performance, in contrast to traditional protocols which experience larger declines. In addition, when vehicle density increases from 50 to 100 vehicles, MMS-DSR maintains near-optimal throughput with only a minimal decrease in performance, in contrast to the more significant declines observed in SOL-DSR and Enhanced OLSR.

The Figure 8 also demonstrates that MMS-DSR provides good performance in terms of throughput across varying vehicle speeds. At 100 km/h, MMS-DSR achieves 9000 Mbps, which significantly surpasses SOL-DSR's 7300 Mbps and Enhanced OLSR's 6300 Mbps. This superior performance can be attributed to MMS-DSR's adaptive routing strategies and predictive capabilities that effectively manage the dynamic nature of vehicular movement.

As vehicle speed increases, MMS-DSR's ability to predict and adapt to changes in the network ensures efficient routing of data packets, minimizing disruptions and maintaining high throughput. For instance, even at high speeds, MMS-DSR's throughput decreases only slightly from 1100 Mbps at 10 km/h to 9000 Mbps at 100 km/h, showcasing its robustness and efficiency in high-mobility scenarios. This minimal decline in performance at higher speeds contrasts with the more significant decreases observed in other protocols.

Additionally, MMS-DSR's ability to manage network resources and prevent congestion becomes even more evident as vehicle speeds increase. The protocol's intelligent routing decisions, driven by LSTM predictions, ensure that data are transmitted smoothly even in high-speed environments, reducing packet loss and maintaining consistent throughput.

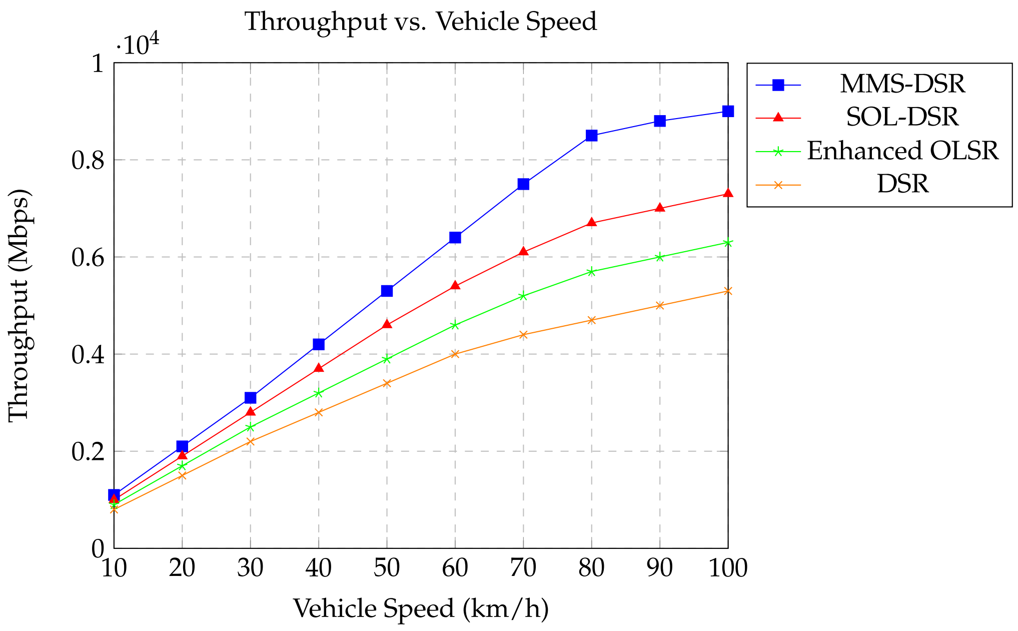

Figure 8. Throughput comparison across varying vehicle speeds.

7.1.1. Average End-to-End Latency

Latency measures the time it takes for packets to travel from the source to the destination, indicating the responsiveness of the network.

The Figure 9 demonstrates that MMS-DSR protocol provides a significant advantage in managing latency due that it increases from 20 ms at a density of 10 vehicles/km^2 to only 60 ms at 100 vehicles/km^2. This increase is notably lower compared to other protocols, such as SOL-DSR, which jumps from 25 ms to 75 ms, and Enhanced OLSR, which escalates from 30 ms to 85 ms under similar conditions. This remarkable performance by MMS-DSR can be primarily attributed to the integration of Long Short-Term Memory networks, which empower the protocol to proactively adjust routes in response to potential congestion and vehicle mobility, thereby avoiding common causes of delay.

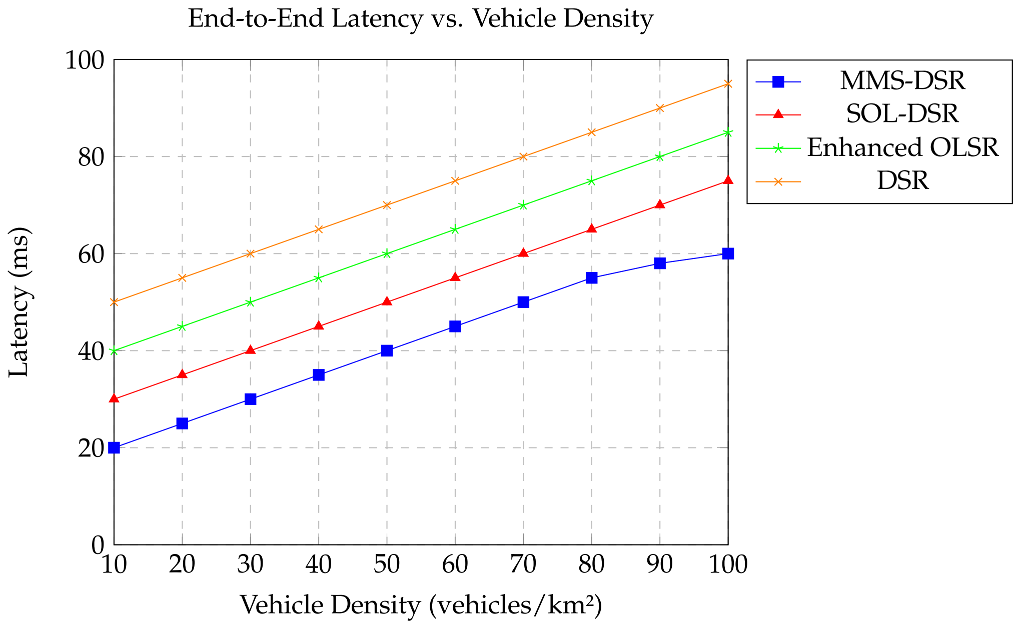

Figure 9. Latency comparison in function of vehicle density.

The trend illustrated in Figure 10 confirms that as vehicle density increases, MMS-DSR effectively uses its LSTM models to foresee and navigate around these potential bottlenecks. This proactive routing ensures that even in scenarios where vehicle density significantly increases, the latency remains optimally low. For example, when vehicle density increases from 50 to 100 vehicles/km^2, MMS-DSR exhibits only a minimal increase in latency from 40 ms to 60 ms, demonstrating an exceptional ability to manage and adapt to increased traffic without a corresponding steep rise in delay. In a dense urban environment characterized by frequent stop-and-go traffic and variable vehicle speeds, MMS-DSR's latency remains around 60 ms at 100 vehicles/km^2, significantly lower than the 75 ms for SOL-DSR and 85 ms for Enhanced OLSR.

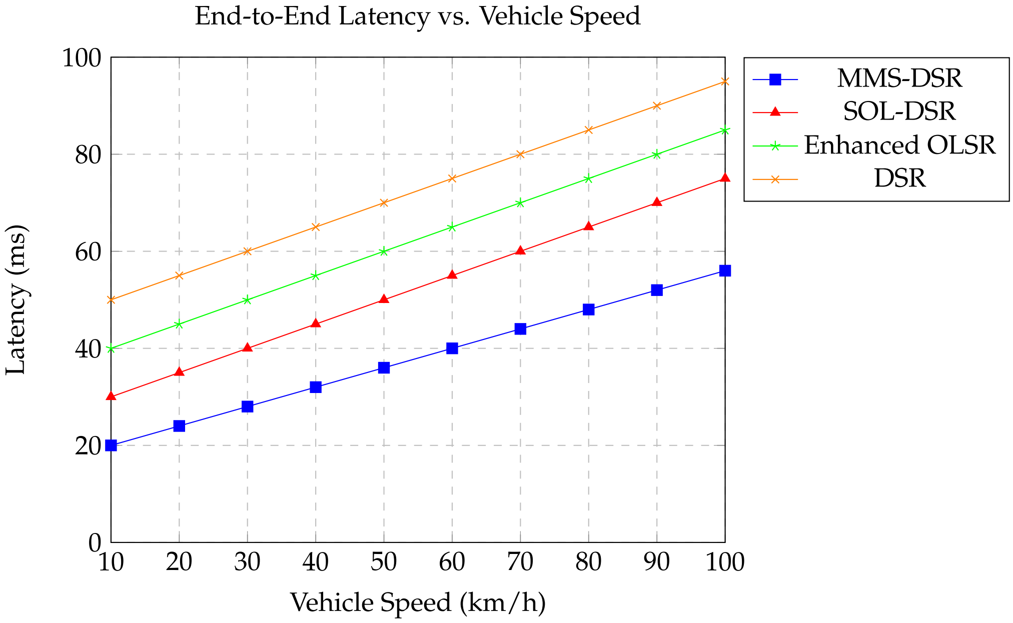

Figure 10. Latency comparison in function on vehicle speed.

7.1.2. Route Discovery Time

The time required to discover a route reflects the protocol's efficiency in establishing connectivity, especially in dynamic networks where rapid changes are common.

As evidenced by Figures 11 and 12 MMS-DSR demonstrates superior performance in this aspect, with route discovery times increasing from only 15 ms at a density of 10 vehicles/km^2 to 55 ms at 100 vehicles/km^2. This progressive increase is significantly slower compared to other protocols like SOL-DSR and Enhanced OLSR, where times escalate more steeply from 20 ms to 70 ms and from 25 ms to 80 ms, respectively, under similar conditions. This highlights the effectiveness of the integrated machine learning model in MMS-DSR, which uses Long Short-Term Memory networks to predict and adjust routes dynamically, facilitating quick and efficient pathfinding even as network complexity increases. As vehicle density improves along with network congestion, MMS-DSR efficiently predicts potential bottlenecks and reroutes data packets through less congested paths. This ability is reflected in the moderate increase in discovery times from 30 ms at 50 vehicles/km^2 to only 55 ms at 100 vehicles/km^2, demonstrating an exceptional capacity to handle increased traffic and complexity without a corresponding steep rise in discovery times.

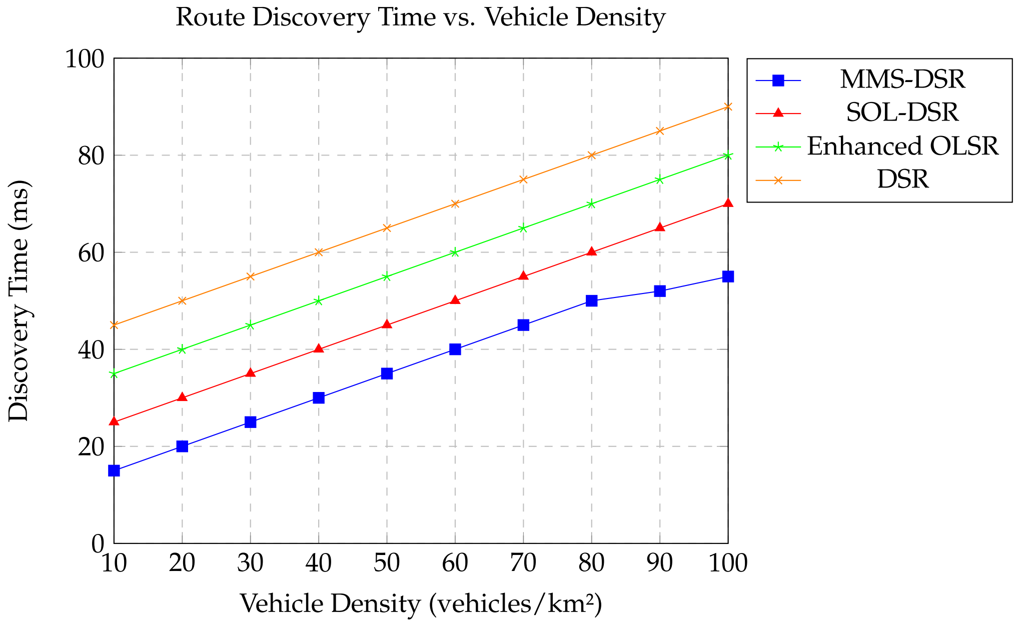

Figure 11. Route discovery time vs. vehicle density.

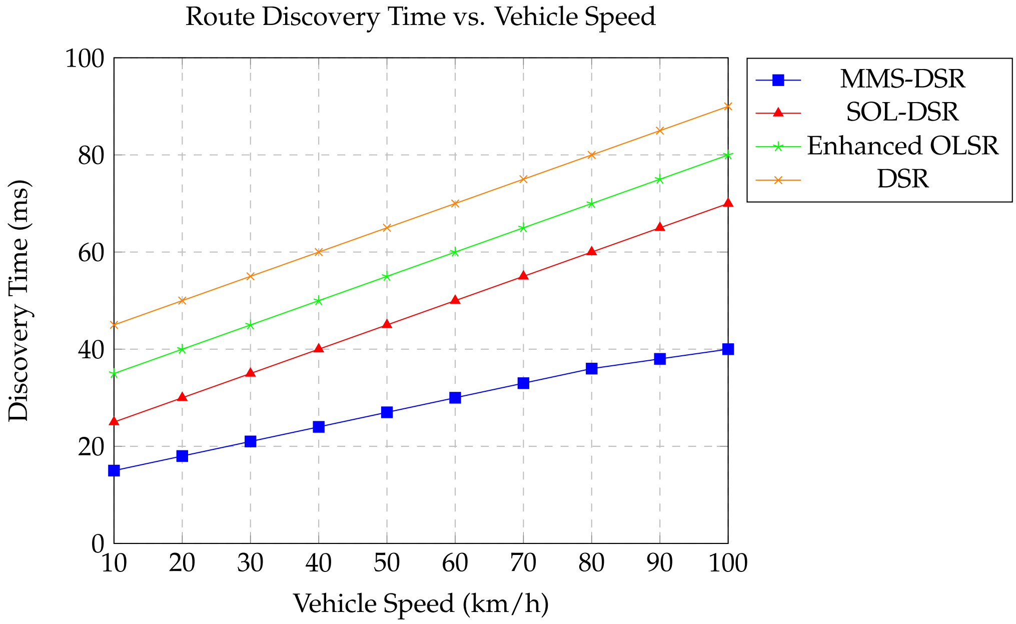

Figure 12. Route discovery time vs. vehicle speed.

7.1.3. Routing Overhead

Routing Overhead measures the extra communication required for maintaining the routing information, indicating the efficiency of the protocol.

MMS-DSR exhibits lower routing overhead compared to SOL-DSR, Enhanced OLSR, and traditional DSR, as depicted in Figures 13 and 14. The overhead increases from 200 control packets at 10 vehicles/km^2 to 1600 control packets at 100 vehicles/km^2 and from 200 control packets at 10 km/h to 1550 control packets at 100 km/h. This is significantly lower than the 2000 control packets required by Enhanced OLSR and DSR. MMS-DSR's on-demand routing strategy, combined with the predictive capabilities of LSTM networks, minimizes unnecessary control packet transmissions, thus reducing overhead and improving overall network efficiency.

A key advantage of MMS-DSR is its ability to predict vehicle movements without relying on GPS systems. By utilizing predictive models that anticipate vehicle trajectories, MMS-DSR reduces the need for frequent route updates, which significantly lowers the routing overhead. This approach ensures efficient use of network resources and enhances the protocol's scalability and adaptability in urban VANET environments.

As vehicle density and speed grow up, MMS-DSR maintains lower overhead by efficiently managing control packet transmissions and preventing unnecessary route discoveries. For example, at a vehicle density of 50 vehicles/km^2, MMS-DSR requires only 1000 control packets compared to 1500 for Enhanced OLSR.

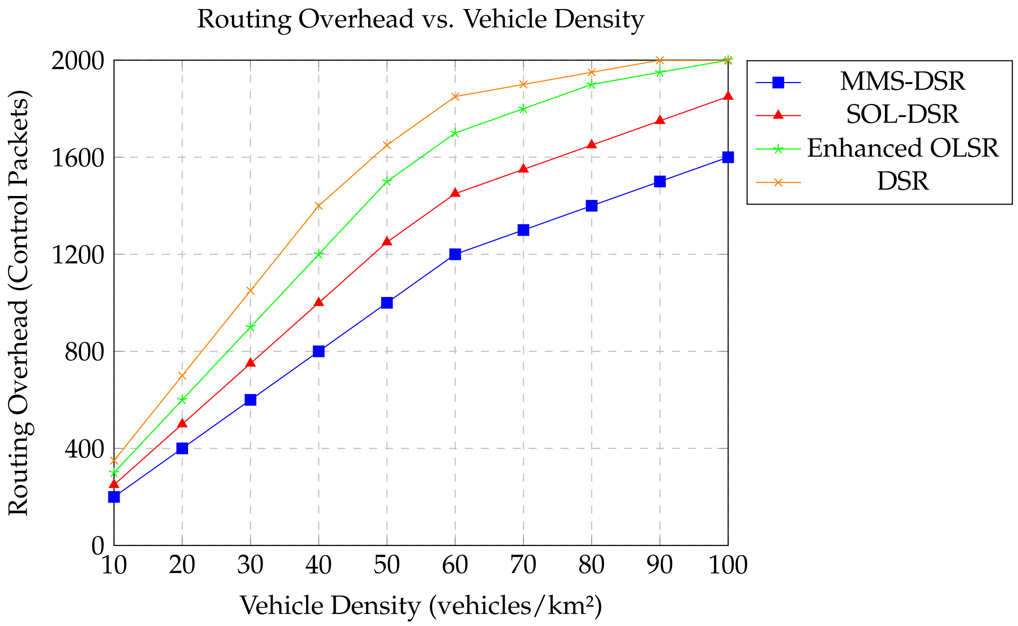

Figure 13. Routing overhead comparison in function of vehicle density.

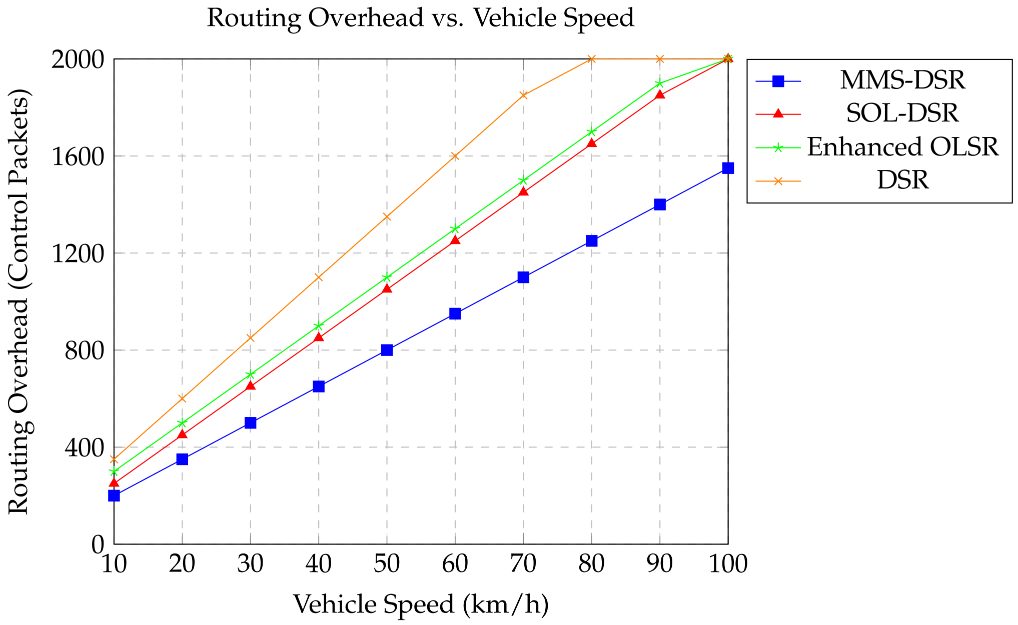

Figure 14. Routing overhead comparison in function of vehicle speed.

7.1.4. Scalability

Scalability examines the protocol's ability to handle increasing network sizes without significant performance degradation.

MMS-DSR demonstrates good scalability performance, as illustrated in Figure 15. The Normalized Performance Index decreases from 0.95 with 10 vehicles/km^2 to 0.63

with 100 vehicles/km^2 and from 0.95 at 10 km/h to 0.63 at 100 km/h. This decline is less pronounced compared to SOL-DSR, Enhanced OLSR, and DSR. MMS-DSR's machine learning-enhanced predictive capabilities and efficient routing strategies allow it to manage increasing network sizes effectively, ensuring consistent performance across various vehicle densities and speeds. This robustness is crucial for maintaining reliable communication in densely populated urban VANET environments.

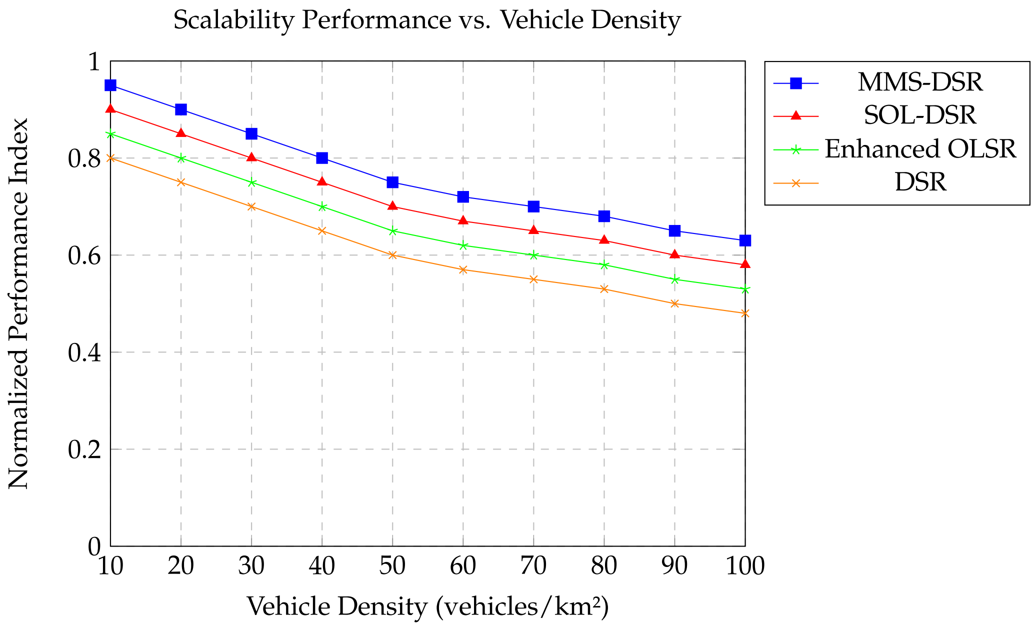

Figure 15. Scalability comparison performance across increasing vehicle density.

In high-density scenarios, MMS-DSR's predictive modeling using Long Short-Term Memory networks allows it to anticipate and adapt to changes in vehicle movements and network conditions dynamically. This capability ensures that even as vehicle density increases, MMS-DSR can maintain high performance with minimal degradation. The MMS-DSR approach benefits from its multi-metric scoring engine, which dynamically adjusts the weights of various routing metrics based on current network conditions. This adaptability is particularly beneficial in urban environments where vehicle densities and speeds can fluctuate rapidly. By expoliting predictive models, MMS-DSR can preemptively adjust routes to avoid congestion, ensuring that data transmission remains smooth and efficient. In addition, the protocol's efficiency is further enhanced by its reduced reliance on frequent route updates. Unlike traditional protocols that may rely on constant GPS data, MMS-DSR predicts vehicle movements and adjusts routes accordingly. This approach reduces the need for continuous control packet transmissions, lowering overhead and preserving network resources.

7.2. Sensitivity Analysis of Key Simulation Parameters

To further evaluate the robustness and adaptability of the MMS-DSR protocol, a sensitivity analysis was performed by varying key network parameters such as latency, bandwidth, and reliability. This analysis aims to explore the protocol's performance under different conditions, ensuring its effectiveness in dynamic urban vehicular environments, as previously discussed in the Simulation and Evaluation section.

The parameter of latency is critical in time-sensitive applications like vehicular ad-hoc networks (VANETs), where quick decision-making is crucial. The analysis examined latency variations ranging from 10 ms to 100 ms, assessing their impact on end-to-end delay, route discovery time and PDR. As expected, increasing the latency led to a proportional rise in end-to-end delay. However, the MMS-DSR protocol demonstrated resilience, with only a moderate increase in delay (15%) even as latency reached 100 ms. The route discovery time and PDR remained largely unaffected by this latency increase, maintaining a high level of performance. This observation is consistent with the findings presented in the Average End-to-End Latency subsection, as shown in Figures 9 and 10, where MMS-DSR consistently outperformed other protocols (SOL-DSR and Enhanced OLSR), particularly under higher latencies.

The sensitivity analysis explored the impact of varying bandwidth from 100 Mbps to 1000 Mbps, observing the corresponding effects on network throughput and routing overhead. The results showed a significant improvement in throughput as bandwidth increased, with an approximate 30% gain in throughput observed when bandwidth was increased to 1000 Mbps. Despite this improvement in throughput, the protocol maintained minimal routing overhead, as evidenced in Figures 7 and 8, where MMS-DSR consistently displayed superior scalability and efficiency across varying vehicle densities and speeds. These findings further align with the Routing Overhead results in Figures 13 and 14, where MMS-DSR exhibited significantly lower overhead compared to SOL-DSR and Enhanced OLSR, even in high-bandwidth scenarios.

Reliability was tested by introducing varying levels of packet loss, ranging from 0% to 15%, to evaluate the protocol's robustness under adverse conditions. The analysis demonstrated that even under high packet loss rates, MMS-DSR maintained a PDR of around 90%, underscoring its resilience. The machine learning-based prediction model integrated into MMS-DSR (via CNN-LSTM) allowed the protocol to anticipate network disruptions and adjust routing decisions dynamically, thus minimizing the impact of packet loss on overall performance. This adaptive capability aligns with the Packet Delivery Ratio results discussed in the simulation section, where the MMS-DSR protocol showed superior reliability in comparison to traditional DSR and other advanced routing protocols, even under fluctuating network conditions.

8. Conclusions and Future Directions

In this work, we proposed the Multi-metric Scoring Dynamic Source Routing protocol to address the challenges of efficient and adaptive routing in Vehicular Ad-hoc Networks within smart city environments. Through extensive simulations and empirical analysis, MMS-DSR has shown improvements in critical performance indicators such as throughput, latency, and route discovery time when compared to established protocols, including traditional DSR, SOL-DSR, and Enhanced OLSR. These findings suggest that MMS-DSR is a promising approach for optimizing network performance in dynamic and complex urban scenarios.

The MMS-DSR protocol incorporates machine learning techniques, specifically Long Short-Term Memory networks, to enable predictive routing, allowing it to adapt dynamically to rapidly changing urban traffic conditions. This method enhances communication efficiency and reliability by optimizing data paths and adjusting to network dynamics in real-time. Furthermore, the integration of advanced beamforming techniques improves communication quality by enhancing signal directionality and reducing interference, which is particularly important in dense urban environments.

Simulation results have proven the effectiveness of MMS-DSR across various urban scenarios, showing its ability to manage high mobility and frequent topological changes typical of VANETs. By utilizing the features of Massive MIMO and IEEE 802.11ax, MMS-DSR provides a scalable and adaptable routing protocol that is suited to the future requirements of 6G networks and smart city infrastructures.

Future Directions

Several future research directions are identified to further enhance the capabilities and extend the applicability of MMS-DSR:

- **Enhanced Security Mechanisms:** while this work focused on performance optimization, future research should incorporate advanced security protocols to protect against cyber threats. Ensuring data integrity and privacy in vehicular communications will require robust encryption standards and reliable authentication mechanisms.

- **Incorporation of Additional Machine Learning Techniques:** future studies could explore the use of other machine learning methods, such as deep reinforcement learning (DRL) and Support Vector Machines (SVMs), to improve the predictive capabilities and adaptability of MMS-DSR.
- **Scalability Improvements:** further research should investigate the scalability of MMS-DSR in larger urban networks, focusing on optimizing the protocol to handle thousands of nodes while maintaining low latency and efficient data handling as the network grows.
- **Real-World Implementation and Testing:** implementing MMS-DSR in real-world smart city environments will be a crucial next step. Pilot projects in collaboration with urban municipalities would provide valuable insights into the protocol's performance in live settings, helping to refine and improve its operational capabilities.
- **Development of Adaptive Beamforming Algorithms:** future work should focus on the development of more advanced beamforming algorithms capable of adapting to real-time changes in the urban environment. This includes addressing signal blockages and reflections more effectively to ensure continuous communication.
- **Support for Heterogeneous Networks:** expanding MMS-DSR to support heterogeneous networks, involving a mix of different wireless communication standards and technologies, will be important to increase the protocol's flexibility and applicability in various smart city scenarios.
- **Energy Efficiency:** future developments should consider energy-efficient routing and communication strategies. Designing algorithms that minimize power consumption while maintaining high performance will be essential for the sustainability of VANETs.
- **Collaborative Routing Strategies:** exploring collaborative routing strategies, where multiple nodes work together to optimize routes and manage traffic dynamically, could further improve the performance and reliability of MMS-DSR in complex urban environments.

Author Contributions: All authors contributed to the study conception and design. Material preparation, data collection and analysis were performed by V.I., D.G. and C.G. Conceptualization, V.I. and D.G.; methodology, C.G.; software, V.I.; validation, V.I., D.G. and C.G.; formal analysis, C.G.; investigation, V.I.; resources, D.G.; data curation, D.G.; writing—original draft preparation, V.I. and D.G.; writing—review and editing, C.G.; visualization, C.G.; supervision, C.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are contained within the article.

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

Appendix A. Fundamentals of CNN-LSTM Networks

To further clarify the Section 4, we provide a brief overview of CNN and LSTM networks, which are integral to the Machine Learning-Based Prediction Unit in the MMS-DSR approach.

A CNN is a type of Deep Learning model that excels at processing grid-like data structures, such as images or traffic data. CNNs use convolutional layers to automatically learn spatial hierarchies of features from input data. In the MMS-DSR protocol, CNNs are employed to extract spatial features from traffic data (e.g., vehicle positions, speed, and density).

CNNs operate by applying a set of learnable filters (or kernels) to the input data. Each filter slides over the input matrix, performing a convolution operation, which results in a feature map. These feature maps capture the local spatial patterns of the input data.

For example, if we have traffic data from K locations at multiple time instances, the input to the CNN layer can be represented as a matrix S, where each element corresponds to the traffic state at a specific location and time, as shown below:

$$S = \begin{bmatrix} s_{1,t-N} & s_{1,t-N+1} & \cdots & s_{1,t-1} \\ s_{2,t-N} & s_{2,t-N+1} & \cdots & s_{2,t-1} \\ \vdots & \vdots & \ddots & \vdots \\ s_{K,t-N} & s_{K,t-N+1} & \cdots & s_{K,t-1} \end{bmatrix}$$

The CNN applies filters to this matrix to generate high-level spatial feature maps, capturing patterns and relationships in the traffic data. The result is passed to the next stage of processing: the LSTM network.

LSTM networks are a type of RNN designed to handle sequential data and capture temporal dependencies. They are particularly effective at learning long-term relationships in time-series data, which makes them ideal for predicting future traffic states based on past observations.

LSTM networks consist of specialized units (LSTM cells) that regulate the flow of information through input gates, forget gates, and output gates. These gates allow the network to remember or forget certain pieces of information over time, making LSTMs robust in handling time-dependent data.

In the context of MMS-DSR, LSTM networks process the spatial features extracted by the CNN to learn temporal patterns in traffic data. The high-level spatial feature map $\mathbf{X}$, obtained from the CNN, is fed into the LSTM for sequential processing. The LSTM generates predictions of future network states (e.g., latency, bandwidth, and reliability) based on this input.

The hybrid CNN-LSTM model used in MMS-DSR combines the strengths of CNNs and LSTMs to extract both spatial and temporal features from traffic data. The CNN first captures spatial patterns in the data, while the LSTM captures the temporal evolution of these patterns over time.

The spatial feature map output by the CNN is denoted as follows:

$$X = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_L \end{bmatrix}$$

where X represents the spatial features, and each x_l corresponds to the feature at time t. This spatial feature map is then fed into the LSTM to capture temporal dependencies. The final output of the CNN-LSTM model is used to compute the predictive reliability score $P_{prediction}(r,t)$, which plays a crucial role in the MMS-DSR multi-metric scoring function.

References

1. Al-shareeda, M.A.; Alazzawi, M.A.; Anbar, M.; Manickam, S.; Al-Ani, A.K. A comprehensive survey on vehicular ad hoc networks (vanets). In Proceedings of the 2021 International Conference on Advanced Computer Applications (ACA), Cairo, Egypt, 20–22 March 2021; pp. 156–160.
2. Yogarayan, S. Wireless ad hoc network of manet, vanet, fanet and sanet: A review. *J. Telecommun. Electron. Comput. Eng. (JTEC)* **2021**, *13*, 13–18.
3. Reddy, A.S. Performance of VANET over MANET in Mobile Computing Environment. In Proceedings of the 2022 7th International Conference on Communication and Electronics Systems (ICCES), Coimbatore, India, 22–24 June 2022; pp. 659–664.
4. Quy, V.K.; Nam, V.H.; Linh, D.M.; Ban, N.T.; Han, N.D. Communication solutions for vehicle ad-hoc network in smart cities environment: A comprehensive survey. *Wirel. Pers. Commun.* **2022**, *122*, 2791–2815. [CrossRef]
5. Amir, E.A.; Altabbakh, S.M.; Beshay, S.T.; Gomaa, I. Performance Evaluation of VANET Statistical Broadcast Protocols for Smart Cities. In Proceedings of the 2023 20th ACS/IEEE International Conference on Computer Systems and Applications (AICCSA), Giza, Egypt, 4–7 December 2023; pp. 1–8.

6. Sumit; Chhillar, R.S.; Dalal, S.; Dalal, S.; Lilhore, U.K.; Samiya, S. A dynamic and optimized routing approach for VANET communication in smart cities to secure intelligent transportation system via a chaotic multi-verse optimization algorithm. *Clust. Comput.* **2024**, *27*, 7023–7048. [CrossRef]
7. Abir, M.A.B.S.; Chowdhury, M.Z.; Jang, Y.M. Software-defined uav networks for 6g systems: Requirements, opportunities, emerging techniques, challenges, and research directions. *IEEE Open J. Commun. Soc.* **2023**, *4*, 2487–2547. [CrossRef]
8. Vaigandla, K.K.; Bolla, S.; Karne, R. A survey on future generation wireless communications-6G: Requirements, technologies, challenges and applications. *Int. J. Adv. Trends Comput. Sci. Eng.* **2021**, *10*, 3067–3076.
9. Waleed, S.; Ullah, I.; Khan, W.U.; Rehman, A.U.; Rahman, T.; Li, S. Resource allocation of 5G network by exploiting particle swarm optimization. *Iran J. Comput. Sci.* **2021**, *4*, 211–219. [CrossRef]
10. Khan, I.; Zhang, K.; Wu, Q.; Ullah, I.; Ali, L.; Ullah, H.; Rahman, S.U. A wideband high-isolation microstrip MIMO circularly-polarized antenna based on parasitic elements. *Materials* **2022**, *16*, 103. [CrossRef]
11. Gururaj, H.; Natarajan, R.; Almujally, N.A.; Flammini, F.; Krishna, S.; Gupta, S.K. Collaborative energy-efficient routing protocol for sustainable communication in 5G/6G wireless sensor networks. *IEEE Open J. Commun. Soc.* **2023**, *4*, 2050–2061. [CrossRef]
12. Raddo, T.R.; Rommel, S.; Cimoli, B.; Vagionas, C.; Perez-Galacho, D.; Pikasis, E.; Grivas, E.; Ntontin, K.; Katsikis, M.; Kritharidis, D.; et al. Transition technologies towards 6G networks. *EURASIP J. Wirel. Commun. Netw.* **2021**, *2021*, 100. [CrossRef]
13. Salameh, A.I.; El Tarhuni, M. From 5G to 6G—Challenges, technologies, and applications. *Future Internet* **2022**, *14*, 117. [CrossRef]
14. Dorothy, A.B.; Sreeja, B.; Chamundeeswari, V.; Madala, R.; Singh, D.P.; Sucharitha, K. 6G Networks in Unmanned Ariel Vehicle Intelligent Internet of Things Secured Communication. In Proceedings of the 2023 6th International Conference on Contemporary Computing and Informatics (IC3I), Gautam Buddha Nagar, India, 14–16 September 2023; Volume 6, pp. 1374–1378.
15. Al Ridhawi, I.; Aloqaily, M.; Karray, F. Intelligent blockchain-enabled communication and services: Solutions for moving internet of things devices. *IEEE Robot. Autom. Mag.* **2022**, *2*, 10–20. [CrossRef]
16. Gupta, R.; Nair, A.; Tanwar, S.; Kumar, N. Blockchain-assisted secure UAV communication in 6G environment: Architecture, opportunities, and challenges. *IET Commun.* **2021**, *15*, 1352–1367. [CrossRef]
17. Albattah, W.; Habib, S.; Alsharekh, M.F.; Islam, M.; Albahli, S.; Dewi, D.A. An overview of the current challenges, trends, and protocols in the field of vehicular communication. *Electronics* **2022**, *11*, 3581. [CrossRef]
18. IEEE. IEEE Standard for Information Technology–Telecommunications and Information Exchange between Systems Local and Metropolitan Area Networks–Specific Requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 1: Enhancements for High-Efficiency WLAN. In *IEEE Std 802.11ax-2021 (Amendment to IEEE Std 802.11-2020)*; IEEE: Piscataway, NJ, USA, 2021; pp. 1–767. [CrossRef]
19. Almazok, S.A.; Bilgehan, B. A novel dynamic source routing (DSR) protocol based on minimum execution time scheduling and moth flame optimization (MET-MFO). *EURASIP J. Wirel. Commun. Netw.* **2020**, *2020*, 219. [CrossRef]
20. Quy, V.K.; Ban, N.T.; Han, N.D. A multi-metric routing protocol to improve the achievable performance of mobile ad hoc networks. In *Modern Approaches for Intelligent Information and Database Systems*; Springer: Berlin/Heidelberg, Germany, 2018; pp. 445–453.
21. Zarzoor, A.R. Enhancing dynamic source routing (DSR) protocol performance based on link quality metrics. In Proceedings of the 2021 International Seminar on Application for Technology of Information and Communication (iSemantic), Semarangin, Indonesia, 18–19 September 2021; pp. 17–21.
22. Natarajan, B.; Shanmugam, A. A trust based model to mitigate black hole attacks in DSR based manet. *Eur. J. Sci. Res.* **2011**, *50*, 6–15.
23. Liang, Q.; Lin, T.; Wu, F.; Zhang, F.; Xiong, W. A dynamic source routing protocol based on path reliability and link monitoring repair. *PLoS ONE* **2021**, *16*, e0251548. [CrossRef]
24. Gao, S.; Li, H.; Zhang, Q. An optimized routing protocol for MANETs based on multi-metric dynamic weighting. In Proceedings of the 2022 IEEE 5th International Conference on Electronics Technology (ICET), Chengdu, China, 13–16 May 2022; pp. 954–958.
25. Shams, E.A.; Rizaner, A.; Ulusoy, A.H. Trust aware support vector machine intrusion detection and prevention system in vehicular ad hoc networks. *Comput. Secur.* **2018**, *78*, 245–254. [CrossRef]
26. Laanaoui, M.; Raghay, S. Enhancing OLSR protocol by an advanced greedy forwarding mechanism for VANET in smart cities. *Smart Cities* **2022**, *5*, 650–667. [CrossRef]
27. Bajracharya, R.; Shrestha, R.; Kim, S.; Jung, H. 6G NR-U based wireless infrastructure UAV: Standardization, opportunities, challenges and future scopes. *IEEE Access* **2022**, *10*, 30536–30555. [CrossRef]
28. Sharma, S.; Mahajan, A.N.; Poonia, R.C. SA-DSR: A Bandwidth Optimizing Technic for Dynamic Source Routing Protocol. *Int. J. Eng. Adv. Technol. (IJEAT)* **2020**, *9*, 3006–3009. [CrossRef]
29. Duan, Z.; Yang, Y.; Zhang, K.; Ni, Y.; Bajgain, S. Improved deep hybrid networks for urban traffic flow prediction using trajectory data. *IEEE Access* **2018**, *6*, 31820–31827. [CrossRef]
30. Liu, Y.; Zheng, H.; Feng, X.; Chen, Z. Short-term traffic flow prediction with Conv-LSTM. In Proceedings of the 2017 9th International Conference on Wireless Communications and Signal Processing (WCSP), Nanjing, China, 11–13 October 2017; pp. 1–6.
31. Sainath, T.N.; Vinyals, O.; Senior, A.; Sak, H. Convolutional, long short-term memory, fully connected deep neural networks. In Proceedings of the 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), South Brisbane, QLD, Australia, 19–24 April 2015; pp. 4580–4584.

32. Wu, Y.; Tan, H. Short-term traffic flow forecasting with spatial-temporal correlation in a hybrid deep learning framework. *arXiv* **2016**, arXiv:1612.01022.
33. Wang, S. Traffic State Prediction and Traffic Control Strategy for Intelligent Transportation Systems. In *Intelligent Electronics and Circuits-Terahertz, ITS, and Beyond*; IntechOpen: London, UK, 2022.
34. Zheng, H.; Lin, F.; Feng, X.; Chen, Y. A hybrid deep learning model with attention-based conv-LSTM networks for short-term traffic flow prediction. *IEEE Trans. Intell. Transp. Syst.* **2020**, *22*, 6910–6920. [CrossRef]
35. Tripp-Barba, C.; Urquiza-Aguiar, L.; Igartua, M.A.; Rebollo-Monedero, D.; De la Cruz Llopis, L.J.; Mezher, A.M.; Aguilar-Calderón, J.A. A multimetric, map-aware routing protocol for VANETs in urban areas. *Sensors* **2014**, *14*, 2199–2224. [CrossRef]
36. Aliyu, A.; Abdullah, A.H.; Isnin, I.F.; Radzi, R.Z.R.M.; Kumar, A.; Darwish, T.S.; Joda, U.M. Road-based multi-metric forwarder evaluation for multipath video streaming in urban vehicular communication. *Electronics* **2020**, *9*, 1663. [CrossRef]
37. Varga, A. OMNeT++ Community. OMNeT++ Discrete Event Simulator (Version 6.0). OpenSim Ltd. 2022. Available online: https://omnetpp.org/ (accessed on 13 April 2022).
38. Lopez, P.A.; Behrisch, M.; Bieker-Walz, L.; Erdmann, J.; Flötteröd, Y.P.; Hilbrich, R.; Lücken, L.; Rummel, J.; Wagner, P.; Wießner, E. Microscopic Traffic Simulation using SUMO. In Proceedings of the The 21st IEEE International Conference on Intelligent Transportation Systems, Maui, HI, USA, 4–7 November 2018.
39. Inzillo, V.; Ariza Quintana, A. Implementation of 802.11ax and cell free massive MIMO scenario for 6G wireless network analysis extending omnet++ simulator. *Simul. Trans. Soc. Model. Simul. Int.* **2024**. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.