DyPoC: Dynamic Proximity Aware Caching

Journal:	*IEEE Vehicular Technology Magazine*
Manuscript ID	Draft
Manuscript Type:	Open Call Paper (S1)
Date Submitted by the Author:	n/a

Complete List of Authors:	Tarkhani, Ahmadreza; Amirkabir University of Technology Department of Computer Engineering, Computer Engineering Shirali-Shahreza, Sajad; Amirkabir University of Technology Department of Computer Engineering, Computer Engineering

Keywords:	Vehicular Network, High-Definition Map, Content Caching

DyPoC: Dynamic Proximity Aware Caching

Ahmadreza Tarkhani, Amirkabir University of Technology

Sajad Shirali-Shahreza, Amirkabir University of Technology

In Vehicular Ad Hoc Networks (VANETs), caching techniques hold promise for improving user experience and reducing the network's core load. While past research has focused on optimizing non-transient content caching without regional specificity, many traffic services such as HD maps, parking management, and dynamic route guidance rely on transient data with expiration dates and regional relevance. To address this, we propose DyPoC (Dynamic Proximity-Aware Caching), a novel caching policy tailored for region-specific transient data. DyPoC accounts for factors like vehicle distance from the region of interest, data freshness, and vehicle trajectories to determine what to cache or remove. Our evaluation using HD map data shows that DyPoC outperforms traditional caching policies, improving the mean hit ratio by 15%.

CCS Concepts: • Networks   Mobile networks; Wireless access networks.

Keywords: Vehicular Network, High-Definition Map, Content Caching

Introduction

In contemporary transportation, VANETs have emerged as an innovative paradigm, seamlessly integrating vehicles into interconnected systems. Diverging from conventional networks [1], VANETs harness the mobility of vehicles to establish dynamic communication links, resulting in a decentralized and self-organizing network structure. Communication in VANETs encompasses Vehicle-to-Vehicle (V2V), Vehicle-to-Infrastructure (V2I), and Vehicle-to-Network (V2N) channels [2] (Figure 1).

V2V communication is particularly essential, enabling direct interactions between vehicles on the road. This facilitates the real-time exchange of information such as traffic conditions, road hazards, and maps. Such peer-to-peer communication constitutes the essence of VANETs, fostering collaborative efforts among vehicles to enhance safety and traffic efficiency.

In tandem with V2V communication, V2I communication extends the network to infrastructure elements like Road-Side Units (RSU), traffic lights, and other fixed installations along the route. This interaction augments the network's capabilities by providing resources at its periphery. Finally, V2N communication allows vehicles to establish direct internet connectivity through cellular networks and access resources that are not available locally.

Within the dynamic VANETs landscape, characterized by continuous information exchange for safety, efficiency, and an improved driving experience, caching emerges as a pivotal concept [3]. The motivation behind caching in VANETs stems from the necessity to efficiently manage and retrieve data, addressing the unique challenges posed by vehicular networks. The intermittent connectivity and swift mobility of vehicles in VANETs create circumstances where traditional data retrieval methods may encounter limitations.

Caching can address these challenges by strategically storing and managing data at various points within the network. This proactive approach ensures that relevant information is readily available and reduces the reliance on constant real-time communication for data retrieval. Caching also minimizes data retrieval latency by providing vehicles with access to frequently requested information locally. This is particularly crucial in time sensitive applications such as autonomous driving, path planning, and collision warnings, where reducing communication delays can significantly enhance safety and efficiency.

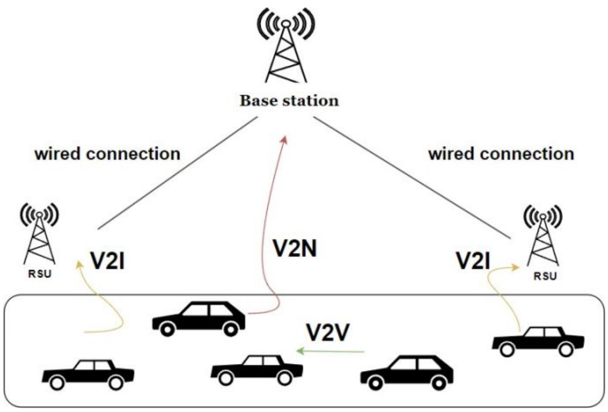

Figure 1 *Communication in VANET.*

By locally storing and retrieving frequently used data, caching also reduces the overall burden on the VANETs infrastructure. This is beneficial for optimizing network resources, minimizing congestion, and ensuring efficient data dissemination. The intermittent connectivity in VANETs may result in temporary communication blackouts. Caching mitigates the impact of such disruptions by allowing vehicles to access cached data during communication gaps, ensuring a more reliable and continuous flow of information.

Due to the constrained onboard storage capacity of vehicles, caching all data is unfeasible. Therefore, the main problem in caching is the strategic selection of which data to cache. Existing caching policies in VANETs are often tailored to non-transient content [4, 5], where files commonly lack expiration times or region-specific relevance.

In contrast, traffic information such as HD maps are inherently transient [6] and only valid for a limited duration. Furthermore, they usually exhibit regionality and are beneficial only for vehicles within a specific region.

This paper introduces *DyPoC*, a caching policy specifically designed for handling transient data, such as traffic information. DyPoC considers factors such as data freshness, proximity of vehicles to the area of interest, and the path that the vehicle plans to travel along when making caching decisions. Our approach emphasizes caching data that are most relevant to the current location and the path of the vehicle, while also considering residual freshness. In our simulation, DyPoC demonstrates a significantly better performance for caching contents such as HD maps with 15% improvement in the mean hit ratio compared to other caching policies.

The remainder of this paper is structured as follows: We begin by reviewing related works and highlighting our contributions within the existing body of research. Next, we define the model that we use for the VANET, the scope of this paper, and the specific problem we address within the VANET system model. We then present our proposed caching policy (DyPoC). Following this, we describe our simulation setup and evaluate our mechanism in comparison with state-of-the-art policies. Finally, we discuss the limitations of our approach and outline avenues for future research.

Related Work

The majority of existing research on edge and IoT caching predominantly focuses on caching data that is not transient data. This creates a noticeable gap in addressing caching for transient IoT data. We now review the few notable exceptions that considered transient data.

Vural et al.'s caching policy [7] incorporated data transiency into their caching policies for Internet content routers. They introduced an analytical model that considers variables like the lifetime of data items and the rate of received requests, providing insights into the trade-off between multi- hop communication costs and data freshness.

In [8], a cooperative in-network caching scheme for information-centric networking based on the Least Recently Used (LRU) replacement policy is introduced. This scheme takes into account the lifetime of IoT data by establishing a freshness threshold for each node to determine whether or not to cache the data.

Amadeo et al.'s Freshness-Driven Caching (FDC) [9] is a noteworthy caching strategy explicitly designed for transient data in VANETs. The FDC strategy considers the residual lifetime of content when making caching decisions. In [10], they further investigate caching decisions by considering residual lifetime, popularity (measured as the ratio of re- quests for a file to the maximum number of requests), and availability (expressed as the number of hops required for content retrieval).

Ostrovskaya et al.'s Multi-Metric Cache Replacement Policy (M2CRP) [11], considers freshness (in terms of how long the item has been cached), frequency of requests for cached data, and the distance between the vehicle and the location where the file was cached for making caching decisions.

Meddeb et al.'s study presents the Least Fresh First (LFF) [12] replacement policy, which focuses on removing the least fresh content from the cache. This policy leverages time series analysis as a predictive tool to anticipate future events and estimate the remaining lifespan of cached contents.

Our work distinguishes itself by being considering the path that vehicle plan to travel, freshness (in term of remain- ing valid time before expiration of the data), and the driving distance of vehicles to the area of interest when making caching decisions in vehicles. This holistic approach aims to enhance the efficiency and effectiveness of caching strategies in dynamic vehicular environments. We evaluated our caching policy for HD map transmission, which is a transient data with regional relevance. Our simulation results demonstrate significantly better performance with at least 15% improvement in the mean hit ratio compared to existing caching policies.

VANET Model

Before describing our caching policy, we define the model that we use to represent the VANET network. In our VANET model, vehicles within a city already selected the path that they will travel. The city is modelled as an intricate road network represented as a graph. To facilitate their travel, vehicles need to acquire high-definition (HD) maps of the city's complex road infrastructure along their path (edges of the city graph). The goal is algorithm is to efficiently cache these HD maps in vehicles.

Each HD map has attributes such as expiration time (considering the dynamic nature of traffic information), coordinates, radius of interest, and the associated road name. The priority for acquiring map data is determined by the proximity of the information to the vehicle's present location, because the navigation system requires the maps based on the order that the car will travel through them. This ensures that the caching prioritizes obtaining data that will be used soon.

These HD maps consist of a static segment and a dynamic component, the latter having validity limited to a specified timestamp. The dynamic aspect is further divided into quasi-static data, relevant for a brief duration (typically a few minutes), and real-time data, which expires almost instantly, making caching of real-time HD data ineffective. Therefore, crowd-sourcing or comparable methodologies become essential for obtaining real-time data, as discussed in Zhang et al. [13], which is beyond the scope of this paper. We only focus on caching the static and quasi-static parts of the HD maps in our approach.

Vehicles can obtain HD maps from different sources: nearby vehicles that have those maps (V2V), Road-Side Units (RSUs) that proactively fetched and have necessary maps (V2I), and the cellular network connected to backbone servers that contain all maps (V2N).

The optimal strategy for a vehicle is to fetch HD maps from nearby vehicle traveling the same route. This will reduce latency to obtain the data and reduces the strain on the infrastructure and cellular network.

The next preferred option is to retrieve maps from RSUs when within their range. The least favourable scenario arises when a vehicle is in close proximity to a road and cannot fetch the map data from either of these sources. In such instances, as shown in Figure 1, the vehicle must establish a direct connection to the cellular network. This option is usually more expensive (due to costs of using cellular network) and has higher latency. It can also add significant load to the backbone network due to the number of vehicles and the size of the maps.

In terms of communication setup, vehicles and RSUs employ the WAVE (Wireless Access in Vehicular Environments) protocol for side-link signalling tasks, including the broadcast of required maps or the establishment of unicast channels for file transfers. This choice stems from the WAVE protocol being the prevailing communication standard for VANETs.

However, due to the WAVE protocol's constrained bandwidth, it is not feasible to transfer the HD maps over that. To overcome this limitation, and in the absence of a widely-accepted standard, vehicles should utilize a V2V and V2I unicast connections to exchange HD maps. One potential unicast option is the 5G Device-to-Device (D2D) connections. This adaptive approach ensures effective communication for HD map-related tasks while navigating the current constraints posed by bandwidth limitations in the WAVE protocol.

We also assume that each vehicle will routinely broadcast requests for all HD maps that it needs to obtain along its designated path through one of the service channels within the WAVE protocol. The transmitted messages will also include information about supported unicast communication technologies such as WiMAX or 5G that it supports for the transmission of the map.

In the event that another vehicle or an RSU receives such a request and possesses one of the requested maps, it will initiate a unicast channel to the requester using one of the communication channels that both of them support. Once a unicast connection is established between them, the requested maps will be efficiently transmitted.

DyPoC : Dynamic Proximity-Aware Caching

The main idea of DyPoC is to selectively cache items with regionality only if they align with the path that the vehicle plans to travel along. This determination is made by comparing the area that the map covers and travel path of the vehicle. The logic behind this approach is that not only the vehicle needs this map for its own travel, but there is also a higher chance of encountering other vehicles in our path that are interested in it, enhancing the relevance and utility of the cached data.

Furthermore, because each map also has an expiration time, we further verify that we can reach the area that the map covers before it expires. This is calculated by considering the current speed and path. We only cache a map if these conditions are met. This ensures that we avoid caching items that would become invalid or provide irrelevant information, thereby ensuring the usefulness of the cached data.

If there is no space in the cache to put a new map in it (i.e., if the cache is full), then we use the same idea for replacement policy. We calculate the score for all cached items and the new candidate content. We then remove items with lower scores than the candidate content until there is enough space.

The following formula is used to calculate the score of each item:

$$\text{Score} = \frac{\text{Residual life}}{\text{Maximum residual life}} \times \frac{\text{Maximum driving distance}}{\text{Driving distance} + 1}$$

The first parameter influencing the score is the residual life of the item, measured in seconds. This value is determined by subtracting the expiration time (which is an attribute of the map) from the current time. If the result is negative, which means that the map is already expired, it is set to zero.

The residual life is then divided by the maximum residual life threshold. This threshold represents the maximum possible life of a map. For example, in scenario one which each map is only valid for 60 seconds, this threshold will be set to 60. This portion of the score prioritizes files with longer remaining validity, as they are more likely to be requested and shared before they expire.

The second parameter influencing the score is the current driving distance of the vehicle to reach that map area. It is measured in meters and calculated based on the travel path of the vehicle. The maximum driving distance threshold acts as a normalizing factor to limit the influence of the current driving distance on the overall score.

This component of the formula aims to prioritize items that are closer to the vehicle, as they will be needed sooner. Furthermore, maps of nearby roads are more likely to be requested by other vehicles travelling in the current road compared to maps of more distant roads.

To summarize, our caching policy (DyPoC) prioritizes the storage of the freshest items to ensure that they can be shared for a longer time (i.e., increasing their utilization). Furthermore, we emphasize the relevance of cached items by favouring those closely aligned with the vehicle's current location and path. This increases utilization as well because it is more probable to share maps while driving towards them, because it is more probable to share the road with other cars who are going to the same area.

Our approach aims to maximizes the utility of cached data and enhancing the overall effectiveness of our caching system in serving the dynamic needs of vehicles on the road. We will discuss the simulation results in the next section which confirms this.

Evaluation

We used *OMNeT++*[1] 6.0 as our primary network simulation tool and *SUMO*[2] 1.11.0 for vehicle movement simulation. We performed the simulation on a 3x3-kilometer map of Tehran city centre (Figure 2), sourced from OpenStreetMap[3]. This provides a realistic evaluation of our approach.

After pre-processing the raw map, we used the SUMO's *netconvert* tool to transform the processed map data into a SUMO-compatible road network format.

The next step used the *randomTrips* tool from SUMO to generates a set of random trips within the map via uniformly selecting source and destination nodes. These generated trips form the foundation for vehicular movement in the simulation.

To manage the routing of vehicles within the network, we utilize the *duarouter* tool from SUMO. This tool generates routes between specified source and destination nodes forthe participating vehicles in the simulation.

We performed the simulation for 900 seconds. Throughout the simulation time, we maintain a consistent count of 100 vehicles. This ensures that as vehicles reaches their destination and exit the simulation, new vehicles are replacing them. We have strategically placed 25 RSUs along the roadsand suitable locations such as intersections. The distance between these RSUs ranges from 500 to 700 meters, a distance proposed in [14, 15].

Figure 2 *City map used in simulation.*

We used *Veins*[4] 5.2 to simulate the WAVE Protocol stack and also as an interface between SUMO mobility and OMNeT. Vehicles will broadcast short WAVE messages periodically (once per second). In these messages, they request all necessary HD maps that they still need for the remaining part of their path.

These requests will be transmitted over a designated WAVE service channel responsible for traffic information. The size of each request message is capped at 1 kilobyte (1KB), which is calculated from the maximum number of maps that a vehicle needs to reach their destination.

If any vehicle in the vicinity possesses one of the requested HD maps mentioned in the broadcast, it will initiate a unicast channel to send the map to the requester. We set the bandwidth of this unicast channel at 1 MB/s data rate which is a speed comparable to a 5G-based Device-to-Device (D2D) connection. The communication medium will experience Nakagami-m fading, incorporating a stochastic fading model to simulate real-world wireless channel conditions. If the vehicle is not currently receiving any HD map data and is within the communication range of an RSU, it will download the required HD maps from an RSU with 1 MB/s data rate, with priority given to maps of nearby roads.

In our simulation model, for simplicity, we assume that each vehicle has selected the path that it will travel when it enters the network and does not change it during the travel. Therefore, it could request all necessary maps of the whole path from the beginning.

HD maps for each road are divided into smaller pieces. These pieces share a common road name and have a consistent size of 5 megabytes across the network. Each of these map pieces is further divided into 1-megabyte chunks,with each chunk representing a specific portion of the road. Transmission and caching decisions are made based on each chunk.

We experimented with different values of map lifetime and present the results of two scenarios here: 60 seconds (scenario 1) and 120 seconds (scenario 2). The expiry time is set when a map is created (e.g., in the server) and remains the same as it is shared between vehicles. Therefore, the map is only valid during a limited timeframe.

Similar to previous works [9–11], we use the hit ratio to compare our method with previous cache policies. Figure 3 presents the results of our proposed method in comparison to other proposed methods. This figure reports the average hit ratio among all vehicles in the simulation. As we can see, our method provides at least 15% higher average hitratio than other methods. This hold for both 60-seconds and120-seconds lifetimes.

1 https://omnetpp.org
2 https://eclipse.dev/sumo/
3 https://www.openstreetmap.org

4 https://veins.car2x.org

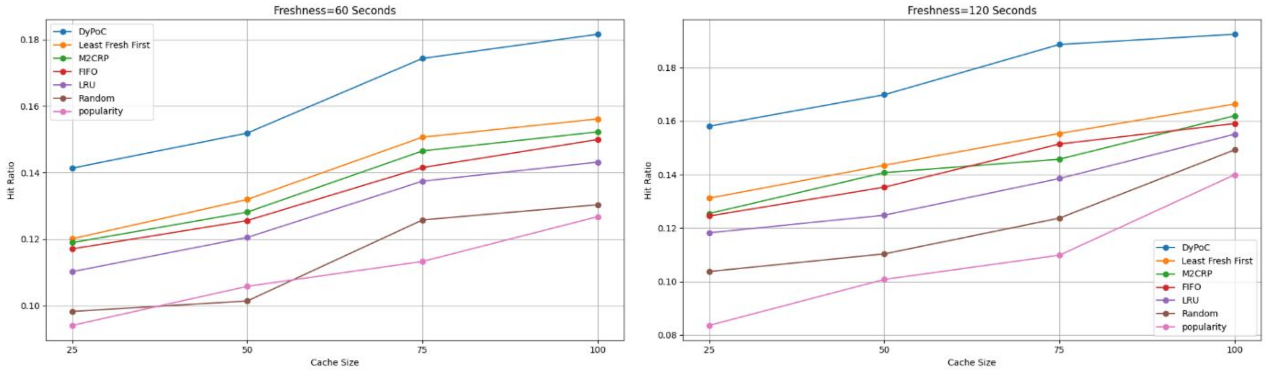

Figure 3 *Measurement of cache hit ratio with respect to different cache sizes.*

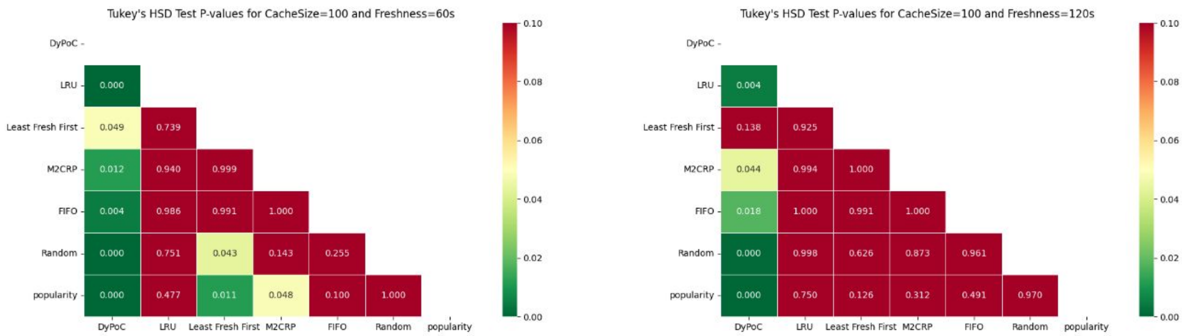

Figure 4 *Post-hoc Tukey's HSD Test.*

We also conducted a one-way ANOVA on the hit ratio of individual vehicles between different caching policies to see if the difference that we see is statistically significance. To do this this, we first conducted the Levene's test to assess the equality of variances. The p-value of this test was higher than 0.05 for all cases (different cache sizes and scenarios), which enables us to run the ANOVA.

The ANOVA test shows a statically significant difference for all cache sizes and scenarios (p-value < 0.05). To see which cache policy results are significantly different than others, we used the Tukey's Honestly Significant Difference (HSD) Test. This provides a more detailed understanding of the performance differences between the caching policies. We conducted these tests for different cache sizes and different map lifetimes. In almost all of them, our proposed method was significantly better than most of the other methods. Two examples of post-hoc Tukey's test for different scenarios are shown in Figure 4. Figure 4 shows that our method is significantly better almost all other methods.

Limitation & Future Work

Our work has not considered the privacy implications tied to the transmission maps that a vehicle needs. Broadcasting this information has the potential to unveil the destination and route that the vehicle will travel. This creates a substantial privacy concern. Future work can focus on how we can preserve the privacy while still requesting the necessary maps.

Another limitation of our simulation is the properties of maps. The current state of HD maps lacks standardization. Major companies such as Google and TomTom each provid their own HD map services featuring diverse formats, sizes, and included information. This lack of uniformity introduces a challenge for researchers, as precise information and map sizes are not publicly available and vary between services. Furthermore, this absence of standardization restricts vehicles from using information from specific services, limiting the exchange of information and collaborative efforts.

Conclusion

In this paper, we have introduced *DyPoC*, a specialized caching policy designed for VANETs that addresses traffic data characterized by expiration time and region-specific relevance. Our approach leverages critical parameters such as data freshness, vehicle path, and proximity to the region of interest to manage caching decisions intelligently. We evaluated the effectiveness of our policy in the context of HD map transmission, a crucial category of traffic information. Our simulation results demonstrate that our caching policy achieves at least 15% improvement in the mean hit ratio compared to alternative policies. Further statistical tests confirms that these results are significantly better than previous methods.

References

[1] S.-U. Rehman, M. Khan, T. Zia, and L. Zheng, “Vehicular ad-hoc net- works (vanets): an overview and challenges,” *Journal of Wireless Net-working and communications*, vol. 3, no. 3, pp. 29–38, 2013.

[2] P. Belanovic, D. Valerio, A. Paier, T. Zemen, F. Ricciato, and C. F. Meck- lenbrauker, “On wireless links for vehicle-to-infrastructure communi- cations,” IEEE Transactions on vehicular technology, vol. 59, no. 1, pp. 269–282, 2009.

[3] C. Chen, C. Wang, T. Qiu, M. Atiquzzaman, and D. O. Wu, “Caching in vehicular named data networking: Architecture, schemes and future directions,” IEEE Communications Surveys & Tutorials, vol. 22, no. 4, pp. 2378–2407, 2020.

[4] L. Jin, C. Zhu, L. Yu, and B. Zhang, “An aae-based proactive content chunked caching scheme with mobility prediction in vanets,” in Proceedings of the 2023 4th International Conference on Computing, Networks and Internet of Things. Xiamen, China: ACM, 2023, pp. 259–264.

[5] L. Zhao, H. Li, E. Zhang, A. Hawbani, M. Lin, S. Wan, and M. Guizani, “Intelligent caching for vehicular dew computing in poor network con- nectivity environments,” ACM Transactions on Embedded Computing Systems, vol. 23, no. 2, pp. 1–24, 2024.

[6] G. Elghazaly, R. Frank, S. Harvey, and S. Safko, “High-definition maps: Comprehensive survey, challenges and future perspectives,” IEEE Open Journal of Intelligent Transportation Systems, vol. 4, pp. 527–550, 2023

[7] S. Vural, N. Wang, P. Navaratnam, and R. Tafazolli, “Caching transient data in internet content routers,” IEEE/ACM Transactions on Network- ing, vol. 25, no. 2, pp. 1048–1061, 2016.

[8] Z. Zhang, C.-H. Lung, I. Lambadaris, and M. St-Hilaire, “Iot data lifetime-based cooperative caching scheme for icn-iot networks,” in 2018 IEEE International Conference on Communications (ICC). Kansas City, MO, USA: IEEE, 2018, pp. 1–7.

[9] M. Amadeo, C. Campolo, G. Ruggeri, G. Lia, and A. Molinaro, “Caching transient contents in vehicular named data networking: A performance analysis,” Sensors, vol. 20, no. 7, p. 1985, 2020.

[10] M. Amadeo, G. Ruggeri, C. Campolo, and A. Molinaro, “Diversity- improved caching of popular transient contents in vehicular named data networking,” Computer Networks, vol. 184, p. 107625, 2021.

[11] S. Ostrovskaya, O. Surnin, R. Hussain, S. H. Bouk, J. Lee, N. Mehran, S. H. Ahmed, and A. Benslimane, “Towards multi-metric cache replace- ment policies in vehicular named data networks,” in 2018 IEEE 29th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC). Bologna, Italy: IEEE, 2018, pp. 1–7.

[12] M. Meddeb, A. Dhraief, A. Belghith, T. Monteil, K. Drira, and H. Math-kour, “Least fresh first cache replacement policy for ndn-based iot networks,” Pervasive and Mobile Computing, vol. 52, pp. 60–70, 2019.

[13] S. Zhang, W. Quan, J. Li, W. Shi, P. Yang, and X. Shen, “Air-ground in- tegrated vehicular network slicing with content pushing and caching,” IEEE Journal on Selected Areas in Communications, vol. 36, no. 9, pp. 2114–2127, 2018.

[14] Y. Gu, J. Liu, and L. Zhao, “Low-latency transmission and caching of high definition map at a crossroad,” in Communications and Networking: 14th EAI International Conference (ChinaCom 2019). Shanhai, China: Springer, 2020, pp. 264–277.

[15] J. Mittag, F. Thomas, J. Härri, and H. Hartenstein, “A comparison of single-and multi-hop beaconing in vanets,” in Proceedings of the sixth ACM international workshop on VehiculAr InterNETworking. Beijing, China: ACM, 2009, pp. 69–78.