Efficient real-time coding-assisted heterogeneous data access in vehicular networks

Ali, G. G. Md. Nawaz; Noor-A-Rahim, Md.; Rahman, Md. Ashiqur; Samantha, Syeda Khairunnesa; Chong, Peter Han Joo; Guan, Yong Liang

2018

Ali, G. G. M. N., Noor-A-Rahim, M., Rahman, M. A., Samantha, S. K., Chong, P. H. J., & Guan, Y. L. Efficient real-time coding-assisted heterogeneous data access in vehicular networks. IEEE Internet of Things Journal, 1-1. doi:10.1109/JIOT.2018.2830315

<https://hdl.handle.net/10356/89632>

<https://doi.org/10.1109/JIOT.2018.2830315>

© 2018 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works. The published version is available at: [<http://dx.doi.org/10.1109/JIOT.2018.2830315>].

Downloaded on 26 Apr 2023 19:51:18 SGT

An Efficient Real-time Coding-assisted Heterogeneous Data Access in Vehicular Networks

G. G. M. Nawaz Ali*, Md. Noor-A-Rahim*, Md. A. Rahman[†], S. K. Samantha[‡], Peter H.J. Chong[§], and Y. L. Guan*

*School of Electrical & Electronic Engineering, Nanyang Technological University, Singapore [†]Department of Computer Science, University of Arizona, Tucson, Arizona, USA [‡]Department of Computer Science, Iowa State University, Ames, Iowa, USA [§]Department of Electrical and Electronic Engineering, Auckland University of Technology, New Zealand Email:{nawaz.ali, narahim, eylguan}@ntu.edu.sg; marahman@cs.arizona.edu; sammy@iastate.edu; peter.chong@aut.ac.nz

Abstract—Recently, much attentions have been paid to network coding-assisted data broadcast in vehicular networks. However, majority of the works consider all the accessed data items are the same size. In this work, we have studied the network coding-assisted heterogeneous on-demand data access in vehicular networks. Firstly, we have investigated the less efficiency of conventional coding assisted approach in accessing heterogeneous data items in real-time vehicular environment. Due to ignoring the impact of heterogeneous data items in decoding, the conventional coding does not achieve expected performance in accessing data items with diverse size. Secondly, based on our observations, we have proposed a new algorithm. The proposed network coding assisted approach exploits the different MCSs (Modulation and coding scheme) of IEEE 802.11p physical layer for leveraging the variable serving rate considering the dynamic positions of vehicles along with the vehicle mobility. Thirdly, we have built a vehicular simulation environment and evaluated the performance of the proposed approach along with competitive no-coding and coding-assisted approaches under various circumstances. The results show that the proposed approach outperforms the state-of-the-art approaches in terms of improving the on-demand requests serving capability and reducing the data access time.

I. INTRODUCTION

Vehicular networks is a popular and promising platform for road safety, collision/accident avoidance, enhanced and comfort driving, value-added and infotainment services etc. in the Intelligent Transportation System (ITS) [1]–[3]. DSRC (Dedicated Short Range Communication) is an exceptional protocol being standardized for supporting both V2V (Vehicle-to-vehicle) and V2I (vehicle-to-infrastructure) communication. The US FCC (Federal Communications Commission) has allocated 75 MHz licensed spectrum for the DSRC communication in 5.9 GHz frequency band [4]. In fact, DSRC refers to a family of standards including IEEE 802.11p amendment for the WAVE (Wireless Access in Vehicular Environments), the IEEE 1609.1~0.4 standards for, respectively, resource management, security, network service, and multi-channel operation, the SAE J2735 message set dictionary and the emerging J2945.1 communication minimum performance requirement [5]. However, despite the recent advancement of wireless communication technologies, the successful deployment of vehicular networks based applications relies on the time and bandwidth efficient data dissemination due to the intrinsic characteristics of vehicular networks, such as, high vehicle

mobility, communication range constraint, dynamic vehicular traffic, intermittent V2V connection, sparse RSU deployment, and diverse application requirements etc. In this paper, we are focusing on the efficient heterogeneous data dissemination in V2I communication.

In DSRC, V2V communication is advised mainly for the safety critical applications, such as, collision avoidance warning, emergency electronic brake warning lane changing warning etc.; whereas V2I communication can be for both safety critical and not-safety services, such as, collision avoidance warning, disseminating updated traffic info, finding a nearby parking lot, getting infotainment data or accessing in-vehicle internet etc. [3], [6]. In V2I communication, other than receiving push based safety-critical data, different vehicles also might be interested in receiving different on-demand non-safety data [1]. On-demand broadcast is a scalable approach for disseminating information for an environment with dynamic needs [7], [8]. A number of recent works have shown interest in network coding based on-demand broadcast, due to the capability of network coding of serving multiple data items in a single broadcast [1], [9], [10]. Note that bitwise exclusive-or ($\oplus$) is used for coding and decoding operations based on the client-side cache information. However, majority of these works consider all the accessed data items are the equal size, which in fact is not true. For instance, traffic update data and infotainment data are not the equal size. Again note that the size of the encoded packet is the size of the maximum size data items among all the data items that are being encoded in the coded packet [11]. Hence, combining diverse size data items together increases the access time of the smaller size data item in decoding, which eventually leads to the increased deadline misses of the on-demand requests.

Recently, we have taken the lead for studying the coding-assisted heterogeneous data access [12]. Initially, we devise a dynamic THRESHOLD based coding for minimizing the system response time. Note that *stretch* is a widely used metric for measuring system responsiveness with heterogeneous data item size [13]. *Stretch is defined as the ratio of response time to the service time, where service time is the transmission time of the data item.* The proposed dynamic THRESHOLD guides in encoding for ensuring that none of the coded data items increases the stretch of the scheduled urgent on-demand

request; which is the key of keeping the overall system stretch lower as well as satisfying more requests per broadcast. We derive an analytical study on the probability of a requested data item to be in an encoded packet. Then we provide an analytical analysis on the probability of deadline miss of a submitted request in each of the studied approaches.

However, later, we will show that performance can be further improved with the integration of variable data rate of the IEEE 802.11p MCS (Modulation and Coding Scheme) and dynamic THRESHOLD. According to the IEEE 802.11p PHY layer specification, different MCSs provides different data rates with various communication ranges [5]. Note that higher data rate corresponds to the lower range [14]. Intuitively, in a given instance, different vehicles' positions are different inside an RSU communication range. Hence, the distance of a vehicle from the RSU is different from one vehicle to another vehicle. Therefore, instead of fixed data rate, an RSU can exploit the distances of vehicles in encoding for serving closer vehicles with higher data rate. Note that the integration of dynamic THRESHOLD in encoding, controls the mixing of too small and too big data items in the same coded packet. Lastly, in vehicular networks the on-demand requests need to be served on time, otherwise, the response becomes useless. For instance, if a vehicle left an RSU at a junction point before receiving the current updated traffic information, the driver might miss taking the optimal rerouting decision onboard. Hence, the submitted request should be served within the dwelling duration of a vehicle inside the RSU range. In summary, our contributions in this paper are as follows.

- Firstly, we have investigated the reason of less efficiency of the conventional network coding based broadcast for serving on-demand requests in accessing heterogeneous data items.
- Secondly, we have proposed a network coding based on-demand broadcast algorithm for accessing heterogeneous data items called ISXD_Θ (Inverse of Slack time multiply Distance with THRESHOLD (Θ)). ISXD_Θ uses slack time of request and distance of vehicle from RSU for scheduling data item aiming to use higher MCS data rate for serving more on-demand requests within the deadlines. The integration of a dynamic THRESHOLD (Θ) in encoding, reduces the decoding delay.
- Thirdly, we have built a vehicular simulation model and evaluated the performance of the proposed algorithm along with other competing alternative solutions in the literature to demonstrate the efficacy of the proposed solution under different circumstances.

The rest of the paper is organized as follows. Section II describes the system architecture, notations, assumptions and initializations. Section III demonstrates the motivation of this research work with a running example. Section IV outlines our proposed network coding-based on-demand broadcast scheme. Section VI describes the simulation setup and performance analysis. Finally, we conclude this paper in Section VII.

Fig. 1. System architecture.

TABLE I
DIFFERENT MCS AND THEIR CORRESPONDING DATA RATE.

MCS (Modulation and Coding Scheme)	Data rate (Mbps)	Communication range	Normalized data rate ($\frac{\text{unit}}{\text{tick}}$)
R12BPSK	3	1000	0.5
R34BPSK	4.5	900	0.75
R12QPSK	6	800	1.0
R34QPSK	9	700	1.5
R12QAM16	12	600	2.0
R34QAM16	18	500	3.0
R23QAM64	24	400	4.0
R34QAM64	27	300	4.5

II. SYSTEM MODEL

A. System Architecture

Similar to [9], we assume that a single radio is operating in V2I communication. According to IEEE 1609.4 standard, a vehicle can switch between a control channel (CCH) and one of the six service channels (SCH) [5]. The time is divided into 100 msec synchronization interval, which consists of 50 msec CCH and 50 msec SCH interval. Safety message is periodically (usually at 10Hz) disseminated through the CCH. RSU also announces the upcoming service index through the CCH, so that vehicles can tune to the SCH if they are interested in the service. A typical coding based on-demand broadcast architecture in vehicular network is shown in Fig. 1. Usually safety-messages are broadcast periodically and consume less bandwidth [1], hence we only consider on-demand requests for non-safety messages for scheduling and encoded broadcasting through SCH.

If a data item is not found in the cache, a vehicle issues on-demand request to the RSU through the CCH. A vehicle also inserts the index of its local cache information into the submitted request. The pending requests are queued in the RSU service queue. The RSU invokes the underlying scheduling algorithm for finding the scheduled request and then based on the requested data items in the service queue and the cached data items of vehicles, the network coded

TABLE II
SUMMARY OF NOTATIONS.

Symbol	Description
R_i	Request i
d_j	Data item j
$S(d_j)$	Size of d_j
Vh_i	A vehicle
RSU_i	An RSU
$T_{R_i}^{gnr}$	The time R_i is generated
$T_{R_i}^{deadline}$	Deadline of R_i
V	Speed of a vehicle
S_i	Slack time of R_i
D_i	Distance of Vh_i from RSU
v_{ij}	A vertex represents R_i requesting d_j
$Cache(Vh_i)$	Cached data items of Vh_i
$\mathbb{C}$	A clique
γ	An encoded packet
BW	Service channel bandwidth

packet will be formed. The low-overhead bitwise XOR ($\oplus$) operation is used for the encoding and decoding operations [1]. Accordingly, a vehicle can decode the received encoded packet if except the requested data item all other data items in the encoded packets are present in its local cache. For instance, in Fig. 1, v_1 and v_9 cached d_1 and d_9 , respectively, and issued requests for respective data items d_9 and d_1 . The server broadcasts encoded packet $P = d_1 \oplus d_9$. Then v_1 can decode d_9 by $P \oplus d_1 = d_9$, similarly, v_9 can decode d_1 . The decoded data item will be added in the vehicle's local cache. The LRU (Least Recently Used) replacement policy is adopted for cache replacement. It is a closed system model [10], where a vehicle only issues a new request after previous submitted request is served or misses its deadline. Multiple RSUs are installed in the system and vehicles mobility are controlled by the Manhattan mobility model [15].

According to the IEEE 802.11p PHY layer specification, the data rate of MCS varies from 3 Mbps to 27 Mbps, whereas the ranges varies from 1000 m to 300 m [5], [14]. The relation between data rate and communication range is shown in Table I.

B. Notations, assumptions and initializations

We assume that N number of outstanding requests are in the service queue. M is the number of available data items in the RSU database. d_j denotes the j^{th} data item in the database, where $1 \leq j \leq M$. $S(d_{min})$ and $S(d_{max})$ represent the minimum and maximum size data items, respectively in the database. The used primary notations are summarized in Table II.

Request: A vehicle Vh_i submits a request R_i at an RSU with the following tuple,

$$R_i = (Vh_i, d_j, T_{R_i}^{gnr}, T_{R_i}^{deadline}, Cache(Vh_i))$$

Here, Vh_i : The vehicle ID; d_j : Data item requested by R_i ; $T_{R_i}^{gnr}$: The generation time of R_i . $T_{R_i}^{gnr}$ follows the Uniform distribution, namely, $T_{R_i}^{gnr} = \text{random}(0.0, \{T_{max}^{dwell}(Vh_i) - T_{R_i}^{serv}\})$. Note that $T_{max}^{dwell}(Vh_i) = \frac{2R}{V}$, is the maximum possible dwelling time of a vehicle inside an RSU range, where R denotes the RSU range, and V denotes the driving speed. $T_{R_i}^{serv}$ is the required serving time of R_i , namely, $\frac{S(d_j)}{BW}$, where $S(d_j)$ is the size of requested data item d_j and BW is the service channel bandwidth;

$T_{R_i}^{deadline}$: The assigning deadline of R_i , which is calculated as $T_{R_i}^{deadline} = T_{max}^{dwell}(Vh_i) - T_{R_i}^{serv} - T_{R_i}^{gnr}$. After expiring $T_{R_i}^{deadline}$, R_i will be discarded from the service queue;

$Cache(Vh_i)$: The index of current cached data items of Vh_i ;

Network coded packet: For forming the server side encoding, the RSU needs the cache information of vehicles. A CR (Cache Request)-graph $G(V, E)$ builds a relationship among the pending requests and their cached data items [10], [12]. In $G(V, E)$, each vertex represents a request with its requested data item. For instance, v_{ij} represents request R_i pending for data item d_j , where $1 \leq i \leq N$; $1 \leq j \leq M$.

The edge formation follows the following two rules. For any two vertices $\{v_{ij}, v_{mn}\} \in V(G)$, there will be an edge $e(v_{ij}, v_{mn})$, if

- $j = n$, which means both vehicles request for the same data item, hence the same broadcast can satisfy both of the requests.
- $j \neq n$, but $d_j \in Cache(Vh_m)$ and $d_n \in Cache(Vh_i)$, which means if $d_j \oplus d_n$ is broadcast, both Vh_i and Vh_m can decode their respective requested data items d_j and d_n .

Note that a clique $\mathbb{C}$ is a subset of vertices in $V(G)$, where every two vertices are connected by an edge. For instance, $\mathbb{C} = \{v_{i_1, j_1}, v_{i_2, j_2}, \dots, v_{i_k, j_k}\}$ is an arbitrary clique in G , where $\mathbb{C} \subseteq V(G)$. The set of data items in $\mathbb{C}$ are $D_{\mathbb{C}} = \{d_{\mathbb{C}(1)}, d_{\mathbb{C}(2)}, \dots, d_{\mathbb{C}(|D_{\mathbb{C}|})}\}$, where $1 \leq |D_{\mathbb{C}}| \leq |\mathbb{C}|$. Denote γ as encoded packet. For $\mathbb{C}$, γ can be defined as $\gamma = d_{\mathbb{C}(1)} \oplus d_{\mathbb{C}(2)} \oplus \dots \oplus d_{\mathbb{C}(|D_{\mathbb{C}|})}$.

Since each vertex represents an unique vehicle, satisfying the maximum number of vehicles per broadcast is equivalent to finding the maximum clique $\mathbb{C}_{max}$ in $G(V, E)$. Nevertheless, finding a maximum clique is a well-known NP-hard problem in graph theory. Therefore, we propose a coding based greedy algorithm for accessing heterogeneous data items with polynomial time complexity.

Theorem 1. *Optimal encoding for accessing heterogeneous data is NP-hard*

Proof. Note that heterogeneous data item sizes in the RSU database follows the random distribution within $[S(d_{min}), S(d_{max})]$ [16]. Consider a special case, $S(d_{min}) = S(d_{max})$. Again, according to the closed system model, a vehicle submits a new request only previous submitted request is served or expired [10], which means each pending request in the service queue represents an unique vehicle, namely, each vertex in graph G represents an unique vehicle. Therefore, broadcasting the encoded packet corresponds to the maximum clique will satisfies the maximum number of vehicles per broadcast. Again, as all the vertices corresponds to the data items with equal size, the encoded packet forming by the data items corresponds to the maximum clique will not increase the accessing delay in decoding. However, finding a maximum clique in a graph G is a well-known NP-complete problem in graph theory; hence finding optimal broadcast with coding for heterogeneous data access, which is the general case, is NP-hard. $\square$

Fig. 2. Suggested coding technique with THRESHOLD.

III. OBSERVED PROBLEMS AND MOTIVATION

The conventional on-demand coding [11] forms the encoded packet with the data items corresponds to the maximum clique (C_{max}) which covers the scheduled request. However, note that when diverse sized data items are encoded together, the size of the encoded packet is the size of the maximum sized data item among all the data items that are being encoded. From our earlier study [12] and experimental results we have observed that the conventional on-demand coding cannot retain its superiority when data item size varies much. The results are presented and analyzed in Section VI. Firstly, we observe that encoding bigger and smaller size data items together,

- Penalizes smaller size data item with higher access time and higher stretch, where stretch is the ratio of the response time to the service time.
- Requests with lower remaining deadlines may miss their deadlines.

However, network coding gain is higher only when more data items are encoded per broadcast. Hence need to find a balanced solution. Therefore, our intuition is to encode as many data item as possible in a coded packet with the following two guidelines.

- The urgent request should meet its deadline.
- The data item size difference between pending data item of urgent request and other encoded data items should be shaped by a dynamic THRESHOLD. The data item size difference should not be beyond the THRESHOLD value, whereas THRESHOLD will be updated according to the heterogeneity of the pending data items in the service queue.

Note that the first guideline ensures that more requests will meet their deadlines; and second guideline controls that too big and too small data items will not be comprised into the same encoded packet. The suggested coding technique with a THRESHOLD is shown in Fig. 2.

Secondly, according to IEEE 802.11p MCS, the data rate is inversely proportional to the RSU serving range [14]. Note that different vehicles' distance from an RSU at any given time is different, hence available higher data rate can be exploited in favor of serving heterogeneous data items for a group of vehicles. At any given time, a group of vehicles closer to the RSU might be served with a higher data rate, which can reduce the access time, stretch and deadline miss ratio.

To sum up, on the one hand, request deadline is a crucial metric to consider for satisfying requests in time, and on the other hand, vehicle distance from the RSU is the key for higher serving rate. Hence, we consider *slack time (remaining deadline)* and *distance from RSU* for scheduling a request. Then for

(a) Pending requests and their attributes.

(b) Conventional coding.

(c) Proposed coding.

Fig. 3. Conventional vs. proposed on-demand coding.

forming the encoded packet we use a dynamic THRESHOLD (denoted by Θ) along with possible higher MCS data rate. The integrated proposed approach is called $ISXD_{\Theta}$ (*Inverse of Slack time multiply Distance with THRESHOLD*).

A. A running example

A running example in Fig. 3, compares conventional coding and the proposed coding. According to the EDF (Earliest Deadline First) scheduling [17], which is the foremost real-time scheduling algorithm, the requests are served according to their deadlines order. Hence, the corresponding data items serving order is $\{d_1, d_2, d_3, d_5, d_4\}$. The stretch is $\frac{15}{15} + \frac{35}{20} + \frac{60}{25} + \frac{110}{50} + \frac{175}{65} = 10.04$. Total five broadcast ticks are required for serving five vehicles. Vehicles Vh_4 and Vh_5 will miss their deadlines.

According to the conventional coding [11], the first serving encoded packet comprises the data items of the maximum clique containing the vertex of the urgent vehicle. Here $C_{max} = \{Vh_1, Vh_2, Vh_3, Vh_4\}$, and urgent vehicle is Vh_1 . Hence, the first encoded packet is $\gamma_1 = d_1 \oplus d_2 \oplus d_3 \oplus d_4$. Next it will serve $\gamma_2 = d_5$. According to the data item size attribute, the access time is $65(\gamma_1) + 50(\gamma_2) = 115$, and the stretch is $\frac{65}{15} + \frac{65}{20} + \frac{65}{25} + \frac{65}{65} + \frac{115}{50} = 13.48$. According to the deadline attribute, two requests will miss their deadlines; while γ_1 is served, $T_1 (Vh_1)$ will miss its deadline, and while γ_2 is serving $T_5 (Vh_5)$ will miss its deadline.

In the EDF_H [12], as the scheduling is done based on the deadline of request (earliest deadline first), the serving order is $\gamma_1 = d_1 \oplus d_2 \oplus d_3 \oplus d_4$ and then $\gamma_2 = d_5$. Access time is $25(\gamma_1) + 65(\gamma_2) = 90$, and the stretch is $\frac{25}{15} + \frac{25}{20} + \frac{25}{25} + \frac{90}{65} + \frac{90}{50} = 7.1$. Surely, the response time is better than the conventional coding. However, $T_5 (Vh_5)$ will miss its deadline, while γ_2 is serving.

In the proposed $ISXD_{\Theta}$, scheduling is done based on the request deadline and vehicle's distance from RSU. In encoding, size of pending data item is considered. Accordingly, serving order is $\gamma_1 = d_1 \oplus d_2 \oplus d_3$ and then $\gamma_2 = d_4 \oplus d_5$. However, based on the distance of the vehicles from RSU

in Fig. 3a, and normalized data rate in Table I, γ_1 and γ_2 will be served with data rate 1.0 and 4.0, respectively. Hence the access time will be $\frac{25}{15}(\gamma_1) + \frac{65}{40}(\gamma_2) = 41.25$, and the stretch is $\frac{25}{15} + \frac{25}{20} + \frac{25}{25} + \frac{41.25}{65} + \frac{41.25}{50} = 5.38$. Hence, clearly both the access time and stretch have been further reduced from conventional coding and EDF_H. Now if we consider request deadlines, all the requests will meet their deadlines, hence there will be no deadline missed request by the proposed approach ISXD $_{\Theta}$ in this example.

IV. PROPOSED CODING BASED DATA BROADCAST ALGORITHM ISXD $_{\Theta}$

In vehicular network, the request needs to be served within the stipulated deadline. Hence deadline should be considered in request scheduling.

Definition 1. Slack Time ($\mathbb{S}$): is the remaining deadline of a request. $\mathbb{S}_i$ denotes the slack time of request R_i . For instance, at time t , $\mathbb{S}_i$ of R_i is $T_{R_i}^{deadline} - t$.

Next, distance of a vehicle from RSU is an important metric to consider in request scheduling. Note that bandwidth (data rate) is inversely proportional to the distance (Table I).

Definition 2. Distance from an RSU ($\mathbb{D}$): $\mathbb{D}_i$ denotes the distance of vehicle Vh_i from an RSU within the range $\mathbb{R}$. $\mathbb{D}_i$ at time t can be calculated as $\mathbb{D}_i(t) = |\mathbb{R} - (T^{dwell}(Vh_i) \times V)|$, where $T^{dwell}(Vh_i)$ is the dwelling duration of Vh_i inside the RSU, and V is the driving speed.

Definition 3. Priority of a request ($\mathbb{P}$): Priority of a request R_i , $\mathbb{P}_i$ is the inverse of multiplication of $\mathbb{S}_i$ and $\mathbb{D}_i$. $\mathbb{P}_i$ of R_i at time t can be calculated as,

$$\mathbb{P}_i = \frac{1}{\mathbb{S}_i(t) \times \mathbb{D}_i(t)}$$

The request with the maximum priority will be chosen as a candidate vertex v_{ck} for scheduling process at the RSU server.

Priority $\mathbb{P}$ is calculated based on the following two intuitive observations:

- Given two requests with the same distance-from-RSU, the one with the smaller slack time should be chosen to serve. This helps to meet the deadlines of requests.
- Given two requests with the same slack time, the one with the smaller distance-from-RSU should be chosen to serve. This helps the RSU to use the MCS with higher bandwidth (lower range), hence access time will be reduced.

After identifying the candidate vertex, the encoded packet formation is performed based on the distances' of the vehicles from the RSU, available serving rate, slack time of the candidate request, and the dynamic THRESHOLD (Θ). The pseudocode of the proposed approach is shown in Algorithm 1. The key steps are stated as below.

Step 1: Initialize Θ . Upon arrival of a request, add the request in the service queue and update the pending data item list.
Step 2: Find the most priority request according to Definition 3. The vertex corresponds to this request will be assigned as the candidate vertex.

Step 3: Find the sub-graph in which all the vertices are connected to the candidate vertex. Find the maximal clique of the candidate vertex from the sub-graph.

Step 4: Find the maximum possible *Data rate* for serving the candidate vertex based on the distance on the candidate vertex from the RSU and by invoking Table I.

Step 5: Find the vertices to encode from the maximal clique by excluding vertices, which requires higher service time with the identified *Data rate* (identified in Step 4) than the slack time of the candidate vertex and data item size difference from the candidate vertex is higher than Θ . Update Θ .

Step 6: Form the encoded packet and broadcast it with the maximum possible *Data rate* for the encodable vertices. Update service queue.

For n number of outstanding requests in the RSU service queue, Steps 1, 2, 4, 5, and 6 each requires maximum $\mathcal{O}(n)$ complexity. Step 3 requires to search the maximal clique for a specific candidate vertex which requires $\mathcal{O}(n^3)$ complexity. Hence, the overall complexity of the proposed approach is $\mathcal{O}(n^3)$, which is practical to implement.

Refer to Algorithm 1 (Lines 38~45), a temporary buffer **TEMP** is used to hold the extracted vertices from $\mathbb{C}_{max}^{v_{ck}}$, which satisfies the conditions as stated in Step 5 above. THRESHOLD (Θ) update is done in Lines 47~50. Θ_{next} is updated by the current THRESHOLD (Θ_{curr}), previous THRESHOLD (Θ_{old}), current service queue size ($|V(G)|$) and current encoded packet size ($|\mathbf{TEMP}|$). A higher $\frac{|\mathbf{TEMP}|}{|V(G)|}$ indicates that Θ can be bigger and hence more vehicles can be satisfied by serving the bigger encoded packet (γ). In contrast, a lower $\frac{|\mathbf{TEMP}|}{|V(G)|}$ value indicates that the data item size difference is much, so Θ should be smaller which directs to the smaller encoded packet.

V. ANALYTICAL ANALYSIS

We use following notations:

- S_h highest limit of data size
- S_l lowest limit of data size
- $S_{min} = BW * T_{min}$.
- $S_q = S(d_q)$

Lemma: Let v_j is a vertex that is included in the maximal clique which covers v_{cq} . We assume that data size S_j of an arbitrary message d_j follows a truncated Gaussian distribution with mean μ , variance σ^2 , and lies within the interval $S_j \in \{S_l, S_h\}$. The probability that a message d_j corresponds to v_j will not be included in the encoded message is given in the top of the next page (see eq. (1)). In (1), $\Gamma(x)$ and $erf(x)$ are defined by,

$$\Gamma(x) = \frac{1}{2} \left\{ 1 + erf \left(\frac{x - \mu}{\sigma \sqrt{2}} \right) \right\},$$

$$erf(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt,$$

Proof: The probability that a message d_j will not be included in the encoded message γ is

$$\Pr(d_j \text{ will not be in } \gamma) = 1 - \Pr(d_j \text{ will be in } \gamma) \quad (2)$$

$$\Pr(d_j \text{ will not be in } \gamma) = 1 - \frac{1}{\Gamma(S_h) - \Gamma(S_l)} \left[\left\{ \Gamma(\min(S_{min}, S_{next} - S_q)) - \Gamma(S_l) \right\} \times \left\{ 1 - \Gamma(S_q) + \Gamma(S_l) \right\} + \left\{ \Gamma(S_{min}) - \Gamma(S_q - S_{next}) \right\} \left\{ \Gamma(S_q) - \Gamma(S_l) \right\} \right] \quad (1)$$

Now $\Pr(d_j \text{ will be in } \gamma)$ can be written as

$$\Pr(d_j \text{ will be in } \gamma) = \Pr(S_j < \min(S_{min}, S_{next} - S_q)) \times (1 - \Pr(S_j < S_q)) + \Pr(S_q - S_{next} \leq S_j \leq S_{min}) \Pr(S_j < S_q) \quad (3)$$

The first line of the above equation corresponds to the case when $S_j > S_q$. For this case, to be a part of γ , S_j needs to satisfy conditions $S_j < S_{min}$ and $S_j < S_{next} - S_q$. Thus, we need to find $\Pr(S_j < \min(S_{min}, S_{next} - S_q))$. The second line of (3) represent the case when $S_j < S_q$ and S_j needs to satisfy $S_q - S_{next} \leq S_j \leq S_{min}$ to be a part of γ . Thus, we need to find $\Pr(S_q - S_{next} \leq S_j \leq S_{min})$. Recall that the size of data follows following truncated Gaussian distribution with mean μ , variance σ^2 , and $S_j \in \{S_l, S_h\}$

$$f(S_j|\mu, \sigma, S_l, S_h) = \frac{\frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(S_j-\mu)^2}{2\sigma^2}\right)}{\Gamma(S_h) - \Gamma(S_l)} \quad (4)$$

where the numerator is the standard Gaussian distribution, which can be represented by $f(S_j|\mu, \sigma)$. Thus, the probability that S_j is below than a given value x is,

$$\begin{aligned} \Pr(S_j < x) &= \int_{S_l}^x f(S_j|\mu, \sigma, S_l, S_h) dS_j \\ &= \frac{1}{\Gamma(S_h) - \Gamma(S_l)} \int_{S_l}^x f(S_j|\mu, \sigma) dS_j \\ &= \frac{1}{\Gamma(S_h) - \Gamma(S_l)} \left[\int_{-\infty}^x f(S_j|\mu, \sigma) dS_j - \int_{-\infty}^{S_l} f(S_j|\mu, \sigma) dS_j \right] \end{aligned} \quad (5)$$

Note that $\int_{-\infty}^x f(S_j|\mu, \sigma) dS_j$ represent cumulative distribution function (CDF) of standard Gaussian distribution. It is well known that CDF of Gaussian distribution can be derived in terms error function erf in the following manner:

$$\begin{aligned} \int_{-\infty}^x f(S_j|\mu, \sigma) dS_j &= \frac{1}{2} \left\{ 1 + erf\left(\frac{x-\mu}{\sigma\sqrt{2}}\right) \right\} \\ &= \Gamma(x) \end{aligned} \quad (6)$$

Replacing (6) in (5), we get

$$\Pr(S_j < x) = \frac{1}{\Gamma(S_h) - \Gamma(S_l)} [\Gamma(x) - \Gamma(S_l)] \quad (7)$$

Similarly, for two given values x_1 and x_2 , we can find $\Pr(x_1 \leq$

$S_j \leq x_2)$ in the following manner:

$$\begin{aligned} \Pr(x_1 \leq S_j \leq x_2) &= \int_{x_1}^{x_2} f(S_j|\mu, \sigma, S_l, S_h) dS_j \\ &= \frac{1}{\Gamma(S_h) - \Gamma(S_l)} \left[\int_{-\infty}^{x_2} f(S_j|\mu, \sigma) dS_j - \int_{-\infty}^{x_1} f(S_j|\mu, \sigma) dS_j \right] \\ &= \frac{1}{\Gamma(S_h) - \Gamma(S_l)} [\Gamma(x_2) - \Gamma(x_1)] \end{aligned} \quad (8)$$

First by using (7) and (8) in (3) and then replacing (3) in (2), we can prove the Lemma.

For analogizing DMR of all the approaches we assume the following notations:

- S_h highest limit of data size
- S_l lowest limit of data size

Lemma: Let the relative deadline of a packet requested from a vehicle follows an exponential distribution with mean μ , while the packet size follows truncated Gaussian distribution. With the presented three data dissemination strategies, the probability that a vehicle's requested data will not be served within its deadline (i.e., deadline miss ratio (DMR) of a vehicle's requested data) is given by:

- **With EDF:**

$$\begin{aligned} DMR_{EDF} &= \psi(\mathbb{C}_T) + \psi(\bar{\mathbb{C}}_T) \times \\ &\quad \left\{ 1 - \frac{1}{\Gamma(S_h) - \Gamma(S_l)} [\Gamma(T_{rem} * BW) - \Gamma(S_l)] \right\} \end{aligned} \quad (9)$$

- **With EDF_C:**

$$\begin{aligned} DMR_{EDF_C} &= \psi(\mathbb{C}_C) + \psi(\bar{\mathbb{C}}_C) \times \\ &\quad \left\{ 1 - \left(\frac{1}{\Gamma(S_h) - \Gamma(S_l)} [\Gamma(T_{rem} * BW) - \Gamma(S_l)] \right)^W \right\} \end{aligned} \quad (10)$$

- **With EDF_H:**

$$DMR_{EDF_H} = \psi(\mathbb{C}_H) \quad (11)$$

where $\psi(\mathbb{C}_i) = 1 - \frac{\mu}{\mathbb{C}_i} \left(1 - e^{-\frac{\mathbb{C}_i}{\mu}} \right)$ with $\mathbb{C}_i \in \{\mathbb{C}_T, \mathbb{C}_C, \mathbb{C}_H\}$; $\mathbb{C}_T, \mathbb{C}_C$, and $\mathbb{C}_H$ are the average interval duration of transmission of two consecutive requests of a vehicle with traditional, coded, and proposed broadcasting schemes; W is the average number of packets that constitute coded packet; T_{rem} is the remaining deadline at the start of the transmission of the requested packet.

Proof:

With EDF: In traditional broadcasting scheme, a packet can miss deadline for the following two cases: (i) the deadline of the requested packet expires before the start of the trans-

Algorithm 1: Proposed coding based broadcast algorithm ISXD_Θ.

```

1 Step 1: THRESHOLD (Θ) and service queue initialization
2 /* THRESHOLD (Θ) initialization */
3  $W_{\Theta_{old}} \leftarrow S(d_{min})$ ; /* Initialize  $W_{\Theta_{old}}$  to  $S(d_{min})$  */
4  $W_{\Theta_{curr}} \leftarrow S(d_{max})$ ; /* Initialize  $W_{\Theta_{curr}}$  to  $S(d_{max})$  */
5  $\Theta_{next} \leftarrow \frac{W_{\Theta_{old}} + W_{\Theta_{curr}}}{2}$ ; /* Calculate initial  $\Theta_{next}$  */
6  $V(G)$ : is the set of vertices in the service queue;
7 If a new request  $R_i$  pending for data item  $d_j$  is arrived, which
  corresponding vertex is  $v_{ij}$ ;
8 /* Add  $v_{ij}$  in the service queue */
9  $V(G) \leftarrow V(G) + v_{ij}$ ;
10 Step 2: Find the most priority request (vertex  $v_{ck}$ )
11  $MaxPriority \leftarrow 0$ ;
12 for each  $v_{mn} \in V(G)$  do
13   if  $\mathbb{P}_m > MaxPriority$  then
14      $MaxPriority \leftarrow \mathbb{P}_m$ ;
15      $v_{ck} \leftarrow v_{mn}$ ;
16 Step 3: Find the subgraph  $G'$  and maximal clique  $\mathbb{C}_{max}^{v_{ck}}$ 
17  $V'(G') \leftarrow v_{ck}$ ;
18 for each  $v_{ij} \in V(G)$  do
19   if  $j = k$  then
20     /* Both vehicles  $Vh_i$  and  $Vh_c$  request for
       the same data item  $d_k$  */
21      $V'(G') \leftarrow V'(G') + v_{ij}$ ;  $E'(G') \leftarrow E'(G') + e(v_{ij}, v_{ck})$ ;
22   else if  $d_j \in \text{Cache}(Vh_c)$  &  $d_k \in \text{Cache}(Vh_i)$  then
23     /* Cache of  $Vh_i$  contains the requested data
       item of  $Vh_c$ , and vice versa */
24      $V'(G') \leftarrow V'(G') + v_{ij}$ ;  $E'(G') \leftarrow E'(G') + e(v_{ij}, v_{ck})$ ;
25   else
26     /*  $v_{ij}$  will not be added in  $G'(V', E')$  */
27  $\mathbb{C}_{max}^{v_{ck}} \leftarrow$  Find the maximal clique of  $v_{ck}$  in  $G'$ ;
28 Step 4: Find the possible serving data rate for the candidate vertex
29  $\mathbb{S}_c \leftarrow$  Slack time of the candidate vehicle;
30  $\mathbb{D}_c \leftarrow$  Distance of candidate vehicle from the RSU;
31  $\mathbb{R}_c \leftarrow$  Serving RSU-range of the candidate vehicle based on  $\mathbb{D}_c$  and
  Table I;
32  $\mathbb{BW}_c \leftarrow$  Find the serving data rate of the candidate vehicle based on  $\mathbb{R}_c$ 
  and Table I;
33 Step 5: Find the vertices for encode and update  $\Theta$ 
34 /* If  $\Theta_{next}$  reaches to the boundary value, update
    $\Theta_{next}$  */
35 if  $\Theta_{next} \leq S(d_{min})$  ||  $\Theta_{next} \geq S(d_{max})$  then
36    $\Theta_{next} \leftarrow \frac{W_{\Theta_{old}} + W_{\Theta_{curr}}}{2}$ ;
37 /* Find vertices for encode */
38 Initialize  $\mathbf{Temp} \leftarrow \emptyset$ ;
39 /*  $\mathbf{Temp}$  is the temporary vertices holder */
40 for each  $v_{ij} \in \mathbb{C}_{max}^{v_{ck}}$  do
41   if  $|S(d_j) - S(d_k)| \leq \Theta_{next}$  then
42     /* If data item size difference is lower
       or equal to  $\Theta_{next}$  */
43     if  $\frac{S(d_j)}{\mathbb{BW}_c} \leq \mathbb{S}_c$  then
44       /* If serving time meets the slack time
       requirement */
45        $\mathbf{Temp} \leftarrow \mathbf{Temp} + \{v_{ij}\}$ ;
46  $\mathbb{BW} \leftarrow$  Find the max. possible serving data rate for the vertices in
   $\mathbf{Temp}$ ;
47 /* Update  $\Theta_{next}$ , for the next broadcast routine */
48  $W_{\Theta_{curr}} \leftarrow \frac{|\mathbf{Temp}|}{|V(G)|}$ ;
49  $\Theta_{next} \leftarrow \Theta_{next} + \frac{(W_{\Theta_{curr}} - W_{\Theta_{old}}) \times \Theta_{next}}{W_{\Theta_{old}}}$ ;
50  $W_{\Theta_{old}} \leftarrow W_{\Theta_{curr}}$ ;
51 Step 6: Broadcast the encoded packet and update service queue
52 /* Form the encode packet  $\gamma$  */
53  $\gamma = d_{\mathbf{Temp}(1)} \oplus d_{\mathbf{Temp}(2)} \oplus \dots \oplus d_{\mathbf{Temp}(|\mathbf{Temp}|)}$ ;
54 Broadcast  $\gamma$  with data rate  $\mathbb{BW}$ ;
55 for each  $v_{ij} \in \mathbf{Temp}$  do
56    $V(G) \leftarrow V(G) - \{v_{ij}\}$ ; /* Update  $V(G)$  */

```

mission. In other words, when the average interval of the consecutive transmission (i.e., $\mathbb{C}_T$) for the requestee vehicle is larger than the relative deadline. (ii) Even though the transmission starts before the deadline expires, the packet can still miss the deadline if the remaining deadline is smaller than the time required to transmit the packet. This case depends on the size of the requested/transmitted packet. Thus deadline miss ratio (DMR) can be written as:

$$DMR_{EDF} = \Pr(\text{The relative deadline} < \mathbb{C}_T) + (1 - \Pr(\text{The relative deadline} < \mathbb{C}_T)) \times \Pr(\text{Size of the requested packet} > T_{rem} * BW) \quad (12)$$

Following Xu et al. (2006), $\Pr(\text{The relative deadline} < \mathbb{C}_T)$ can be written as:

$$\Pr(\text{The relative deadline} < \mathbb{C}_T) = \frac{1}{\mathbb{C}_T} \int_0^{\mathbb{C}_T} f(\mathbb{C}_T - t) dt, \quad (13)$$

where $f(t)$ is the cumulative density function (cdf) of relative deadline. With exponential distribution (mean μ) of the relative deadline, i.e., $f(t) = 1 - e^{-\frac{t}{\mu}}$, we get

$$\begin{aligned} \Pr(\text{The relative deadline} < \mathbb{C}_T) &= \frac{1}{\mathbb{C}_T} \int_0^{\mathbb{C}_T} 1 - e^{-\frac{\mathbb{C}_T - t}{\mu}} dt, \\ &= 1 - \frac{\mu}{\mathbb{C}_T} \left(1 - e^{-\frac{\mathbb{C}_T}{\mu}} \right) \\ &= \psi(\mathbb{C}_T) \end{aligned} \quad (14)$$

On the other hand, the packet size follows truncated Gaussian distribution with mean μ , variance σ^2 , and $S_j \in \{S_l, S_h\}$. Thus, following the earlier Lemma, $\Pr(\text{Size of the requested packet} > T_{rem} * BW)$ can be written as

$$\begin{aligned} \Pr(\text{Size of the requested packet} > T_{rem} * BW) &= 1 - \Pr(\text{Size of the requested packet} < T_{rem} * BW) \\ &= 1 - \frac{1}{\Gamma(S_h) - \Gamma(S_l)} \left[\Gamma(T_{rem} * BW) - \Gamma(S_l) \right] \end{aligned} \quad (15)$$

Replacing (14) in (15), we can prove (9).

With EDF_C: Similar to the traditional broadcasting scheme, in coded broadcasting scheme, an event of deadline miss can occur for the following two cases: (i) when the average interval of the consecutive transmission (i.e., $\mathbb{C}_C$) for the requestee vehicle is larger than the relative deadline. Due to the encoding of multiple packets, it is obvious that $\mathbb{C}_C \leq \mathbb{C}_T$. (ii) When at least one of the packets, among the W packets which constitute encoded packet, has larger size than $T_{rem} * BW$. Thus deadline miss ratio (DMR) can be written as:

$$\begin{aligned} DMR_{EDF_C} &= \Pr(\text{The relative deadline} < \mathbb{C}_C) + (1 - \Pr(\text{The relative deadline} < \mathbb{C}_C)) \times \\ &\Pr(\text{size of at least one packet among } W \text{ packets} > T_{rem} * BW) \end{aligned} \quad (16)$$

From the earlier analysis, we get

$$\Pr(\text{The relative deadline} < \mathbb{C}_C) = \psi(\mathbb{C}_C) \quad (17)$$

$\Pr(\text{size of at least one packet among } W \text{ packets} > T_{rem} * BW)$
On the other hand, the last term of eq. (16) can be written as

$$\begin{aligned} &\Pr(\text{size of at least one packet among } W \text{ packets} > T_{rem} * BW) \\ &= 1 - \Pr(\text{size of all } W \text{ packets} < T_{rem} * BW) \\ &= 1 - \left\{ \Pr(\text{size of a packet} < T_{rem} * BW) \right\}^W \\ &= 1 - \left(\frac{1}{\Gamma(S_h) - \Gamma(S_l)} \left[\Gamma(T_{rem} * BW) - \Gamma(S_l) \right] \right)^W \end{aligned} \quad (18)$$

Replacing (17) in (18), we can prove (10).

With EDF_H: Unlike the traditional and coded broadcasting scheme, an event of deadline miss occurs in the proposed scheme only for the first case i.e., when the average interval of the consecutive transmission (i.e., $\mathbb{C}_H$) for the requestee vehicle is larger than the relative deadline. The second case of the earlier schemes does not apply for the proposed scheme since the removal of the vertex based on the T_{min} ensures that once an encoded packet is transmitted, none of the packets that constitute encoded packet will miss the deadline. Hence, the probability of the deadline miss due the packet size larger than the remaining deadline is zero. Note that, $\mathbb{C}_H$ will satisfy the constraints: $\mathbb{C}_H \geq \mathbb{C}_C$ and $\mathbb{C}_H \leq \mathbb{C}_T$. The rationales behind the constraints are that the transmitted encoded packet in the proposed scheme will constitute of one or more packets while the number of packets that constitute encoded packet will be less than equal W due to the elimination of the vertexes as proposed in the algorithm. From the above discussion, the deadline miss ratio (DMR) of the proposed scheme is given by:

$$\begin{aligned} DMR_{EDF_H} &= \Pr(\text{The relative deadline} < \mathbb{C}_H) \\ &= 1 - \frac{\mu}{\mathbb{C}_H} \left(1 - e^{-\frac{\mathbb{C}_H}{\mu}} \right) \\ &= \psi(\mathbb{C}_H) \end{aligned} \quad (19)$$

In Fig. 4, we present the theoretical DMR obtained from the above analysis. Note that we present the DMR with respect to the $\mathbb{C}_T$, from which $\mathbb{C}_C$ and $\mathbb{C}_H$ can be obtained by subtracting 5 and 2, respectively. Note that all $\mathbb{C}_T$, $\mathbb{C}_C$, and $\mathbb{C}_H$ can be obtained numerically which will have a proportional relationship with the number of vehicles.

VI. PERFORMANCE EVALUATION

A. Simulation model

The simulation model is based on the system architecture described in Section II and implemented using CSIM19 [18]. Table III shows the used parameters in simulation with relevant description. The justification of the parameters' values can be found in the related studies [3], [5], [11]. Unless stated otherwise, the simulation is conducted under the default setting. We follow the IEEE 802.11p PHY and MAC layers standard.

Each vehicle is simulated by a CSIM process. The vehicle mobility follows the Manhattan mobility model [15]. The

Fig. 4. DMR.

Fig. 5. Simulated multiple RSUs area.

simulation topology is shown in Fig. 5. Following Manhattan mobility, a vehicle can exit the simulation area through any of the nine exits. Then the CSIM process of the corresponding vehicle will be terminated. We continued the simulation until 95% confidence interval was achieved.

The data access pattern in submitting a request follows the commonly used Zipf distribution. The access probability of the i^{th} data item with a database of size M and with skewness parameter κ is $\frac{1}{i^\kappa} / \sum_{j=1}^M \frac{1}{j^\kappa}$; where κ equals 0 means the random distribution and increasing κ means the more skewed distribution.

For generating the heterogeneous data item sizes in an RSU database, we follow the random distribution, where the size of a data item d_i is,

$$S(d_i) = S(d_{min}) + \left\lfloor \text{random}(0.0, 1.0) \times (S(d_{max}) - S(d_{min}) + 1) \right\rfloor \quad (20)$$

where, $1 \leq i \leq M$, and $S(d_{min})$ and $S(d_{max})$ are the minimum and maximum size data item in the database, respectively.

B. Performance metrics

We use the following performance metrics for evaluating the system performance.

TABLE III
SIMULATION PARAMETERS.

Parameters	Default	Range	Description.
N_{RSU}	24	-	Number of RSUs
$N_{Vehicle}$	240	140-340	Number of vehicles
R	800	-	Maximum RSU transmission range
M	400	-	Size of an RSU database
κ	0.4	-	Zipf distribution parameter
$[Cache(Vh)]$	180	-	Cache size of a vehicle
$S(d_{min})$	1	-	Minimum size data item in database
$S(d_{max})$	45	15-55	Maximum size data item in database
L_{min}, L_{max}	2,3	1-3, 2-4	Min. and Max. laxity

Deadline miss ratio (DMR): DMR is the ratio of the number of deadline missed requests to the number of submitted requests in the system. A lower DMR means better system.

Average Response Time (ART): ART is the measurement of the responsiveness of the system. It is the elapsed time for satisfying a request from the instance the request is issued. However, note that ART is not a fair metric for measuring the system responsiveness while accessed data items are size variant. Hence we adopt another metric called *stretch*, which is defined as follows.

Average Stretch (AS): Stretch is defined as the ratio of the response time to the service time [12], where service time is the transmission time of the requested data item while the system is idle. Average stretch (AS) is the total stretches divided by the number of requests.

Available Serving Bandwidth (ASB): Available Serving Bandwidth (ASB) is the average available service channel bandwidth per broadcast. Note that without losing the general applicability, instead of true bandwidth we have used the normalized bandwidth as shown in Table I, where 6 Mbps equals 1.0 unit per broadcast tick. As the no-coding, conventional coding, and EDF_H approaches do not exploit the different MCS techniques, their ASBs' are always 1.0. In contrast, in our proposed ISXD_Θ approach, $ASB \geq 1.0$, and it varies based on system workload and vehicles' dynamic positions.

C. Performance analysis

Here we analyze the comparative performance of the proposed ISXD_Θ approach against EDF (no-coding) [?], [3], EDF_C [11] and EDF_H [12]. Note that EDF (Earliest Deadline First), is the foremost classical real-time on-demand scheduling algorithm which schedules requests based on the deadline urgency of a request. EDF_C is the representation of the conventional coding-assisted EDF. EDF_H is the representation of the coding-assisted EDF for heterogeneous data access.

1) *Impact of system workload:* The impact of system workload in terms of number of vehicles is shown in Fig. 6. With the increasing number of vehicles, all the approaches endure increased deadline miss ratio (DMR) (Fig. 6a). This is because, with the higher number of pending requests in the RSU service queue, average waiting time increases for a request before being scheduled for service (as a consequence average response time (ART) (Fig. 6b) and average stretch (AS) (Fig. 6c) also increase), hence more pending requests may miss their deadlines.

Fig. 6. Performance under different number of vehicles.

Surprisingly, we see that conventional network coding (EDF_C) cannot retain its superiority over the no-coding EDF in DMR and ART for accessing the diverse sized data items. Recalling that when different sizes data items are encoded, the size of the encoded packet will be the size of the maximum size data item among all the data items in the encoded packet. Hence if the encoded packet contains bigger and smaller size data items together, the urgent request (having smaller slack time than the encoded packet transmission time) may miss its deadline (Fig. 6a). Due to the same reason, smaller size data item might be penalized (needs to wait till the complete transmission of the encoded packet) due to the higher transmission time of the encoded packet; hence ART of EDF_C is also higher than EDF as shown in Fig. 6b.

However, in EDF_H, admission of data item in the encoded packet is controlled by a dynamic THRESHOLD (Θ) and the slack time of the most urgent request. By this way, EDF_H overcomes the limitation of EDF_C. As shown in Fig. 6a~6c, EDF_H has better DMR, ART and AS than EDF_C and EDF. Nevertheless, the proposed approach ISXD_Θ further improves the performance of EDF_H. Because, along with the Θ , ISXD_Θ also exploits the different bandwidth of different MCSs with the dynamic positions of vehicles. In addition, ISXD_Θ chooses the most priority request not only by considering the slack time of the request but also considering the vehicle position. This helps to exploit the higher serving bandwidth as well as serving the urgent request with priority. Moreover, ISXD_Θ greedily chooses the higher available serving bandwidth if in case some vertices are excluded due to the restriction of Θ . This is reflected in Fig. 6d, which shows that while ASB of all the approaches is always 1.0, the ASBs of ISXD_Θ are always higher than 2.0 under different workload settings. This also affects in the DMR, ART, and AS performance as well,

Fig. 7. Deadline miss ratio (DMR) under different deadline ranges.

Fig. 8. Latency under different maximum data item sizes.

which shows that according to all the performance metrics ISXD_Θ is the best.

2) *Impact of stipulated deadline of request:* The impact of varying deadline ranges on the DMR performance is shown in Fig. 7. The deadline of a request R_i pending for data item d_j is $T_{R_i}^{deadline} = ClockTime + RelativeDeadline$, where $RelativeDeadline$ is a value chosen within $T^{dwell}(Vh_i)$ inside an RSU range, which is defined by,

$$RelativeDeadline = \left\{ 1 + Uniform(L_{min}, L_{max}) * \frac{S(d_j)}{BW} \right\} \leq T^{dwell}(Vh_i)$$

where, L_{min} and L_{max} are the minimum and maximum laxity for varying $RelativeDeadline$, respectively. With the more strict to looser deadlines, all the approaches show improving DMR performance. Nevertheless, under different deadline ranges, ISXD_Θ shows the better DMR performance than each of the studied approaches and relative performance position of each of the approaches are consistent with the result found in Fig. 6a.

3) *Impact of data item size:* The responsiveness of the system is tested by varying maximum size data item ($S(d_{max})$) in Eq. (20) and the impact is shown in Fig. 8. The both ART and AS results are consistent with the results found in Fig. 6b and Fig. 6c, respectively, under different settings of data item size distribution. It confirms that the proposed ISXD_Θ consistently performs the best among all the studied approaches.

In summary, the proposed approach ISXD_Θ can further improve the system capability in satisfying more requests and improving the system responsiveness in accessing heterogeneous data items.

VII. CONCLUSIONS

Majority of the coding based on-demand data access existing solutions consider the accessed data items are equal size. In this paper, we have studied the coding based on-demand heterogeneous data access problem in vehicular networks. We have proposed a new algorithm, which exploits the integration of application layer scheduling with IEEE 802.11p physical layer different data rates (MCS) in data coding, for efficient serving of heterogeneous data items. Simulation results verify the superiority of the proposed approach.

REFERENCES

- [1] K. Liu, L. Feng, P. Dai, V. C. S. Lee, S. H. Son, and J. Cao, "Coding-assisted broadcast scheduling via memetic computing in sdn-based vehicular networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. PP, no. 99, pp. 1–12, 2017.
- [2] Z. Wang, J. Zheng, Y. Wu, and N. Mitton, "A centrality-based rsu deployment approach for vehicular ad hoc networks," in *2017 IEEE Int. Conference on Communications (ICC)*, May 2017, pp. 1–5.
- [3] G. Nawaz Ali, P. H. J. Chong, S. K. Samantha, and E. Chan, "Efficient data dissemination in cooperative multi-RSU vehicular ad hoc networks (VANETs)," *J. Syst. Softw.*, vol. 117, no. C, pp. 508–527, Jul. 2016.
- [4] G. N. Ali, E. Chan, and W. Li, "Supporting real-time multiple data items query in multi-RSU vehicular ad hoc networks (VANETs)," *Journal of Systems and Software*, vol. 86, no. 8, pp. 2127–2142, Aug. 2013.
- [5] J. B. Kenney, "Dedicated short-range communications (DSRC) standards in the united states," *Proc. of the IEEE*, vol. 99, no. 7, 2011.
- [6] K. Abboud, H. A. Omar, and W. Zhuang, "Interworking of dsrsc and cellular network technologies for v2x communications: A survey," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 12, pp. 9457–9470, Dec 2016.
- [7] D. Aksoy and M. Franklin, " $R \times W$: A scheduling approach for large-scale on-demand data broadcast," *Journal of IEEE/ACM Transactions on Networking (TON)*, vol. 7, no. 6, pp. 846–860, December 1999.
- [8] J. Chen, V. C. S. Lee, K. Liu, and J. Li, "Efficient cache management for network-coding-assisted data broadcast," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 4, pp. 3361–3375, April 2017.
- [9] K. Liu, J. K. Y. Ng, J. Wang, V. C. S. Lee, W. Wu, and S. H. Son, "Network-coding-assisted data dissemination via cooperative vehicle-to-vehicle/-infrastructure communications," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 6, pp. 1509–1520, June 2016.
- [10] J. Chen, V. C. Lee, K. Liu, G. Ali, and E. Chan, "Efficient processing of requests with network coding in on-demand data broadcast environments," *Information Sciences*, vol. 232, no. 0, pp. 27 – 43, 2013.
- [11] C. Zhan, V. Lee, J. Wang, and Y. Xu, "Coding-based data broadcast scheduling in on-demand broadcast," *Wireless Communications, IEEE Transactions on*, vol. 10, no. 11, pp. 3774–3783, November 2011.
- [12] G. G. M. N. Ali, M. A. Rahman, S. K. Samantha, Y. Gao, P. H. J. Chong, and Y. L. Guan, "Efficient coding based heterogeneous data access in vehicular networks," in *2017 IEEE International Conference on Communications (ICC)*, May 2017, pp. 1–6.
- [13] S. Acharya and S. Muthukrishnan, "Scheduling on-demand broadcasts: new metrics and algorithms," in *Proceedings of the 4th annual ACM/IEEE international conference on Mobile computing and networking (MobiCom'98)*, 1998, pp. 43–54.
- [14] Q. Wang, S. Leng, H. Fu, and Y. Zhang, "An IEEE 802.11p-based multichannel MAC scheme with channel coordination for vehicular ad hoc networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 2, pp. 449–458, June 2012.
- [15] F. Bai, N. Sadagopan, and A. Helmy, "THE IMPORTANT framework for analyzing the impact of Mobility on Performance Of Routing protocols for Adhoc Networks," *Journal of Ad Hoc Networks*, vol. 1, pp. 383–403, 2003.
- [16] J. Xu, Q. Hu, W. Lee, and D. L. Lee, "Performance evaluation of an optimal cache replacement policy for wireless data dissemination," *IEEE Transactions on Knowledge and Data Engineering*, vol. 16, no. 1, pp. 125–139, January 2004.
- [17] P. Xuan, S. Sen, O. Gonzalez, J. Fernandez, and K. Ramamritham, "Broadcast on demand: Efficient and timely dissemination of data in mobile environments," in *Proceedings of the 3rd IEEE Real-Time Technology and Applications Symposium (RTAS'97)*, 1997, pp. 38–48.

- [18] H. Schwetman, “CSIM19: A powerful tool for building system models,” in *Proceedings of the 33th IEEE Winter Simulation Conference*, Arlington, VA, USA, 2001.