Applications of Temporal Network Coding in V2X Communications

Xiaoli Xu(✉), Yumeng Gao, and Yong Liang Guan

School of Electrical and Electronic Engineering,
Nanyang Technological University, Singapore, Singapore
{xu0002li,ygao005}@e.ntu.edu.sg, eylguan@ntu.edu.sg

Abstract. Due to network dynamics and channel fading, multi-hop communication in vehicular networks usually suffers from much higher packet loss rate than the conventional static or single-hop networks. By encoding over packets received at different time slots at the intermediate nodes, temporal network coding (TNC) is a promising technique to avoid erasure accumulation with communication hops. In this paper, we present different strategies of TNC schemes to meet the decoding and delay requirements of different V2X (vehicle to everything) applications. Specifically, for applications with stringent delay requirement, such as live video streaming, we propose to use chunked TNC without precoding. For multi-hop communications with high throughput requirement, we propose to apply carefully designed precoding on top of the TNC to enhance the end-to-end throughput. Different from the conventional TNC code designs, we apply TNC design with overhearing, exploiting the broadcast nature of wireless communication. Specifically, we assume that a vehicle can not only receive the packets from its immediate upstream vehicle, but also overhear some packets from further-upstream vehicles. The number of network coded packets generated at the intermediate nodes is designed by considering the packets received via both the upstream and overheard transmission, which helps to maximize the communication throughput delivered to the destination node.

Keywords: Temporal network coding · V2X communications
Video streaming · Content distribution · Overhearing

1 Introduction

With the emerging technology of vehicular ad-hoc networks (VANETs), vehicles are able to communicate with fixed roadside infrastructure as well as other moving vehicles while traveling on the road - this is referred to as V2X communications. Historically, V2X communications are mainly driven by safety applications such as collision avoidance, emergency warning and crossing in the absence of signaling. Most of the safety applications rely on the periodically broadcasting of basic safety message (BSM), which contains the vehicles' locations, speeds, accelerations and etc. With the increasing popularity of V2X communication system,

P. H. J. Chong et al. (Eds.): SmartGIFT 2018, LNICST 245, pp. 53–63, 2018.
https://doi.org/10.1007/978-3-319-94965-9_6

the designs of non-safety use-cases to offer drivers enhanced comfort and entertainment have extended the mission of intelligent transportation system (ITS) and attracted significant research interests [1]. The infotainment services, such as the video streaming and location-aware content distribution, usually need to deliver a large amount of information during short-lived communication sessions, which hence impose stringent delay and throughput requirement.

The characteristics of VANETs, such as highly dynamic network topology, error-prone wireless channels and a limited end-to-end bandwidth, impose great challenges on providing high quality safety and non-safety applications. By allowing the information to be encoded at the intermediate nodes, network coding is believed to be a promising technique for tackling the challenges and enhancing the end-to-end throughput of VANETs [2]. In general, network coding can be implemented in two dimensions: spatial and temporal. Spatial network coding (SNC), where the packets coming from different network links are coded together, is useful for alleviating the network bottleneck and it has been applied to enhance the efficiency of BSM exchange via road-side unit (RSU) in VANETs [3]. On the other hand, with temporal network coding (TNC), the packets arriving via the same link at different time slots are coded together. TNC helps to alleviate erasure accumulation across multiple communication hops and hence enhances the end-to-end network throughput. It has been shown that random linear network coding (RLNC) over a sufficiently large number of time slots can achieve the capacity of multi-hop erasure networks [4]. However, as the number of encoded packets increases, the network coding overhead and encoding/decoding complexity grows quickly, which may even overwhelm the benefit of network coding for networks with limited packet length.

In practice, a large file may be divided into multiple blocks and TNC is applied within each block [5], i.e., only the packets within the same block will be coded together, which is referred to as "chuncked TNC" in this paper. Then, the resultant network coding overhead and encoding/decoding complexity will be proportional to the coding block size, instead of the original file size. However, when the encoding block size is small, the number of erased packets for each coding block may deviate from its expectation obtained from long-term channel statistics. Therefore, extra redundant packets, over and above those expected from the network capacity, are usually required to ensure that the coding blocks can be decoded successfully with high probability. An effective approach to enhance the end-to-end throughput of TNC without increasing the encoding block size is via applying proper precoding across different coding blocks. For instance, in random annex code [6] and Gamma network code [7], the parity check constraints among different coding blocks are defined based on an erasure correction code, such as low-density parity check (LDPC) code. By designing parity check constraints based on the classical fountain code, the batched sparse (BATS) code introduced in [8] was shown to have negligible reception overhead for any given coding block size. A BATS code consists of an outer code and an inner code. The outer code is an extension of the traditional fountain code to matrix form, and the inner code employs RLNC at the intermediate nodes for packets within the same batch. The BATS outer and inner code can be jointly decoded using efficient belief-propagation algorithm. Note that while the throughput of the BATS network code is enhanced with precoding, the decoding delay may also increase dramatically. Specifically, the "avalanche" effect is observed for decoding the BATS code, i.e., most of packets can only be decoded after receiving sufficient number of packets for decoding the whole file. The application of BATS code for broadcasting to a group of closely-located nodes in wireless ad-hoc network has been considered in [10].

In this paper, we study the application of TNC for V2X communications. Based on the delay and throughput requirements of different vehicular applications, we propose different code designs. For the delay-stringent applications, such as live video streaming, we propose to apply chunked TNC without precoding so that each coding block can be decoded independently after a sufficient number of coded packets is received. To enhance the network throughput, we consider TNC design with overhearing enabled. Specifically, we assume that a vehicle can not only receive the packets from its immediate upstream vehicles, but also overhear the packets from other vehicles, due to the broadcasting nature of wireless communication. We derive explicit expressions for the number of network coded packets to be generated at each vehicle to ensure that its downstream vehicle can decode each coding block with high probability.

On the other hand, for the content distribution applications, e.g., map or promotion video distribution, the file can be used only after it is fully decoded, hence we should minimize the total number of required packet transmissions by applying proper precoding across the information blocks. To this end, we propose to apply the BATS coding scheme. To further enhance the network throughput, the recoding operations in BATS code are designed with the assumption that overhearing is enabled. The design of BATS code for multi-hop wireless communication with unknown channel statistics has been investigated in [9]. However, the recoding operations of BATS code in [9] follows the conventional RLNC without considering the packet overhearing. In [11,12], the number of recoded packets for each batch is optimized according to the number of innovative packets received for that batch in channel with and without memory, respectively. Note that additional computational complexity for rank evaluation of each batch is induced by implementing the recoding design proposed in [11,12]. In contrast, we consider the same number of recoded packets for all the batches at each intermediate node, which is designed to maximize the information delivered with overhearing enabled. To the authors' best knowledge, we are the first to consider BATS code design with overhearing.

Lastly, the application of BATS code for joint V2I and V2V content distribution is discussed, where the content distribution is completed in two phases: the RSU broadcasting phase and the peer-to-peer (P2P) cooperative sharing phase. The rateless nature of BATS outer code allows the RSU to broadcast continuously without repetition, and the BATS inner code enables efficient P2P communication with network coding among vehicles.

2 Multi-hop V2V Communication

As shown in Fig. 1, we consider multi-hop V2V communication where certain file, e.g., live video captured by the camera at V_1 or infotainment content downloaded from the RSU by V_1, needs to be transmitted from the front vehicle V_1 to all other vehicles in the platoon, $V_2, \ldots, V_n$. We assume that the distance between adjacent vehicles V_i and V_{i+1} is constant and denoted by d_i. This is a reasonable approximation in practice when the vehicles are traveling with similar velocity. Further denote by $p_{ji}, j > i$ the packet loss rate experienced by the potential receiver V_j when V_i is transmitting. In general, we have $p_{j_1 i} > p_{j_2 i}$ when $j_1 > j_2$ due to the larger transmitting distance. We assume that the file is partitioned into K packets, each of L bits. We further assume that one common channel that supports transmission rate R_b bits/second is shared by all the vehicles based on time-division method. The vehicles in the platoon are assumed to be time synchronized, and time is slotted so that each packet transmission consumes exactly one time slot, i.e., time slot duration is specified as $T_s = L/R_b$. The packets are transmitted in the form of broadcasting with no feedback.

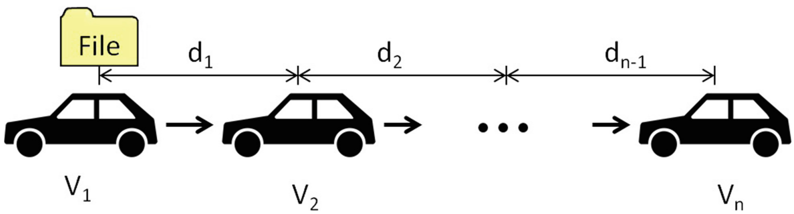

Fig. 1. Multi-hop V2V communication

The end-to-end throughput of this network, ρ, is defined as the ratio between input file size K and the total number of times slots required for delivering the file from V_1 to V_n, which is denoted as T_{total}, i.e.,

$$\rho = \frac{K}{T_{\text{total}}}. \tag{1}$$

The decoding delay, D, is defined as the average number of time slots between a packet being generated at V_1 until it is decoded by the last vehicle V_n. For chunked TNC without precoding, the decoding delay depends on the number of coded packets to be generated for each coding block. On the other hand, due to the "avalanche" decoding effect, the decoding delay for BATS code is equivalent as the total number of time slots required for decoding the complete file, i.e., $D = T_{\text{total}}$.

2.1 Live Video Streaming with Chunked TNC

In this subsection, we consider live video streaming from V_1 to all the subsequent vehicles. To minimize the delay, the video file is divided into small blocks of M

packets, where $M \ll K$, and RLNC is applied within each block. We assume that the network coding coefficients are chosen from a sufficiently large field, and hence the video block can be decoded from any M network coded packets. With chunked TNC, the front vehicle V_1 will send out Y_1 network coded packets when a block of M packets is generated. The number of packets received by V_2 out of the Y_1 coded packets that have been sent out by V_1 is a random variable distributed according to binomial distribution $\mathcal{B}(N, 1-p_{21})$, which can be approximated by the Gaussian distribution $\mathcal{N}(\mu_{21}, \sigma_{21}^2)$, where $\mu_{21} = Y_1(1-p_{21})$ and $\sigma_{21}^2 = Y_1(1-p_{21})p_{21}$. To ensure that V_2 can decode the video block with probability at least $1-\epsilon$, where $\epsilon \ll 1$, we should have

$$Q\left(\frac{M-\mu_{21}}{\sigma_{21}}\right) \geq 1-\epsilon, \tag{2}$$

where $Q(x) \triangleq \frac{1}{2\pi}\int_x^\infty e^{-u^2/2}du$ denotes the Gaussian Q-function.

By solving (2), we can find the minimum number of packets to be sent by V_1 as

$$\begin{aligned} Y_1 &= \frac{2M + \alpha^2 p_{21} + \alpha\sqrt{4Mp_{21} + \alpha^2 p_{21}^2}}{2(1-p_{21})}, \\ &\overset{(a)}{\approx} \frac{M + \alpha\sqrt{Mp_{21}}}{1-p_{21}}, \end{aligned} \tag{3}$$

where $\alpha \triangleq |Q^{-1}(1-\epsilon)|$ and (a) follows from the fact that $\alpha^2 p_{21} \ll M$.

After V_2 successfully decode the block, it will re-generate Y_2 network coded packets from this block, where Y_2 is chosen to ensure that V_3 can decode the block with high probability, together with those overheard packets during the previous Y_1 transmissions. Note that the number of packets overheard by V_3 is a random variable distributed according to $\mathcal{B}(Y_1, 1-p_{31})$, and the number of coded packets received by V_3 from the dedicated transmission by V_2 is a random variable distributed by $\mathcal{B}(Y_2, 1-p_{32})$. Therefore, the total number of packets received by V_3 can be approximated by a Gaussian random variable distributed according to $\mathcal{N}(\mu_3 + Y_2(1-p_{32}), \sigma_3^2 + Y_2(1-p_{32})p_{32})$, where $\mu_3 = Y_1(1-p_{31})$ and $\sigma_3^2 = Y_1(1-p_{31})p_{31}$. To ensure that V_3 can decode with probability $1-\epsilon$, we can solve for the minimum number of transmissions by V_2 as

$$Y_2 = \frac{M - \mu_3 + \alpha\sqrt{(K-\mu_3)p_{32} + \sigma_3^2}}{1-p_{32}}. \tag{4}$$

Following the similar analysis, the total number of coded packets overhead from V_i can be approximated by a random variable distributed according to $\mathcal{N}(\mu_i, \sigma_i^2)$, where

$$\mu_i = \sum_{j=1}^{i-2} Y_j(1-p_{ij});\ \sigma_i^2 = \sum_{j=1}^{i-2} Y_j p_{ij}(1-p_{ij}). \tag{5}$$

Hence, the number of network coded packets generated at all the vehicles can be evaluated in a recursive manner as

$$Y_{i-1} = \frac{M - \mu_i + \alpha\sqrt{(M - \mu_i)p_{i(i-1)} + \sigma_i^2}}{1 - p_{i(i-1)}}. \; i = 2, \ldots, n-1, \tag{6}$$

The network throughput achieved with the proposed chunked TNC is then given by $\rho_{\text{TNC}} = \frac{M}{\sum_{i=1}^{n-1} Y_i}$ and the decoding delay is given by $D_{\text{TNC}} = \sum_{i=1}^{n-1} Y_i$.

2.2 Content Distribution with BATS Code

In this subsection, we consider content distribution from V_1 to the subsequent vehicles. To achieve a good balance between the coding cost and the achievable throughput, we propose to apply BATS coding scheme with the outer code applied at the source V_1 and the inner code applied at the intermediate nodes $V_2, \ldots, V_{n-1}$. Thank to the BATS outer code, we don't have to receive sufficient number of coded packets for every batch to decode the complete file. Therefore, the number of recoded packets should be designed to maximize the useful information delivered, instead of ensuring high decoding probability of each block as in the preceding subsection.

We assume that the content consists of K packets in total. The front vehicle V_1 uses the BATS outer code to generate coded packets in batch, with each batch containing M innovative packets. In the conventional BATS coding scheme, the source node will send out exactly M coded packets and each intermediate node will re-generate M coded packets for each batch from their received packets. To incorporate those packets received via overhearing, we consider a general BATS coding scheme where V_i will send out Y_i coded packets for each batch, for $i = 1, \ldots, n-1$. The number of innovative packets received by V_i for each batch is a random variable denoted by $\mathbf{X}_i$. Denote by η, where $\eta \ll 1$, the coding overhead for BATS code. Then, the original file with K input packets can be decoded from N batches, where $N = \frac{K(1+\eta)}{\mathbb{E}[\mathbf{X}_i]}$ with $\mathbb{E}[\cdot]$ denoting the expectation function. Hence, the end-to-end throughput achieved with the proposed BATS coding scheme is given by $\rho_{\text{BATS}} = \frac{K}{N\sum_{i=1}^{n-1} Y_i}$ and the decoding delay is approximated by $D_{\text{BATS}} = N\sum_{i=1}^{n-1} Y_i$.

Our main task is to maximize the achievable throughput ρ_{BATS} via optimization of $Y_1, \ldots, Y_{n-1}$. However, due to the dependence between those packets received via overhearing and those received via dedicated transmission, it is challenging to characterize the distribution of $\mathbf{X}_i$. For simplicity, we assume that those packets received via overhearing are independent of each other. This assumption is valid when the batch size M is relatively large and the packet loss rates of overhearing links are high. Denote the average number of packets overheard by V_i per batch as O_i, we then have

$$O_i = \sum_{j=1}^{i-2} Y_j(1 - p_{ij}), i = 3, .., n. \tag{7}$$

Further denote by Z_i the average number of packets received by V_i per batch via dedicated transmission from V_{i-1}, and we have $Z_i = Y_{i-1}(1 - p_{i(i-1)})$. Since a batch contains at most M innovative packets, the expected number of innovative packets received by V_i can be estimated as

$$\mathbb{E}[\mathbf{X}_i] \approx \min\{O_i + Z_i, M\}. \tag{8}$$

We assumed that the packets overheard by V_{i+1} form a subspace of $\mathbf{X}_i$ innovative packets received by V_i. Then, the additional number of innovative packets that V_{i+1} can received from the dedicated transmission by V_i is given by $\mathbb{E}[\mathbf{X}_i] - O_{i+1}$. Since the channel between V_i and V_{i+1} has packet loss rate $p_{(i+1)i}$, the expected number of transmissions required for delivering all the additional information is

$$Y_i = \frac{\mathbb{E}[\mathbf{X}_i] - O_{i+1}}{1 - p_{(i+1)i}}, \tag{9}$$

By substituting the initial condition $O_2 = 0$, we can derive the recursive expressions for $Y_i, i = 1, \ldots, n-1$ from (7)–(9) as

$$Y_1 = \frac{M}{1 - p_{21}}, \tag{10}$$

$$Y_i = \frac{M - \sum_{j=1}^{i-1} Y_j(1 - p_{(i+1)j})}{1 - p_{(i+1)i}}, i = 2, \ldots, n-1. \tag{11}$$

The actual throughput and decoding delay of BATS coding scheme depends on the number of batches N required for decoding the file, which further depends on the number of innovative packets received at V_n per batch, i.e., $\mathbb{E}[\mathbf{X}_n]$. It is difficult to obtain explicit expression for $\mathbb{E}[\mathbf{X}_n]$, but it can be easily obtained from simulations.

2.3 Numerical Example

In this subsection, numerical examples are presented to evaluate the performance of the proposed scheme. We assume that there are n vehicles in the network, i.e., $n = 8$ and the neighboring vehicles are separated by equal distance of 50 m. Each vehicle transmits with constant power at 20 dBm. The V2V communication channel has path loss exponent 2.3 and suffer from Nakagami fading with shaping factor 2. A packet can be successfully received if the received SNR is above the receiving threshold of 15 dB. Then, the packet loss rate between V_i and V_j can be obtained as

$$p_{ij} = \begin{cases} 0.0642, & i-j=1 \\ 0.5987, & i-j=2 \\ 0.9636, & i-j=3 \\ 1, & i-j \geq 4 \end{cases} \tag{12}$$

We assume that TNC is applied within a block of 32 packets, i.e., $M = 32$ for both coding schemes proposed in the preceding subsections. The maximum decoding failure probability is set as $\epsilon = 0.01$ in the chunked TNC schemes and the BATS coding overhead is assumed to be $\eta = 0.01$. The achievable throughput for the proposed chunked TNC and BATS coding schemes are compared with the benchmarks where overhearing is not allowed in Fig. 2. As expected, BATS coding schemes usually achieve higher throughput than chunked TNC with the same coding block size, at the cost of larger decoding delay. Furthermore, it is observed that the proposed code designs with overhearing enabled always outperform the benchmark schemes where overhearing is not allowed.

Note that the delay of the chunked TNC is inversely proportional to the achievable throughput as $D_{TNC} = \frac{M}{\rho_{TNC}}$ and hence the proposed chunked TNC with overhearing will have smaller delay as compared to the conventional chunked TNC without overhearing. On the other hand, the delay of the BATS coding schemes is not only related to the coding block size M, but also the original file size K. If a relatively large file is considered, the delay of BATS coding scheme will be much larger than those chunked TNC schemes, regardless whether overhearing is enabled.

3 Joint V2I and V2V Content Distribution

In this section, we briefly discuss the application of BATS coding scheme for content distribution from a single RSU to a group of vehicles passing by it, as shown in Fig. 3. To combat the short connection time and the lossy channel, we propose to further share the contents among the vehicles after they leave the communication range of the RSU. The rateless nature of BATS code allow the RSU to generate unlimited number of coded packets without repetition, and hence greatly simplify the scheduling as compared with the conventional

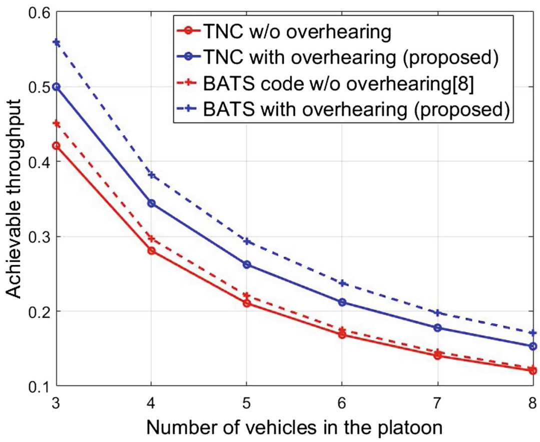

Fig. 2. Comparison of the achievable throughput.

fixed rate coding schemes. Furthermore, the RLNC code within the each batch enhances the efficiency of P2P cooperative sharing. Specifically, the proposed scheme consists of the following two phases:

- *RSU information broadcasting*: The input file is encoded into a set of batches at the RSU and they are sequentially broadcast during the period from the first vehicle entering the RSU communication range until the last vehicle leaving this range. Note that the total transmission time is much larger than the contacting time of a single vehicle with the RSU. Furthermore, due to spatial diversity, the probability that a packet is received by at least one of the vehicles is much larger than the probability that it is received by one specific vehicle. Hence, the number of innovative packets received by the whole group is usually much larger than the number of packets received by one individual vehicle.
- *P2P cooperative sharing*: If the file size is very large such that one vehicle cannot decode the original file based on its own received packets from the RSU, it will broadcast the request for initiating the P2P cooperative sharing phase. In P2P sharing phase, when a vehicle has the chance to access the channel, it will generate and broadcast a coded packet from all the received packets of a batch. The scheduling on which batch should be selected at a given time instance has been designed to maximized the utility of each transmission, with the details provided in [13]. Those packets received from RSU broadcasting phase and P2P cooperative sharing phase can be jointly decoded using belief-propagation algorithm and the P2P sharing phase ends when all the vehicles successfully decode the original file.

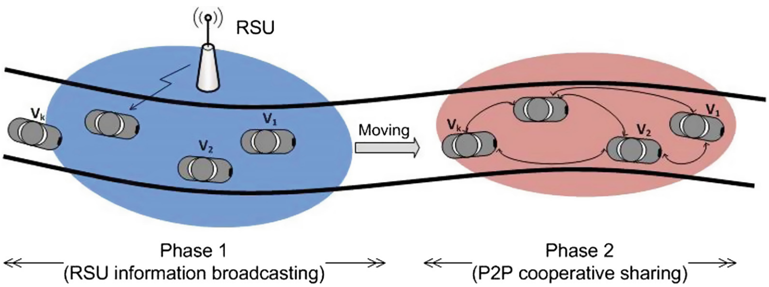

Fig. 3. Joint V2I and V2V content distribution

A numerical example is presented by considering a file with 18 MB distributed by the RSU which is located by the side of the two-lane straight road, to a group of vehicles with relatively close speeds. The packet length is 1500 Bytes. Both the RSU and each vehicle transmits with constant power at 20 dBm. A dual-slop model is adopted to describe the large-scale path loss [13]. The BATS is applied within a batch of 16 packets, i.e., $M = 16$. Since the number of transmissions from the RSU to the vehicles in the first phase depends on the moving speed of the vehicles, we focus on the performance evaluation on the number of transmissions required in the P2P sharing, as shown in Fig. 4. Two benchmark schemes, CodeTorrent [14] and CodeOnBasic [15], adopting TNC, are used for the comparison purposes. Our proposed content distribution scheme with BATS coding requires the lowest number of P2P transmissions to successfully decode the file, significantly outperforming the benchmark schemes.

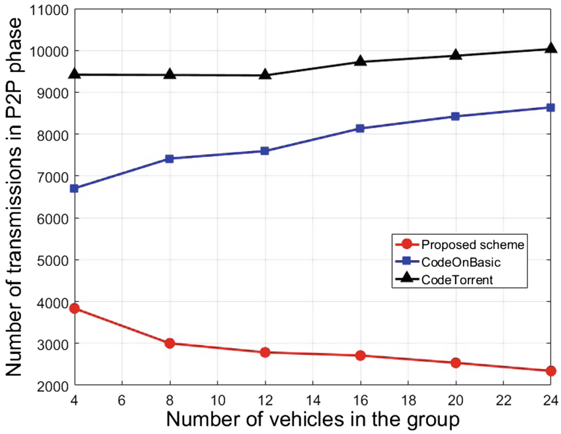

Fig. 4. Comparison of the number of required P2P transmissions.

4 Conclusion

We have discussed the design of TNC, with or without precoding, for different V2X applications to meet their delay and throughput requirements. Multi-hop V2V unicast, as well as joint V2I and V2V content broadcast have been considered. For applications with stringent delay requirement, we considered chunked TNC without precoding. For applications with high throughput requirement, we proposed to use BATS coding scheme. In both cases, the network code designs are augmented with overhearing. The proposed code designs with overhearing are shown to always outperform the benchmark schemes without overhearing.

References

1. Gerla, M., Wu, C., Pau, G., Zhu, X.: Content distribution in VANETs. Veh. Commun. **1**(1), 3–12 (2014)
2. Jamil, F., Javaid, A., Umer, T., Rehmani, M.H.: A comprehensive survey of network coding in vehicular ad-hoc networks. Wirel. Netw. **23**(8), 2395–2414 (2017)
3. Nawaz Ali, G.G. Md., Rahman, A. Md., Chong, P.H.J., Samantha, S.K.: On efficient data dissemination using network coding in multi-RSU vehicular ad hoc networks. In: IEEE Vehicular Technology Conference (2016)

4. Lun, D.S., Medard, M., Effros, M.: On coding for reliable communication over packet networks. Phys. Commun. **1**(1), 3–20 (2008)
5. Chou, P.A., Wu, Y., Jain, K.: Practical network coding. In: Allerton Conference on Communication Control, and Computing, October 2003
6. Li, Y., Soljanin, E., Spasojevc, P.: Effect of the generation size and overlap on throughput and complexity in randomized linear network coding. IEEE Trans. Inf. Theory **57**(2), 1111–1123 (2011)
7. Mahdaviani, K., Yazdani, R., Ardakani, M.: Overhead-optimized gamma network codes. In: International Sympousim Network Coding (NetCod), June 2013
8. Yang, S., Yeung, R.W.: Batched sparse codes. IEEE Trans. Inf. Theory **60**(9), 5322–5346 (2014)
9. Xu, X., Guan, Y.L., Zeng, Y., Chui, C.-C.: Quasi-universal BATS code. IEEE Trans. Veh. Technol. **66**(4), 3497–3501 (2017)
10. Xu, X., Kumar, M.S.G.P., Guan, Y.L., Chong, P.H.J.: Two-phase cooperative broadcasting based on batched network code. IEEE Trans. Commun. **64**(2), 706–714 (2016)
11. Tang, B., Yang, S., Ye, B., Guo, S., Lu, S.: Near-optimal onesided scheduling for coded segmented network coding. IEEE Trans. Comput. **65**(3), 929–939 (2016)
12. Xu, X., Guan, Y.L., Zeng, Y.: Batched network coding with adaptive recoding for multi-hop erasure channels with memory. IEEE Trans. Commun. **63**, 1042–1052 (2017)
13. Gao, Y., Xu, X., Guan, Y.L., Chong, P.H.J.: V2X content distribution using batched network code (2017). https://arxiv.org/abs/1712.00946
14. Lee, U., Park, J.-S., Yeh, J., Pau, G., Gerla, M.: Code torrent: content distribution using network coding in VANET. In: Proceedings of the 1st International Workshop on Decentralized Resource Sharing in Mobile Computing and Networking, pp. 1–5. ACM (2006)
15. Li, M., Yang, Z., Lou, W.: CodeOn: cooperative popular content distribution for vehicular networks using symbol level network coding. IEEE J. Sel. Areas Commun. **29**(1), 223–235 (2011)