Available online at www.sciencedirect.com
ScienceDirect
Procedia Computer Science 83 (2016) 131 – 138
Procedia Computer Science

The 7th International Conference on Ambient Systems, Networks and Technologies (ANT 2016)

Pollution Attack Resistance Dissemination in VANETs Based on Network Coding

Guowei Wu[a], Jie Wang[a], Yongchuan Wang[b], Lin Yao[a]

[a]*School of Software, Dalian University of Technology, 116621, Dalian, China*
[b]*531,Jiangcheng Avenue,Wuhan Economic &Technological Development Zone,430056,China*

Abstract

Network coding is widely used in the dissemination schemes of VANETs, because it can improve the network throughput. However, it will bring the pollution attack into the network, making the decoding procedure error, so vehicles can not recover the original file. Therefore, we need adopt a signature scheme to validate a piece without decoding. In the current signing schemes, the linear subspace signature scheme is to defend the pollution attack. But the length of the signature equal to the piece size required several packets to be transmitted together. Moreover, even one lost packet or polluted packet may make the whole piece dropped including the unpolluted packets, causing the limited resources to be wasted. In this paper, we adopt the padding scheme, obtain a packet-size vector which is orthogonal to linear space spanned by all packets in a generation and sign the vector, reducing the length of the signature into packet size and more importantly validating coded packets other than coded pieces in a generation. The simulation shows that our scheme has higher downloading rate, and lower downloading delay.

Peer-review under responsibility of the Conference Program Chairs
Keywords: VANET; dissemination; pollution attack; network coding.

1. Introduction

VANETs gain much more attention because it can provide safety and optimize road traffic. The content dissemination, related to both commercial and safety service, is the most promising one. In VANETS, the road side unit(RSU) transfers the content from the passing vehicles, and then disseminates it in the area of the interest. The content dissemination includes emergency video from the ambulance to the nearby vehicles, a map of current region, etc..

In VANETs, the simplest dissemination way is flooding. In flooding, a message is broadcasted from every node to all its encounters until the message reaches a predefined maximum hop count (i.e., Time-To-Live or TTL value) or the destination. But it suffers from the problems of large overhead and high redundancy, leading to the so-called broadcast storm problem. Network coding has seen its application in information dissemination [1]. Network coding can

E-mail address: yaolin@dlut.edu.cn

Peer-review under responsibility of the Conference Program Chairs
doi:10.1016/j.procs.2016.04.108

achieve broadcast capacity, improve bandwidth efficiency and network throughout. However, adversaries in VANETs can inject polluted messages or forged messages. Such packet pollution attacks can severely lower dissemination performance[2 3 4 5]. Due to network coding's unique usage of packets, dissemination techniques using networking coding in VANETs suffer more severely from pollution attacks. For example, one polluted packet can easily invalidate all downstream packets, wasting all the resources to prepare/transmite/receive/decode those messages.

Another issue is the block size used network coding strategies[6 7 8 9 10]: block sizes have been chosen as 2 kilobytes (KB), 4KB, 10KB, or 32KB. And both the signature and the block must be transmitted in several packets. If one packet is polluted or missing, the whole block becomes useless, which wastes the bandwidth.

In this paper, we present a dissemination protocol in VANETs using the padding skill to defend the pollution attack. Due to the large pieces and the signature in a generation, we divide the generation into many small sub-generations. By padding an extra symbol to each piece in the sub-generation, all the sub-generations can have the same signature in a generation. Both the pieces in the sub-generations and the signature can be transmitted in a single packet. All the packets belonging to the same generation can be verified by the corresponding signature.

This paper makes the following contributions:

- To defend the pollution attack, we adopt the homomorphic signature scheme in[11]. When service time slot starts, the relay node will broadcast the correspondence signature to in case of that some neighbors may not hold the signature before transmitting the coded packets of a sub-generation.
- We reduce the signature size from a piece size to a packet size. Therefore, the signature can be transmitted in one packet and be used to validate every packet in the correspondence generation.
- We improve the network bandwidth efficiency by adopting the network coding technique. It removes the waiting time at the beginning of the service time slot compared to CodeOnBasic([7]). Moreover, it utilizes every unpolluted packet to avoid dropping the whole piece even when a single packet is polluted.
- We compare our scheme with CodeOnBasic and CodeOnBasicP(adopting piece-length signatures[11] to defend the pollution attack). Compared with CodeOnBasic, our dissemination can not be completed when no defending scheme existing. Compared with CodeOnBasicP, our scheme can reduce the download delay by 42.5%

The rest of this paper is organized as follows. In section 2, we describe the related works about pollution attack and content dissemination in VANETs. In section 3, we introduce the network coding and pollution attack. In section 4 we analyze the problem formulation and proposes solution. The main design of this paper is presented in Section 5. In section 6 we present the performance and results. Finally, the paper is summarized in Section 7.

2. Related Works

Pollution attacks are fatal to the network coding system if they are out of control. The main methods to solve this attack are based on algebraic and cryptography.

Algebra schemes can be used to defend the pollution attack.In[12], it calculates a vector which is orthogonal to the linear space spanned by the plain pieces in a generation and sends to the nodes. All coded pieces belong to the subspace. When receiving a coded piece, the corresponding vector can be used to validate the integrity of the piece. However, a malicious node can generate a fake piece which does not belong to the subspace to pass the validation. In[13], every node has a vector which is orthogonal to all the plain pieces of the source generation, it can rapidly verify the coded pieces by validating whether it belongs to the linear subspace or not.

Apart from the algebraic schemes, cryptography has been well-recognized as a effective method to solve pollution attack, including hash, signature and MAC. Homomorphic hash scheme is first proposed in[14]. The source node first computes the hash of each piece in a generation, and all the hash should be pre-disseminated to all the nodes. In order to check a received coded piece, nodes should compute the hash of this piece, and compare it with the linearly combining hash distributed at the beginning. The drawback is that before the plain pieces are transmitted, all the hashes should be delivered to all nodes. In[3], it proposes a new homomorphic signature schemes based on weilpairing on elliptic curve to defend the pollution attack. The main character of these schemes is that the signature of the linear combined coded piece is equal to the linear combination of the signatures. But the computational overhead is too high. In[5], it signs a subspace spanned by plain pieces in a generation. When receiving a coded piece, the signatures are used to validate if the pieces belong to the specific linear subspace. It is convenience to validate if the coded pieces are polluted, while the signature should be transmitted before the corresponding generation. In[11] it pads an extra tag to obtain a vector orthogonal to the pieces in a generation and can obtain the vector before all pieces are ready. However, both in[5] and[11] the size of the vector is equal to the piece size. The third type of cryptography to defend pollution attack is homomorphic MAC[15 16], which the MAC of a linear combination of pieces is equal to the linear combination of MACs correspondence to the pieces.

While the pollution attack is very serious in the network coding schemes, many protocols do not consider how to extend to defend the pollution attack when adopting network coding. In[7], nodes share its file reception status, and choose a node who can supply the most innovative pieces to neighbors as a relay, and then calculate a transmission backoff delay. The protocol using symbol level network coding(SLNC[17]) while no method can be used to defend the pollution attack at our best knowledge. The CodeOnBasic is also proposed in[7]. It adopts randomly linear network coding(RLNC[18]), but the pieces in a generation need many packets to send, while even one packet lost or polluted makes the whole piece unavailable. Both in[7] and in[8] the authors adopt SLNC which can be more tolerate to collision. If symbol size is equal to the packet size, the SLNC transforms to RLNC. In[9] source nodes send out a file description which represent the part received and other nodes keep sending request to neighbors for packets. In[10], content are downloaded from gateways to vehicles and exchanged between vehicles out of the range of gateways.

3. Preliminary

In this part, we first introduce the network coding skills[18], then present a solution to solve the pollution attack[11]. The frequently used notions in this paper are listed in Table 1.

Table 1: Symbols

Notation	Description
F	file
K	the number of pieces a generation consisting of
N	the number of generations divided by F
G_i	the i-th gerneration
$P_{i,j}$	a piece of G_i
$G_{i,j}$	the j-th sub-generation in G_i
$P_{i,j,k}$	a piece of $G_{i,j}$
J	the size of a piece in G_i
S	the size of a packet
Sig_i	the Signature of G_i

3.1. Network Coding

We assume a large file F has N generations G_1,G_2,...,G_N, and each generation G_i has K original pieces, $p_{i,0}$,$p_{i,1}$,...,$p_{i,K-1}$. Each piece in G_i is fixed to size J and the size of a packet is S, therefore each piece in $P_{i,j}$ has [J/S] packets $p_{i,j,0}$,$p_{i,j,1}$,...,$p_{i,j,[J/S]-1}$. When disseminating content using network coding, intermediate nodes broadcast coded pieces, rather than original pieces. At source node, each piece has a unit vector e which is orthogonal to the others piece's vector, that is $P_{i,k} = [e_{i,k}\ p_{i,k}]$ ($e_{i,k}$ is the unit vector and $p_{i,k}$ is the data set). The intermediate nodes randomly select coefficients to generate coded piece $\overline{P}_{i,k}$ for generation G_i using

$$\overline{P}_{i,k} = \sum_{j=1}^{K} c_{i,j} P_{i,j}. \tag{1}$$

The coefficient $c_{i,k}$ is randomly chosen from the Galois Field(GF). The receivers only keep the innovative pieces which are linearly independent to other pieces and generate coded packets using Equation (1). We assume the K pieces make up a K×K matrix K_i which is made up of the coefficient of each piece, and a K×J matrix X_i for coded data portions. When receiving K linear independent pieces, the node can recover the original piece using $P_i = K_i^{-1} X_i$.

3.2. Pollution Attack

Network coding can reduce the redundant packets and improve the bandwidth, it also brings serious pollution attack which can do great damage to the whole network. For example in Fig 1, if node Z is polluted, it will send a polluted piece Z_1. After Node B receives the the pieces, it perform a liner operation on Z_1 and A_1 obtaining two polluted coded pieces B_1 and B_2 and send to the downstream nodes. Therefore all nodes in the downstream will receive polluted pieces, which means they get polluted. Nodes can not receive enough correct pieces with the existence of polluted pieces, so that it is impossible to recover the original content correctly. To make full use of the channel resource, intermediate nodes should be able to distinguish the polluted pieces from the correct ones. If so, normal node can only filter out the polluted piece and broadcast the coded piece without pollution.

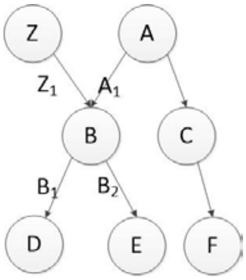

Fig. 1: The Influence of Pollution Attack.

4. Problems Formulation And the Solution

In this section, we first present the flaw in the existing signature schemes, then elaborate the improvement. In the remaining model, we present our algorithms based on utility, making relay decision.

4.1. Problems Formulation

In many schemes, the piece of a generation needs more than one packets to transmit[6 7 8 9 10] and the signature of a generation validates the correctness of each piece belongs to this generation. The size of the signature is equal to the size of the piece which is too large. What's more, collision often occurs, and pollution threat exists, some packets in a piece may be lost or polluted, which causes the whole piece dropped including the unpolluted packets. Therefore, finding a way to validate each packets rather than pieces is essential.

4.2. the Solution

In Fig 3(a), a piece size is $K + J$ which needs $(K + J)/S$ packets to transmit. We divide the generation G_i into J/S sub-generations as shown in Fig 3(b). In sub-generation $G_{i,j}$, the piece size is equal to the packet size, thus the concept of piece and the packet are the same. First, a vector v' is randomly chosen whose size is $K + J/S + S + 1$. Then every piece in $G_{i,j}$ is padded with an extra tag and the tag is set to make $wv = 0$. Therefore, the vector is orthogonal to the linear subspace spanned by $G_{i,j}$. Moreover, in G_i there are K/J sub-generations and each sub-generation has K original pieces, the length of each piece is $K + J/S + S + 1$. Because of $KJ/S < K + J/S + S + 1$, it is easy to make v' orthogonal to all packets in G_i according to the procedure list before. Signing v' according to[11], thus G_i can has a signature Sig_i for generation G_i. The signature Sig_i can be transmitted in one packet to validate every packet rather other pieces belonging to the G_i.

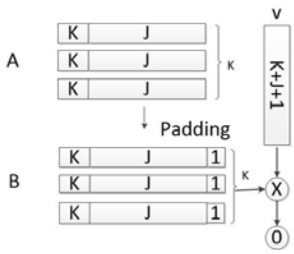

Fig. 2: Padding Packets and Getting Orthogonal Vector.

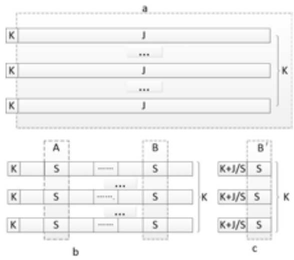

Fig. 3: Dividing G_i into $G_{i,j}$.

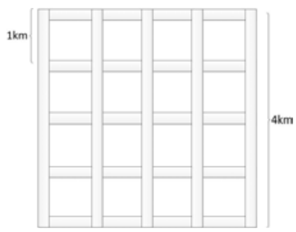

Fig. 4: Road Map.

5. Dissemination Based on Network Coding for Vanets

This paper is a push-based protocol using network coding which nodes/RSUs disseminate a large file F to all vehicles in a certain region to defend pollution attack. The source nodes/RSUs hold F and divide it into L generations. Every generation G_i consists of J/S sub-generations and they have the same signature. Network coding is performed in the intermediate nodes. In this section we will describe the channel first and then elaborate the routing decision.

5.1. Channel

According to DSRC[19], the frequency band allocated for VANETs consists of multiple channels. One channel is used as the control channel to transmit control messages including nodes' speed, generation reception status,signature reception status and etc. Other channels are used as the service channels to transmit coded data. We adopt the channels described in[7]. There are two channels, a control channel and a service channel. Each channel lasts for 50ms. And all vehicles include RSUs are synchronized to change between the control channel and the service channel. According to DSRC, the data rate can be up to 27Mb/s, which means 143KB data can be transmitted during 50ms.

5.2. Routing Decision

Every node/RSU broadcasts the coded content and exchanges their generations and signatures reception status. Relays are chosen according to the neighbors' status and coded packets are broadcasted to neighbor nodes. Our dissemination protocol consists of two steps as below:

- In control time slot, each node broadcasts its generations' and signatures' reception information. After receiving neighbors' reception status, node s calculates the number of innovative packets it can supply to all its neighbors for every sub-generation. We take the utility as the maximum number of innovative packets it can supply to neighbors among sub-generations.Then nodes exchanges its $Utility$ among its neighbors. The utility of node s is calculated:

$$Utility = Max(\sum_{t \in Nei}(R_{i,j,s} - R_{i,j,t})), t \in Nei \quad (2)$$

- In the service time slot, Nodes with the largest utility among its neighbors will access the service channel and distribute the coded packet. To defend the threat of pollution attack, the signature should be pre-distributed before transmitting coded packets of sub-generation $G_{i,j}$. At the beginning of the service time, node s can know whether all the neighbors have the signature of this sub-generation correspondence to its $Utility$. If all neighbors have this signature, node s will calculate the number of innovative packets $NumOfG_{i,j}$ it should send. Otherwise, node s will distribute the signature to neighbors at the beginning of the service time slot, and then disseminate the coded packets. After the $NumOfG_{ij}$ packets are sent and the service time slot is not over, to improve the bandwidth efficiency, node s will calculate the next biggest $Utility$ and repeat procedure b. When the service time slot is over, then stop broadcasting.

When node s accesses the channel, if it randomly create $NumOfG_{i,j}$ coded packets, the limit channel resource is wasted because some packets may be linearly dependent which does no help to decode and recover the original generation. The probability of all the $NumOfG_{i,j}$ is linearly independent is

$$Prob = \frac{K!}{(K - NumOfG_{i,j})!K^{NumOfG_{i,j}}} \tag{3}$$

To make as many as packets useful, nodes should record the packets sent before as a matrix M for each sub-generation, and M has K packets at most. If rank of M is less than K, then generate a innovative packet, broadcast to neighbors and add to M£ If rank of M is equal to K, then node s will resend the oldest packet recorded in M.

6. Performance

To evaluate the performance of our dissemination scheme, we use the NS2.34 simulator[20]. The vehicle movement patterns are generated by VanetMobiSim[21]. Vehicles are randomly placed in the road area. To evaluate the effect of the traffic density, we consider both the sparse and the dense urban scenarios. And we also consider the urban scenario and the highway scenario. The urban scenario is a 4km×4km urban zone as shown in Fig 4. The highway scenario is a 12km length highway zone. In urban scenario, the sparse setting has 200 vehicles while the dense setting has 400 vehicles. The speed of vehicles in urban scenario ranges from 30km/h to 60 km/h. In highway scenario, the sparse setting has 50 vehicles while the dense setting has 100 vehicles. The speed of vehicles in highway scenario ranges from 60km/h to 80 km/h. Each vehicle is equipped with a wireless device. The transmission rate is 11 Mbps and the transmission range is 250 meters. We assume each vehicle has a small percentage to generate a polluted pieces. And we evaluated the following metrics:

a) Downloading Progress. It shows the downloading percentage of the file with time passing by.
b) Average Download Delay. It is the average time cost to complete disseminating contents to all.
c) Dropping Packets. This is the number of packets dropped because of the polluted pieces received.

We compared the scheme this paper presented with other works. First, to demonstrate the effect of pollution attack when malicious nodes exists, we adopt CodeOnBasic[7] watching the performance difference when malicious nodes exists and no malicious nodes exists. While to our best knowledge, there is no other dissemination protocols in VANETs using pollution attack defending methods. Therefore, we introduce piece pollution validation skills to CodeOnBasic, and name it CodeOnBasicP. Then we compare our packet pollution validation scheme with CodeOnBasicP.

6.1. The Effect of Pollution Attack

From Fig5 and Fig6, it can be seen that with the existence of the malicious vehicles, the 95% packets are polluted packets in CodeOnBasic. At the beginning, source vehicle/RSU disseminates the content generations to neighbors. When polluted packets are broadcasted to neighbors, all the neighbors will receive and store the polluted packets, making the neighbors become pollution sources. Vehicles can not recover the original file due to the pollution packets in a generation. While in our scheme, vehicles can obtain the original file F, because the polluted packets are filtered out by the pollution attack solution. Therefore, it is necessary to use a pollution attack scheme in VANETs when using network coding, which should consider the specialty of VANETs.

6.2. Downloading Progress

From Fig 7 and 8, we can see it is very fast to reach to 90% percentage of the file downloaded both in our scheme and CodeOnBasicP in urban scenario, and then the downloading progress becomes slower. The comparison between our scheme over CodeOnBasicP in urban scenario demonstrates our protocol performs faster on completing the downloading both in sparse and dense settings. From Fig 9 and 10, we can see in the highway scenario, our scheme also performs better than CodeOnBasicP. When a single packet gets polluted in CodeOnBasicP, the whole piece which the packet belongs to will be dropped including the others unpolluted packets. Therefore, CodeOnBasicP

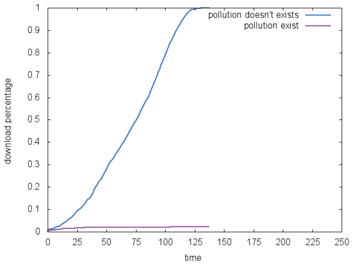

Fig. 5: Pollution in Urban Sparse Scenario.

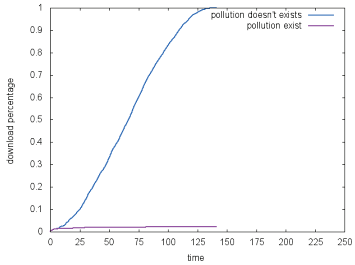

Fig. 6: Pollution in Urban Dense Scenario.

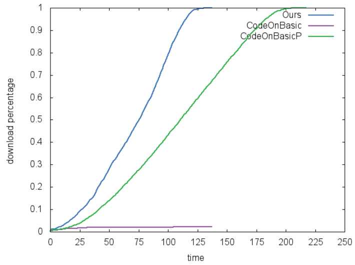

Fig. 7: Progress in Sparse Urban Scenario.

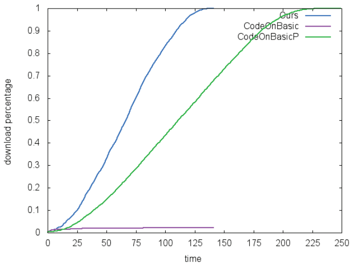

Fig. 8: Progress in Sparse Urban Scenario

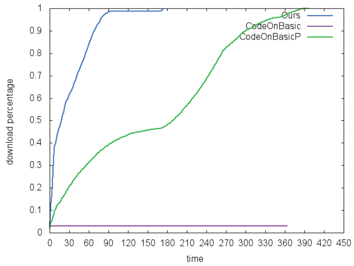

Fig. 9: Progress in Sparse Highway Scenario

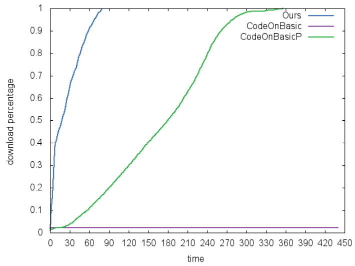

Fig. 10: Progress in Dense Highway Scenario

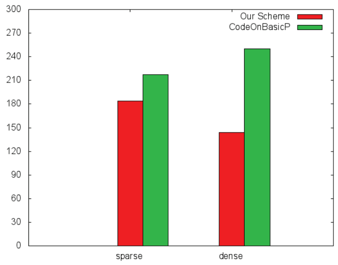

Fig. 11: Delay in Urban Scenarios.

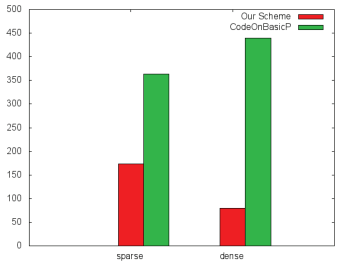

Fig. 12: Delay in Highway Scenarios.

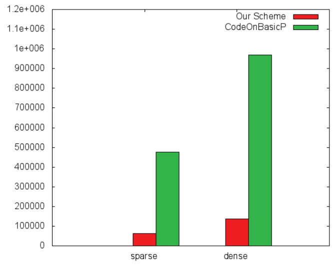

Fig. 13: The Number of Packets Dropped.

does not make full use of the bandwidth. In contrast, our scheme validates the correctness of every packet, make full use of every unpolluted packets and only drop the packets which can not pass the validation.

6.3. Downloading Delay

From Fig 11, we can see the downloading delay reduces 26% in sparse settings and 42.5% in dense settings compared to CodeOnBasicP in urban scenario. In highway scenario, the delay of our scheme can reduce 56.3% in sparse settings and 77.5% in dense settings compared to CodeOnBasicO from Fig 12. We can see when traffic becomes dense, we can see the downloading delay gets less in our scheme from Fig 11 and Fig 12. Although there is more vehicles, the collision frequency does not get higher because only one vehicle is chosen as a relay among its neighbors. Moreover, more neighbors a relay broadcasts content to, faster the contents are spread to other vehicles.

6.4. Dropping Packets

From Fig 13, we can see the number of dropped packets in CodeOnBasicP is about 10 times than that in our scheme. Because one piece in CodeOnBasicP is 10KB which needs ten packets to transmit. If a single packet in a piece gets polluted in CodeOnBasicP, the whole piece will be dropped directly. Moreover, the topology changes rapidly in VANETs, when a node moves out of the transmission range of the relay, the uncompleted received coded pieces will be dropped which wastes the bandwidth. In contrast, in our scheme we divide the generation into small sub-generation. A piece in our scheme only needs a packet to transfer. Even if a packets is polluted, vehicles will directly drop this packet and stop the pollution spreading to other vehicles.

7. Conclusion

In this paper, we propose a push-based dissemination scheme while defending the pollution attack. We study the impact of the pollution attack and adopt a novel way to reduce the signature of a generation. NS2 simulator is used to demonstrate the effectiveness of our dissemination scheme in terms of downloading progress and downloading delay. Simulation result shows our scheme can improve the download rate and reduce the delay in VANETs.

References

1. Shuo-Yen Robert Li, Raymond W Yeung, and Ning Cai. Linear network coding. *Information Theory, IEEE Transactions on*, 49(2):371–381, 2003.
2. J Krithiga and RC Porselvi. Efficient codeguard mechanism against pollution attacks in interflow network coding. In *Communications and Signal Processing (ICCSP), 2014 International Conference on*, pages 1384–1388. IEEE, 2014.
3. Denis Charles, Kamal Jain, and Kristin Lauter. Signatures for network coding. *International Journal of Information and Coding Theory*, 1(1):3–14, 2009.
4. Zhen Yu, Yawen Wei, Barathram Ramkumar, and Yong Guan. An efficient signature-based scheme for securing network coding against pollution attacks. In *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*. IEEE, 2008.
5. Fang Zhao, Ton Kalker, Muriel Médard, and Keesook J Han. Signatures for content distribution with network coding. In *Information Theory, 2007. ISIT 2007. IEEE International Symposium on*, pages 556–560. IEEE, 2007.
6. Mea Wang and Baochun Li. How practical is network coding? In *Quality of Service, 2006. IWQoS 2006. 14th IEEE International Workshop on*, pages 274–278. IEEE, 2006.
7. Ming Li, Zhenyu Yang, and Wenjing Lou. Codeon: Cooperative popular content distribution for vehicular networks using symbol level network coding. *Selected Areas in Communications, IEEE Journal on*, 29(1):223–235, 2011.
8. Zhenyu Yang, Ming Li, and Wenjing Lou. Codeplay: Live multimedia streaming in vanets using symbol-level network coding. In *Network Protocols (ICNP), 2010 18th IEEE International Conference on*, pages 223–232. IEEE, 2010.
9. Uichin Lee, Joon-Sang Park, Joseph Yeh, Giovanni Pau, and Mario Gerla. 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*, pages 1–5. ACM, 2006.
10. Shabbir Ahmed and Salil S Kanhere. Vanetcode: network coding to enhance cooperative downloading in vehicular ad-hoc networks. In *Proceedings of the 2006 international conference on Wireless communications and mobile computing*, pages 527–532. ACM, 2006.
11. Peng Zhang, Yixin Jiang, Chuang Lin, Hongyi Yao, Albert Wasef, and Xuemin Sherman Shen. Padding for orthogonality: Efficient subspace authentication for network coding. In *INFOCOM, 2011 Proceedings IEEE*, pages 1026–1034. IEEE, 2011.
12. Bin Dai, Shijun Zhang, Yipeng Qu, Jun Yang, and Furong Wang. Orthogonal vector based network coding against pollution attacks in n-layer combination networks. In *Communications and Networking in China (CHINACOM), 2010 5th International ICST Conference on*, pages 1–5. IEEE, 2010.
13. Elias Kehdi and Baochun Li. Null keys: Limiting malicious attacks via null space properties of network coding. In *INFOCOM 2009, IEEE*, pages 1224–1232. IEEE, 2009.
14. Maxwell N Krohn, Michael J Freedman, and David Mazieres. On-the-fly verification of rateless erasure codes for efficient content distribution. In *Security and Privacy, 2004. Proceedings. 2004 IEEE Symposium on*, pages 226–240. IEEE, 2004.
15. Shweta Agrawal and Dan Boneh. Homomorphic macs: Mac-based integrity for network coding. In *Applied Cryptography and Network Security*, pages 292–305. Springer, 2009.
16. Chi Cheng and Tao Jiang. A novel homomorphic mac scheme for authentication in network coding. *Communications Letters, IEEE*, 15(11):1228–1230, 2011.
17. Sachin Katti, Dina Katabi, Hari Balakrishnan, and Muriel Medard. Symbol-level network coding for wireless mesh networks. In *ACM SIGCOMM Computer Communication Review*, volume 38, pages 401–412. ACM, 2008.
18. Tracey Ho, Muriel Médard, Ralf Koetter, David R Karger, Michelle Effros, Jun Shi, and Ben Leong. A random linear network coding approach to multicast. *Information Theory, IEEE Transactions on*, 52(10):4413–4430, 2006.
19. Daniel Jiang, Vikas Taliwal, Andreas Meier, Wieland Holfelder, and Ralf Herrtwich. Design of 5.9 ghz dsrc-based vehicular safety communication. *Wireless Communications, IEEE*, 13(5):36–43, 2006.
20. NS2.34. `http://www.isi.edu/nsnam/ns`.
21. VanetMobiSim. `http://vanet.eurecom.fr`.