





On Scalable Network Communication for Infrastructure-Vehicle Collaborative Autonomous Driving

SHENGAO YANG ¹ (Member, IEEE), HOOVER H. F. YIN ² (Member, IEEE),
RAYMOND W. YEUNG ^{2,3} (Fellow, IEEE), XI XIONG⁴, YANLONG HUANG ¹, LINGFEI MA¹,
MO LI¹, AND CHAOYANG TANG¹

(Invited Paper)

¹School of Science and Engineering, The Chinese University of Hong Kong, Shenzhen, Shenzhen 518172, China

²Department of Information Engineering, The Chinese University of Hong Kong, Hong Kong, SAR, China

³Institute of Network Coding, The Chinese University of Hong Kong, Hong Kong, SAR, China

⁴Shenzhen Research Institute of Big Data, Shenzhen 518172, China

CORRESPONDING AUTHOR: SHENGAO YANG (e-mail: shyang@cuhk.edu.cn)

The work of S. Yang was supported in part by the National Key R&D Plan of China under Grant 2022YFA1005000, in part by the National Natural Science Foundation of China under Grants 62171399 and 12141108, and in part by the Science and Technology Innovation Committee under Grant JCYJ20180508162604311. The work of R. W. Yeung was supported in part by a fellowship award from the Research Grants Council of the Hong Kong Special Administrative Region, China under Grant CUHK SRFS2223-4S03.

ABSTRACT Network communication is crucial for Infrastructure-Vehicle Cooperative Autonomous Driving (IVCAD), which can help to improve both driving safety and efficiency. Existing IVCAD systems either use RoadSide Units (RSUs) deployed sparsely in critical locations to extend the on-board sensing, or use RSUs deployed densely in a limited area to provide both sensing and decision-making for vehicles. For both scenarios, the numbers of RSUs and vehicles are small, and the network communication solutions are based on the existing wireline and cellular network infrastructures and their extensions. However, it remains unknown whether these existing network communication solutions are scalable when the IVCAD system is deployed city-widely with hundreds of thousands of RSUs and millions of vehicles. In this paper, we analyze the network communication requirements for different types of IVCAD systems, and show that when the numbers of RSUs and vehicles in the system are large, using the existing solutions to support IVCAD can incur very high extra network communication facility cost. We further discuss how to use wireless mesh networks formed by RSUs themselves to solve the network communication problem without any deployment of extra network communication facilities. To enable the RSU network, we provide a network communication framework based on network coding and highlight some research directions.

INDEX TERMS Cooperative autonomous driving, network communication, network coding.

I. INTRODUCTION

Roadside infrastructure can support autonomous driving by improving the sensing capability and robustness better than the on-board perception equipment [1], and can coordinate the driving decision among vehicles to improve the traffic efficiency [2]. Infrastructure-Vehicle Cooperative Autonomous Driving (IVCAD) is drawing a lot of research interest re-

cently [3]. Communications and networking technologies are crucial for achieving the advantages of roadside infrastructure for autonomous driving [4]. One of the notable feature of the roadside infrastructure is its large scale in terms of both the device numbers and the geographical distribution. In this paper, we share our thoughts on the network communication solutions for supporting advanced IVCAD systems on a large scale.

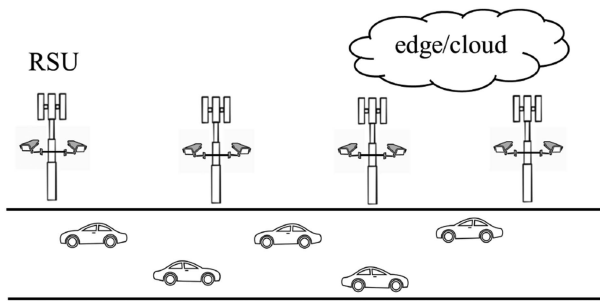


FIGURE 1. Illustration of an infrastructure-vehicle cooperative autonomous driving (IVCAD) system, which involves vehicles on the roads, RSUs on the roadside and the computing/storage devices on the edge/cloud.

A. RELATED COMMUNICATION AND NETWORK TECHNOLOGIES

We illustrate in Fig. 1 the major components involved in an IVCAD system. The roadside infrastructure is formed by RoadSide Units (RSUs) equipped with sensing, computation and communication devices. In addition to the vehicles on the road, an IVCAD system may also include some extra edge/cloud computing and storage units. The communication between vehicles and RSUs are necessary for infrastructure-vehicle collaboration, and there may be communication between the edge/cloud servers and RSUs for data processing and storage. In the following, we review some existing communication and networking technologies that can be used in an IVCAD system.

The existing wireline and the cellular network infrastructures can be extended to support the IVCAD systems. For example, RSUs can be connected by fibre to the Internet and then communicate with the edge/cloud servers. Vehicles can connect to the base stations of a cellular network and then communicate with the roadside infrastructure. However, the existing wireline and cellular infrastructures were primarily developed for Internet access, and whether it is feasible and efficient for new applications like IVCAD is in general unclear [5]. Existing experiments of infrastructure-vehicle cooperation based on these existing network communication infrastructures have reported the communication issues in terms of both bandwidth and latency [4]. Most of the existing systems are either the basic sensing extension type of IVCAD with RSUs deployed sparsely in critical locations, or are of very small scale formed by several of RSUs and tens of vehicles. More sophisticated IVCAD systems that consists of hundreds of thousands of densely deployed RSUs and millions of vehicles face more serious communication issues (see our analysis in Section II).

Dedicated vehicular communications have been considered as either an amendment of wireless LAN or a new feature of 4G/5G. To support direct vehicle-to-vehicle and vehicle-to-RSU communications, the dedicated short-range communication (DSRC) technology was developed based on the IEEE 802.11p amendment to IEEE 802.11 WLAN. The vehicle-to-vehicle/RSU feature (called C-V2X) was added

TABLE 1 Comparison of Different Types of Infrastructure-Vehicle Cooperation in Terms of Sensing and Decision Making Types and the RSU and Communication Requirements

Type	Sensing	Decision making	RSU deployment	Communication requirement
self-driving	on-board	on-board	no	no
sensing extension	on-board, with roadside assist	on-board	sparse	sensing data
roadside sensing	roadside, with on-board assist	on-board	dense	sensing data
roadside control	roadside	roadside	dense	sensing data, control signals

to LTE (in 3GPP Release 14 [6]) and is further improved in 5G in terms of ultra-reliable low-latency communication (URLLC) [7], [8] and side link communication [9]. The original purpose of the DSRC technology is to improve the driving safety and efficiency (such as car parking, lane changing and platooning) [10], which is not completely the same as for supporting autonomous driving. We will analyze further in this paper the role of these vehicle-to-vehicle/RSU communication technologies in IVCAD (see Section II). Except for some basic cases, these direct vehicle-to-vehicle/RSU communication techniques cannot provide a complete network communication solution for the whole IVCAD system.

B. PAPER CONTRIBUTION AND ORGANIZATION

In this paper, we study network communication solutions for supporting IVCAD systems. To make the basic network communication requirement clear, in Section II, we review how a roadside infrastructure can help autonomous driving in terms of sensing and decision making. We classify IVCAD systems in three types according to the role of the roadside infrastructure in terms of sensing and decision making, namely, sensing extension, roadside sensing and roadside control (see the summary in Table 1).

A network communication solution for IVCAD systems are discussed in terms of *the network framework* and *the network protocol* in Sections III and IV, respectively. The network framework is mainly about the layout of the physical communication devices and their connectivity, and the network protocol includes the operational principles and procedures.

In Section III, we analyze several network frameworks for the three types of the IVCAD systems. When the RSUs are deployed sparsely only in crucial locations, as required by the sensing-extension type of IVCAD, feasible network frameworks can easily be found. The major issue comes when the RSUs are densely deployed, as desired by the more advanced roadside sensing/control types of IVCAD. If the deployment area is relatively small, the network frameworks based on the wireline and cellular networks can still be feasible using the existing network infrastructures. However, when the deployment area increases to a city level with thousands and even

millions of RSUs/vehicles, the network frameworks based on the existing network infrastructures could incur unaffordable communication network deployment and maintenance costs.

We also discuss in Section III a network framework that does not depend on the existing wireline and cellular network infrastructures. In this network framework, the RSUs are densely deployed and nearby RSUs are connected by wireless communications to form a mesh network, called an *RSU network*, which provides the roadside infrastructure services to vehicles. This network framework is deployed together with the RSU deployment and hence does not incur extra communication network deployment and maintenance costs. We showed that the RSU network can be used with other network frameworks to provide a scalable network communication solution for the IVCAD systems.

To complete the solution, however, we still need a feasible network protocol for the RSU network, which is mainly formed by wireless communication links without a wireline backbone. In Section IV, we first analyze the challenge of using existing network protocols based on the TCP/IP principles. As mainly designed for networks with a wireline backbone, the basic principles of TCP/IP (like store-and-forward and end-to-end congestion control) assume that the network links are mostly of high reliability and fixed latency. As the existing cellular network (5G) is designed based on IP, it is not suitable for the RSU network.

Network coding can outperform the store-and-forward mechanism of TCP/IP at the intermediate nodes [11], [12], [13]. Theoretically, random linear network coding (RLNC) achieves the capacity of wireless networks with packet loss for a wide range of scenarios (see e.g., [14], [15], [16], [17], [18]), and hence is suitable for the RSU network. In Section IV-IV-B, we discuss a class of efficient network coding schemes called *batched network coding* (see [19], [20], [21] and the reference therein) for the RSU network. Compared with the baseline RLNC schemes, batched network coding can achieve close-to-optimal end-to-end reliable communication with a low computation and coding overhead [19], [21]. Batched network coding can provide a general approach for designing network communication protocols [22].

Based on batched network coding we further discuss how to design a cellular-like network protocol to make the RSU network feasible for a large scale IVCAD. Roughly speaking, we need to reconsider the design of 5G using the network coding approach. Moreover, we should take the properties of the RSU network into consideration. In Section IV-IV-C, we discuss how to combine the various parts of 5G with batched network coding into a feasible network protocol for the RSU network, and research issues towards improving the overall system performance.

Last, our concluding remarks are in Section V.

II. AUTONOMOUS DRIVING AND ROADSIDE INFRASTRUCTURE

For *self-driving*, we refer to autonomous driving with only on-board sensing and decision-making [23]. Though self-driving

technologies have made significant progress over the past decade, self-driving has some intrinsic limitations. In this section, we first briefly summarize the limitations of self-driving vehicles, including sensing and decision making, and then discuss how a roadside infrastructure can help to resolve these limitations. The purpose here is to understand the network communication requirements for infrastructure-vehicle cooperative autonomous driving (IVCAD).

A. LIMITATIONS OF SELF-DRIVING TECHNOLOGIES

An existing self-driving vehicle tries to emulate the human driving capability based on the on-board perception and decision-making devices [23]. This approach of autonomous driving, however, faces fundamental limitations in terms of safety and efficiency.

As all the sensors are equipped on the vehicle, the perception capability of a self-driving vehicle has a limited range and can be blocked by other objects such as vehicles, buildings and trees [1]. Various sensing technologies, such as camera, ultrasonic sensor, Radar and LiDAR, have been used for self-driving vehicles [24], [25]. Each sensing technology has its own limitations in terms of range, angle, resolution etc., so that most self-driving cars are equipped with multiple types of sensors at different positions to compensate for these limitations. However, all these sensing technologies are vulnerable to weather conditions such as rain, snow and fog [25]. Moreover, these technologies need line-of-sight for accurate sensing, and hence if blocked by other objects, the sensing range and angle can be significantly reduced [24]. Due to the complicated road scenarios, compromised sensing capability is not avoidable using only sensing devices on a vehicle in many real-world scenarios.

From the decision-making point of view, a self-driving vehicle tries to control its own driving efficiently, but such a mechanism does not enable the cooperation among vehicles. In transportation science, the junction level and network level transportation managements deal with cooperation among autonomous driving vehicles [26]. The cooperative driving at the junction layer, i.e., vehicle sequencing at intersections, can yield better efficiency and environment benefits than vehicles making their own decisions [27]. As for the network layer, the coordination between autonomous vehicles can facilitate a large-scale route planning to improve the city-wide traffic efficiency [28].

There are many ongoing researches towards overcoming these limitations of the existing self-driving technologies. A system-level approach is to use roadside infrastructure to support autonomous driving.

B. ROADSIDE INFRASTRUCTURE FOR AUTONOMOUS DRIVING

Roadside infrastructure, such as traffic lights and signs, has long been used to improve the efficiency and safety of human driving. To support autonomous driving, however, a new infrastructure formed by intelligent devices with proper sensing, computation and communication capabilities is required [29],

[30], [31]. Here, an intelligent device located at the roadside is called a RoadSide Unit (RSU). Equipped with sensing and computation devices, an RSU can detect and track road objects. With its communication capability, an RSU can also exchange information with the road vehicles. A roadside infrastructure formed by RSUs can potentially overcome the limitations of self-driving by the individual vehicles.

Roadside infrastructure can compensate for the weaknesses of the on-board sensing capability of self-driving vehicles [32]. It is not necessary that one RSU covers a big area. Multiple ones can work together to cover different aspects of a scene, and the views of different RSUs can have overlapping regions to gain sensing reliability. The placement of RSUs can be carefully planned to provide a robust omnidirectional street view. For example, on a road where a large number of cars are parked on both sides, pedestrians often pass through the gap between cars that cannot be seen by the driving vehicle, which can easily lead to accidents. In this case, some RSUs can be deployed in a higher position to avoid the blocking by cars, to provide a bird's eye view to the road and to warn the driving vehicles on the road. In particular, perception fusion between the driving vehicle and the roadside infrastructure has been studied [33], [34].

Roadside infrastructure can also assist the cooperation of vehicles for the junction and network level of traffic management. Roadside infrastructure can help to create a network of vehicles around an intersection with a higher stability than the network formed by only vehicle-to-vehicle communications, and hence improve the efficiency and safety of autonomous driving for passing the intersection [35]. Moreover, the roadside infrastructure can collect the road information and the vehicle driving information in a large area and facilitate the network-level cooperation in complex traffic networks [28], [36]. Furthermore, autonomous vehicles can make ethical judgments with more information from the roadside infrastructure to incorporate morality demands under dilemma scenarios [37].

The promising application of roadside infrastructure for autonomous driving has attracted many research interest, where topics like perception [1], edge computing [38], and security [39] have been extensively discussed. In this paper, we study the network communication solutions for IVCAD.

C. RSU DEPLOYMENT AND COMMUNICATION REQUIREMENT

A real-world IVCAD system could be very complicated and has many functions. If we focus on a specific case of IVCAD, our network communication solution may not be suitable for many other cases. If we discuss IVCAD in general, the high-level solution would be of little guidance value. To make a good balance, we discuss three basic types of IVCAD systems:

- The first type is called *sensing extension*. The roadside infrastructure is mainly used to extend the vision of the self-driving vehicles in critical locations where the view

of cars can be blocked. Self-driving vehicles can still drive by themselves without the roadside infrastructure.

- The second type is called *roadside sensing*. Autonomous vehicles have a limited sensing capability and depend on the roadside infrastructure for sensing the road scene. However, the driving decision-making is mainly done on the vehicles.
- The third type is called *roadside control*. Autonomous vehicles have a limited sensing and computing capability, and depend on the roadside infrastructure for both sensing and decision-making. Vehicles only execute the driving instructions from the roadside infrastructure.

A real-world IVCAD system may combine the functions of the all these types. These three types of IVCAD systems have different requirements to the roadside infrastructure. To illustrate the RSU deployment requirement, we use as example a city consisting of 10,000 kilometers of road and 5,000 traffic intersections.

The sensing-extension IVCAD system can be adopted by the existing self-driving vehicles, and it does not depend on a heavy deployment of RSUs. RSUs are only needed at the critical locations such as intersections [40], which is also called the *sparse* deployment of RSUs. Suppose 4 RSUs are required at each intersection. The city needs 20,000 RSUs. To assist autonomous driving, an RSU collects the road information in real-time and shares this information to the self-driving vehicles nearby to extend their vision. The RSU can also transmit this information collected to an edge/cloud server for processing before transmitting to vehicles. The information from the RSUs are complementary, the vehicles can drive by themselves without the roadside information.

The roadside sensing/control types of IVCAD systems mainly depend on the sensing capability of the roadside infrastructure. Therefore, the deployment of the RSUs should be dense enough to cover all the road scene. For example, we may need an RSU every 20 meters in an area with complicated road scenario, i.e., about 100 RSUs per kilometer, considering both sides of the road. Such roadside sensing/control IVCAD systems heavily depend on the deployment of RSUs, and have been used in special areas with the roadside infrastructure properly deployed, such as industrial parts and harbors. To cover all the 10,000 kilometers of road in the city we consider, one million RSUs are required, which is 50 times the number of RSUs deployed for sensing extension.

To reduce the communication cost, the sensing-extension and roadside sensing types of IVCAD may also require the sensing data to be processed at edge/cloud servers before transmitting them to the vehicles [41]. The sensing data of a road segment is common for all vehicles on the road, and hence it is not necessary to customize the data for each vehicle. For the roadside control IVCAD, the sensing data can be used to plan the driving of the vehicles collaboratively. Roadside control IVCAD can improve the traffic efficiency more than self-driving, where the autonomous vehicles decide their driving individually. However, the driving instructions

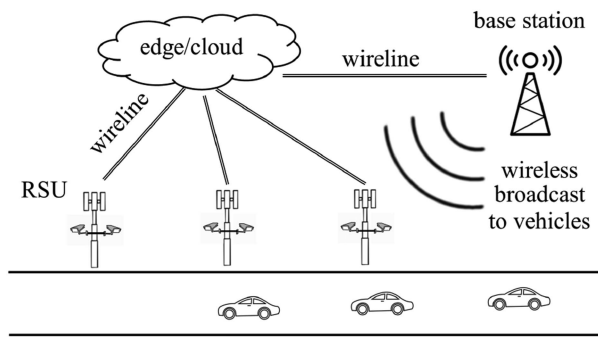


FIGURE 2. Illustration of Framework 1. This network framework needs the support of the existing wireline and wireless network infrastructures. It can only be used in locations with cellular network coverage. Otherwise, new cellular network deployment is necessary. Wireline network deployment is required for each RSU to connect to the existing network infrastructure.

are generated for each individual vehicles and should be transmitted to the particular vehicles on time.

All these three types of IVCAD systems require the transmission of the sensing data from RSUs to the vehicles. The roadside sensing/control IVCAD systems also have higher communication requirement due to the higher RSU density. The roadside-control-type of IVCAD further requires the communication of the control signals between vehicles and the roadside infrastructure. In Table 1, we summarize the comparison of the different types of infrastructure-vehicle collaboration with self-driving.

III. ANALYSIS OF BASIC NETWORK FRAMEWORKS

A network communication solution for IVCAD systems are discussed in this section and Section IV. This section focuses on the *network framework* that includes the layout of the physical communication devices and their connectivity. In addition to the communication feasibility, the network framework design should take the deployment and maintenance feasibility into consideration [42]. The *network protocol* includes the operational principles and procedures, and will be discussed in Section IV.

We discuss four basic network frameworks with respect to their feasibility for various IVCAD applications.

A. WIRELINE-CONNECTED RSUS

We first consider a network framework that uses wireline network to connect the RSUs, which is illustrated in Fig. 2.

Framework 1: RSUs are connected by wireline cables to the Internet, and hence to the edge/cloud servers. Vehicles are connected wirelessly to the base stations of a cellular network. The information collected at an RSU is first processed by the edge/cloud server and then transmitted to the vehicles nearby by a base station.

This network framework depends on both the wireline broadband network infrastructure and the cellular network infrastructure. Wireline network deployment must come with almost every new RSU, which incurs an extra deployment and

maintenance cost of the roadside infrastructure in addition to the RSU cost. The extra cost of deploying a wireline network to support RSUs may not be significant when building new roads, but the extra cost and time for deploying a wireline network for renovating an existing road to install RSUs could be a major concern. Moreover, for locations where wireline deployment is not feasible, this network framework is not valid. Once properly deployed, due to the dedicated wireline network connection, the communication bandwidth of the RSUs to the edge/cloud servers can be guaranteed.

The coverage of the cellular network is necessary so that the information collected by the RSUs can be transmitted to the vehicles. The cellular network communication load depends on the type of the IVCAD system:

- For the sensing extension and the roadside sensing types of IVCAD, as all the vehicles near an RSU require the same information, the broadcast nature of wireless communication can be applied to improve the efficiency: It is not necessary that the base station transmits the same information to each vehicle one by one. The communication load of a cellular base station depends on how many RSUs it serves, but not on the number of vehicles.
- For the roadside control IVCAD, the roadside infrastructure generates dedicated driving instructions for each vehicle jointly. Therefore, the communication load of a cellular base station depends on how many vehicles it serves, but not on the number of RSUs.

The existing cellular network is not dedicated for vehicular applications. The network communication resources of a cellular network must be shared between the IVCAD systems and the other users. Though it may be possible to guarantee a certain communication latency and bandwidth using the network slicing technique for the IVCAD systems, the scale of the IVCAD systems is still limited by the capacity and the coverage of the cellular network.

Framework 1 is suitable for the sensing extension IVCAD where RSUs are only sparsely deployed at the critical locations such as intersections [40]. For the early stage deployment of the roadside infrastructure, maybe at most one or two RSUs are covered by one base station. Note that even for this case, whether the end-to-end communication latency and bandwidth from the RSUs to the vehicles can satisfy the requirement or not depends on many other factors of the network framework, such as the location of the edge/cloud servers and their data processing capability, which are beyond the scope of this paper.

In contrast, for the roadside sensing/control IVCAD, more RSUs must be deployed in one area in order to provide full perception information, including not only the objects on the road but also the pedestrians in the blind corners. Therefore, considering the wireline network deployment cost and the cellular network resource, this network framework is suitable for roadside sensing/control in a relatively small dedicated area with a relatively small number of RSUs and vehicles.

With the development of IVCAD, more RSUs can be deployed and more vehicles are needed to be served. In addition

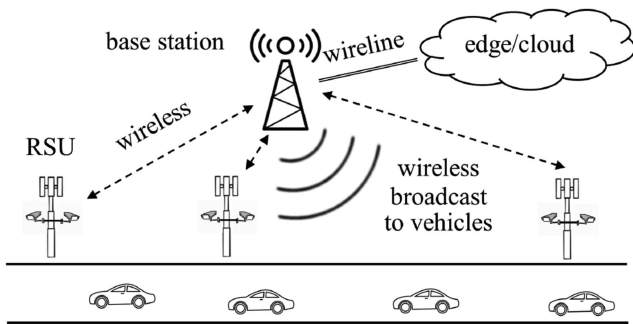


FIGURE 3. Illustration of Framework 2. This network framework mainly depends on the wireless network infrastructure, and can only be used in the locations with cellular network coverage. Otherwise, new cellular network deployment is necessary.

to the wireline network deployment, this network framework require more communication resource of the cellular network. One possible solution to increase the capacity of a base station is to use millimeter wave, which incurs further deployment cost of cellular networks (to be elaborated in the next subsection). Therefore, this network framework has a high cost for the deployment of both the wireline network and wireless network when the IVCAD systems become large.

B. CELLULAR-CONNECTED RSUS

One of the major hurdles of Framework 1 is the wireline network connecting to each RSUs. The second framework reduces the requirement of wireline network by allowing RSUs to connect to the base stations of cellular networks, illustrated in Fig. 3.

Framework 2: Both RSUs and vehicles are connected to the base stations of a cellular network. The information collected at an RSU is first transmitted via the base station to an edge/cloud server for processing, and then transmitted to the vehicles nearby.

In Framework 2, the RSUs first connect to the base station of a cellular network and then communicate to the edge/cloud servers through wired links. How the information is transmitted from the servers to the vehicles is the same as in Framework 1. Framework 2 does not need to deploy new wireline network if the existing cellular network can cover the locations of RSUs. For areas without the coverage of the cellular network, deploying new wireless networks is necessary. As it is not required to deploy wireline to each RSUs, the deployment cost can be lower than the wireline solution in Framework 1.

Although long-range, low-bit rate communication technologies such as LoRa and ZigBee are common in sensing and actuation applications, the bandwidth required in IVCAD suggested in [4] is already way beyond the capability of these technologies. Therefore, we focus on the cellular network in the following discussion.

Compared with Framework 1, Framework 2 increases the cellular communication cost for uploading the information

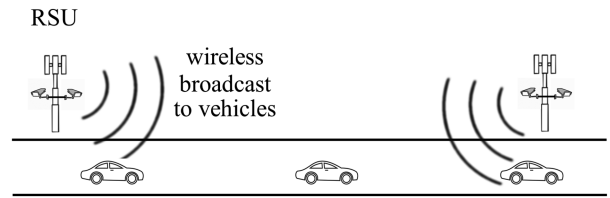


FIGURE 4. Illustration of Framework 3. This network framework does not need the support of any network infrastructure. It only uses the native wireless communication capability, such as C-V2X or DSRC, of an RSU.

collected by RSUs. As each RSU has their own information to upload and different RSUs do not have collaboration before connecting to the base stations, the uploading bandwidth requirement increases linearly with the number of RSUs served by one base station. For example, in the experiment in [4], each RSU has the uploading bandwidth of about 12 Mbps. The sub-6 GHz 5G can deliver 1,000 Mbps, i.e., one base station can support at most 80 RSUs in one cell. However, the bandwidth of 5G needs to be shared by other applications, and hence the practical number of RSUs can be supported in one cell is much smaller.

By using millimeter wave (mmWave) bands, 5G can deliver up to 4 Gbps. However, the mmWave bands are limited to short distance and line-of-sight communications. The existing coverage of mmWave 5G is still limited. To benefit from the high bandwidth of mmWave, an extensively new wireless network infrastructure must be built, which may further require new wireline network deployment.

Framework 2 is suitable for the sensing extension IVCAD where each base station covers only a small number of RSUs. It can also be used for roadside sensing/control in a relatively small area with tens of RSUs in one cellular network cell. To support a high density deployment of RSUs in a large area, it is necessary to deploy new wireless network infrastructure, which incurs a significant deployment cost for the IVCAD system.

C. INDEPENDENT RSUS

Framework 2 reduced the wireline deployment cost of Framework 1, but incurs a higher deployment cost of wireless network infrastructure to support a large scale IVCAD system with the roadside sensing and control capability. Now we consider a network framework without new network infrastructure deployment, illustrated in Fig. 4.

Framework 3: An RSU and a vehicle can communicated directly using communication techniques such as C-V2X or DSRC. The information collected at an RSU is first processed by the RSU itself and then transmitted to the vehicles nearby.

The major advantage of Framework 3 is that it does not depend on the existing network communication infrastructure, nor deploy new networking devices in addition to the RSUs. It uses the communication capability of an RSU to directly transmit the information to nearby vehicles. As the bandwidth of V2X communications (like DSRC and C-V2X)

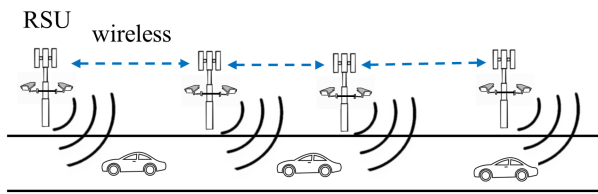


FIGURE 5. Illustration of Framework 4. This solution does not need the support of any network infrastructure. Instead, the RSUs form a network by themselves and jointly provide the services to vehicles.

are dedicated, the communication bandwidth and latency can be guaranteed. Using Framework 3, an RSU must employ a sufficient information processing capability, as it cannot benefit from the edge/cloud servers.

Framework 3 provides a scalable solution for the sensing extension IVCAD. In each location that sensing is required for self-driving vehicles, an RSU can be installed. Compared with Framework 1 and 2, Framework 3 is the most efficient in deploying RSUs on a large scale for sensing extension. However, as the effective communication range of an RSU is limited by both the media and the environment, one drawback of this framework is that the location of an RSU may not be jointly optimized for both sensing and communication: a good location for monitoring the road scene may be blocked for transmitting the monitoring data to some vehicles.

Framework 3, however, cannot provide a good solution for roadside sensing and control IVCAD systems, which require the sensing data from multiple RSUs to be jointly processed. In Framework 1 and 2, the data from multiple RSUs can be joint process at the edge/cloud server before transmitting to the vehicles, which may help reduce the data for communication and the computation cost at the vehicles. In Framework 3, however, as there is no communication between RSUs, the joint processing of data is not possible. Using Framework 3 for roadside control, each RSU has to make decision by itself and hence cannot resolve the issues of self-driving vehicles.

Using Framework 3 for roadside sensing, a vehicle must receive the data from all the nearby RSUs and processes all the data it received efficiently for making the driving decision. As RSUs cannot jointly process the data to reduce the communication cost, Framework 3 imposes a critical communication and computation requirement for the vehicles. However, due to the mobility of the vehicles, this communication requirement is difficult to be fulfilled.

D. SELF-CONNECTED RSUS

To compensate for the issues of using independent RSUs, we provide an enhancement of Framework 3 by allowing communications between RSUs, illustrated in Fig. 5.

Framework 4: Direct wireless communication exists between nearby RSUs, and all the RSUs form a communication network (called an RSU network) to provide roadside infrastructure services to vehicles. The information collected at the

RSUs can be processed by themselves, and then transmitted to the vehicles.

This network framework inherits the same advantage of Framework 3 that it does not depend on the existing network communication infrastructures. Instead of serving the vehicles individually as Framework 3, all the RSUs form a network using their wireless communication capability. In other words, the RSUs form a network infrastructure by themselves to serve the IVCAD system, where the computation resources of the RSUs can form a distributed computing platform for sensing data processing and decision making. It is not necessary to deploy any extra wireline and wireless network infrastructures to support the IVCAD system using Framework 4. Therefore this network framework is scalable for a large scale IVCAD system.

The major challenge of Framework 4 is how to form a robust multihop wireless network by RSUs that can support the IVCAD systems. We will discuss the communication and networking techniques to support this RSU network in Section IV. Here, we suppose that the communication between the vehicles and the RSU network is similar to the wireless access of cellular network, where the RSUs play a similar role as the base stations.

We discuss how to use the RSU network to support the IVCAD systems. When the RSUs are deployed far away from each other so that direct communication between RSUs is not possible, Framework 4 degrades to Framework 3, and hence can support the sensing extension of IVCAD as the latter. However, Framework 4 can have extra benefits when some of the RSUs are close enough to communicate with each other. The sensing information of an RSU now can also be transmitted to vehicles covered by other RSUs via the RSU network. Roughly, more widely the information is broadcast, the safer the driving can be.

The RSU network can make use of the distributed computing capabilities of the RSUs to process the sensing data from the nearby RSUs. Compared with processing the data at the edge/cloud servers, processing the data at local RSUs significantly reduces the data communication latency. For roadside sensing, the processed road scene data will be broadcast to nearby vehicles for making the driving decisions. For roadside control, the processed road scene data will be used to further plan the driving of the vehicles collaboratively, and the driving control messages will be transmitted by the RSU network to the vehicles.

In Table 2, we summarize the general properties of the network frameworks we have discussed. These basic network frameworks can be used together to support the IVCAD systems, as illustrated in Fig. 6. In an RSU network, most RSUs are self-connected, and a small number of RSUs are connected to the base stations or by wireline cables. Moreover, a small number of RSUs in the RSU network may be connected by wireline or cellular to some edge/cloud servers for better planning of the traffic in a large area. No matter how we combine these network frameworks, the self-connected

TABLE 2 Comparison of the Basic Network Frameworks

Framework	infrastructure dependency	deployment in addition to RSUs	feasible location	sensing extension	roadside sensing	roadside control
1: wireline	wireline broadband, cellular network	wireline network	wireline reachable, cellular coverage	support if RSUs are sparse, scalable	small scale only	small scale only
2: cellular	cellular network	no	cellular coverage	support if RSUs are sparse, scalable	small scale only	small scale only
3: independent	no	no	everywhere RSUs can be deployed	support, scalable	not support	not support
4: self-connected	no	no	everywhere RSUs can be deployed	support, scalable	support, scalable	support, scalable

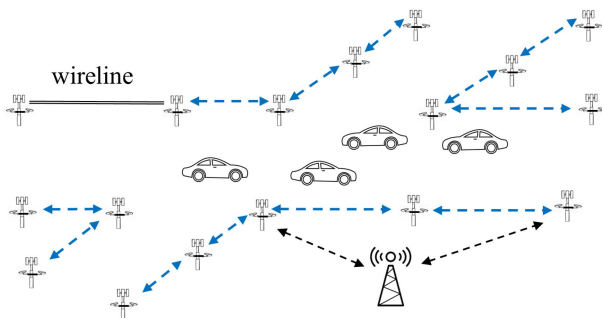


FIGURE 6. Combination of all the basic network frameworks.

RSU network is the crucial component to make the system scalable.

IV. NETWORK PROTOCOL FOR RSU NETWORKS

The first two frameworks in the last section are extensions of the wireline network and cellular network, and hence can be supported by the TCP/IP protocol stack together with 5G. The framework of independent RSUs can be supported by point-to-point communications like DSRC and C-V2X. In this section, we focus on the network communication solution for the RSU network discussed in Section III-III-D, which is also the crucial part to make the network framework scalable.

In the remainder of this section, we first discuss why the TCP/IP and 5G combination is not suitable for the RSU network. We then show that network coding should be employed in the network protocol for the RSU network. Last, we present the research issues towards a network protocol to make the RSU network similar to a cellular network so that it can serve the vehicles on the road for IVCAD systems.

A. CHALLENGE AND SOLUTION

As illustrated in Fig. 5, the RSU network includes RSUs and vehicles as major components. The RSUs form a wireless network that provides the roadside infrastructure services to the vehicles, and the vehicles access the roadside infrastructure services wirelessly. The RSU network is a multihop wireless network, in contrast to the traditional cellular network that is mainly formed by base stations connected by a dedicated

wireline backbone. Multihop wireless network is generally considered a challenging problem due to the unreliable wireless links [43]. It is expected that the communication rate decreases and the communication latency accumulates hop by hop. Due to the dense deployment, however, the RSUs network we consider may have tens or even more than a hundred hops. Therefore, to enable the RSU network, a network communication protocol that can support efficient multihop wireless communication is essential.

The design of the Internet assumes that the network links are of high reliability and fixed latency. The wireline communication media such as twisted-pair cable and optical fiber can satisfy these requirements. The routing and congestion control mechanisms of the Internet, as defined in the TCP/IP protocol stack, are designed based on these assumptions. Both WiFi and 5G are designed as a wireless access network for the Internet, and hence are required to satisfy the high reliability and fixed latency requirements. Under the Internet, these wireless links are assumed to be reliable communication links similar as wirelines.

Though WiFi and 5G have adopted multihop wireless communications for extending the coverage of wireless access, the wireless access role of wireless networks is not changed in the Internet. In the existing design of WiFi mesh and 5G integrated access and backhaul (IAB) (defined in 3GPP Release 16 [44] and enhanced in Release 17 [45]), multiple concatenated wireless links are encapsulated as a single reliable link to be compatible with the existing network protocol. However, both the 802.11 physical layer and 5G NR are designed for reliable wireless access, i.e., one-hop communication. Simple concatenation of such wireless communication techniques cannot achieve a good multihop performance in terms of either throughput or latency [22]. This issue is not a concern in most existing adoptions when there are at most a couple of wireless hops. However, this is not the case in the aforementioned Framework 4 of RSU networks.

A general solution to multihop wireless networks is network coding, which enables the coding of packets at the intermediate network nodes [11], [12], [13]. Random linear network coding (RLNC) [16] achieves the capacity of wireless networks with packet loss for a wide range of scenarios [14], [15], [16], [17], [18]. See an illustration of RLNC in Fig. 7.

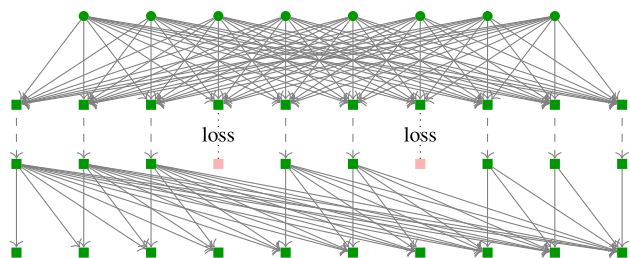


FIGURE 7. An illustration of the baseline RLNC scheme. The first two layers of nodes represent the random linear encoding process at a source node. Each coded packet in the second layer (represented by a square) is a random linear combination of the input packets in the first layer (represented by the circles). The source node transmits the coded packets to an intermediate node, and the third layer nodes are the coded packets received at the intermediate node. Packet loss occurs during the transmission so that the intermediate node may not receive all the packets transmitted by the source node. The bottom layer nodes are the packets generated by network coding at the intermediate node using random linear combinations of the packets it has received. As the network coding coefficients are random, a coefficient vector is usually attached with each packet so that the linear network coding transfer matrix can be recovered for decoding.

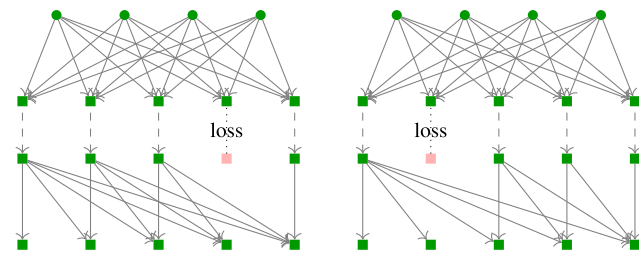


FIGURE 9. An illustration of generation-based RLNC. The basic encoding and network coding are similar to that of the based line RLNC (see Fig. 7). The input packets are grouped into smaller generations to reduce the computation cost and the coefficient vector overhead.

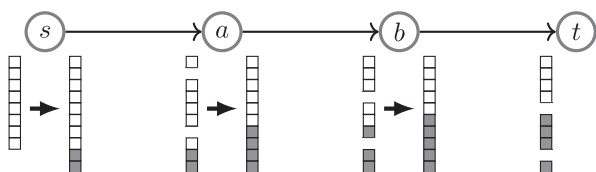


FIGURE 8. An example of network coding in a network formed by four nodes s , a , b and t , which form a line topology. Here the network links are assumed to be directional as we only consider the communication from node s to node t . The blank squares are the original packets and the gray blocks are the coded packets. The source node s generates 2 coded packets and transmits 10 packets in total. Node a receives 8 packets from node s and generates two extra coded packets.

Existing works have studied the used of RLNC for vehicle related communications, such as content distribution [46], data offloading [47] and platooning [48].

We use the example in Fig. 8 to illustrate the advantage of network coding for networks with unreliable links. Suppose the source node has 8 input packets for communication. Assume that each hop loses exactly 2 out of 10 consecutive packets. We can design the following network coding scheme: the source node generates two coded packets by taking random linear combinations of the 8 input packets. All the 10 packets are transmitted by the source node to the first intermediate node. According to the link assumption, the first intermediate node receives 8 packets. The intermediate node does not perform any decoding operations, but instead generates two coded packets by taking random linear combinations of the 8 received packets. The same operation is performed by the second intermediate node, so on and so forth. The destination node also receives 8 packets, which are linear combinations of the original 8 input packets. The linear transformation of the 8 input packets from the source node to the destination node can be conveyed by the coefficient vectors transmitted with each packet. If network coding uses a

large finite field, the linear transformation is of full rank with a high probability and hence the original 8 input packets can be decoded.

In the above example, for any number of hops with the same packet loss model, the destination node can always receive 8 packets. So this network coding scheme achieves the optimal rates for the loss model assumed. However, a practical network would have a random loss pattern, and in general, the linear transformation is not always full rank. Moreover, when the number of input packet is large, the baseline RLNC also incurs a cost of large coefficient vector overhead and high computation and storage requirements, which limits its adoption in a general communication environment. Over the past twenty years, extensive researches have been done towards solving various implementation issues of RLNC, such as reducing the delay [49], [50], [51], [52], [53], [54], selecting a suitable packet size [55], [56], [57], [58], [59], and reducing the coefficient vector overhead [60], [61], [62]. A general RLNC framework called *batched network coding* is gradually formed, which includes the baseline RLNC scheme as a special case, and can achieve efficient implementation in terms of achievable rate, coefficient overhead and computation cost [19], [20], [21], [63], [64], [65], [66], [67], [68].

In the remainder of this section, we introduce batched network coding, and discuss how to achieve efficient multihop wireless network communication for RSU networks using batched network coding.

B. BATCHED NETWORK CODING

To reduce the computation cost and the coefficient vector overhead, generation-based RLNC [63] partitions the packets to be transmitted into multiple disjoint subsets (also called generations) and applies RLNC to each generation separately. See an illustration of generated-based RLNC in Fig. 9. By using a small generation size, generation-based RLNC significantly reduces the cost and coefficient vector overhead of the baseline RLNC. However, as each generation is decoded individually, using a small generation size incurs the generation scheduling issues [63], [69].

Besides applying RLNC to each disjoint subsets of the packets, overlapped subsets of packets were also investigated [20], [64], [65], [66]. More sophisticated approaches

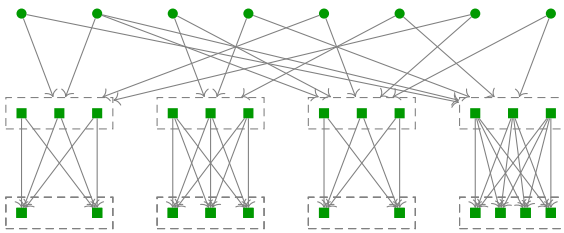


FIGURE 10. An illustration of batched network coding. The first two layers illustrate the outer code, where 4 batches are generated using 8 input packets. Each batch has 3 packets, i.e., the batch size is 3. The bottom two layers illustrate the end-to-end operation of the inner code, where each batch is encoded by network coding individually.

apply codes such as LDPC [21], [68] and generalized fountain code [19] on the packets to generate a sequence of subsets of coded packets, and apply RLNC to each subset of coded packets separately. A subset of coded packets is referred to as a *batch* and these approaches are collectively called *batched network coding*, where the code for generating the batches is called the *outer code* and the RLNC applied to each batch is called the *inner code*. See Fig. 10 for an illustration of the outer-code-inner-code structure of batched network coding. For multihop wireless communications, batched network coding provides a scalable solution with a flexible tradeoff between achievable rate and computation cost [22], [70], [71].

Batched sparse (BATS) code is a class of batched network codes using a generalized fountain code as the outer code, and achieves the nearly optimal outer code performance [19], [67], [72]. The outer code of a BATS code preserves the features of the fountain code, such as low encoding/decoding complexity and the rateless property, i.e., the number of batches can be generated is unlimited. To facilitate our further discussion, we use BATS code as an example to show how to use batched network coding for network communication. Most of our following discussion applies to other batched network codes as well except for the outer code part. We introduce the basic usage of BATS code for communicating from a source node to a destination node connected by multiple intermediate nodes forming a line topology. The outer code of a BATS code is a matrix-generalized fountain code. At the source node, the message consisting of K input packets is encoded by the outer code into a sequence of batches, each of which consists of M packets. The value M is called the *batch size*. The number of input packets, K , can be optimized via approaches such as [73]. Fix a finite field (e.g., $GF(256)$), called the base field. Regard a packet as a column vector with symbols from the base field. A batch is generated using the following procedure:

- 1) Sample a degree distribution to obtain an integer d .
- 2) Uniformly at random choose d input packets. Let B be the d column matrix formed by the d chosen input packets.
- 3) Generate M linear combinations of the chosen packets using coefficients obtained uniformly at random from the base field. Denote by X the M column matrix formed

by the M packets of a batch generated. Then $X = BG$, where G is a $d \times M$ matrix with entries uniformly at random from the base field.

More batches can be generated by repeating the above procedure. The outer code illustrated in Fig. 10 has $K = 8$ and $M = 3$, and the four batches have degrees 4, 3, 5 and 5, respectively. The degree distribution can be obtained via different models for different objectives [74], [75], [76], [77]. Further, Steps 1 and 2 can be combined and solved via machine learning [78].

Under certain conditions, it is provable that a constant batch size is an optimal choice [79]. In any case, for the ease of implementation and analysis, most works on BATS codes use a constant batch size. When $M = 1$, the outer code becomes a fountain code [80]. The advantage of using a batch size larger than 1 is to allow the application of network coding to the packets belonging to the same batch, i.e., enabling the inner code. In general, a different number of packets can be generated by the inner code for different batches [81], [82], [83], [84].

The end-to-end operation on a batch using linear inner code is a matrix multiplication channel given by

$$Y = XH \tag{1}$$

where X is an M -column matrix formed by the M packets of a batch generated by the source node; H is called the batch transfer matrix which can have a random number of columns; and Y is formed by the received packets of the batch at the destination node. The batch transfer matrix can be obtained from the coefficient vectors. In the general case, H is not necessarily of rank M as in the example illustrated in Fig. 8. When H has a general distribution, the matrix multiplication channel formulated in (1) has been studied for linear network coding [85]. When the channel is memoryless, its capacity is the expected rank of H . From this perspective, the outer code of a batched network code is a channel code for the matrix multiplication channel. Multiple batches should be decoded jointly to achieve the channel capacity. For a BATS code, the fountain-like outer code has an efficient belief propagation decoding algorithm, and achieves rates very close to the capacity of the matrix multiplication channel.

In practice, the batch size is taken to be a small number such as 8 or 16 to provide a good balance between the computation complexity and the achievable rates. When the batch size is 1 and each intermediate node only forwards the packets it has received, the inner code degenerates to the store-and-forward strategy. Suppose that each hop has a packet loss rate ϵ . The end-to-end achievable rate for L -hop communication using store-and-forward is $(1 - \epsilon)^L$, i.e., the rate decreases exponentially fast when the number of hops increases. Using a larger batch size and proper inner coding, a BATS code can achieve rates very close to the cut-set upper bound of the network capacity [22], [70], [71].

As a brief summary, BATS code provides an efficient approach for network communication with unreliable links, and

achieves end-to-end reliable communication at a nearly optimal rate. Therefore, BATS code is very suitable for the RSU network, which is mainly formed by unreliable wireless links and has communications through a long concatenation of wireless links. A practical network has multiple users sharing the network resources, for which the network utility maximization [86] and a network coding design [87] have been studied for multiple BATS code communication flows.

C. RESEARCH TOWARDS COMPLETE NETWORK PROTOCOL

A general approach about using batched network coding to design a network communication protocol has been discussed in [22], [88]. However, towards designing a network protocol for the RSU network, there are still specific research issues related to the properties of the RSU network that need to be resolved. At a high level, we hope to design a network protocol so that the RSU network behaves like a cellular network, where the RSUs play two roles:

- The RSUs provide the access services to the vehicles; and
- The RSUs form the transport network to support the wireless access services.

We can emulate 5G to design the network protocol for the RSU networks based on batched network coding. However, when comparing with the 5G cellular network, the RSU network has some special properties that need to be considered. As the RSUs are mainly deployed along the roads, the topology of the RSU network can be roughly estimated according to the road map. For example, the RSUs on a long road should form a line-type topology, while the RSUs around the intersection of multiple roads should form a grid topology. More importantly, in addition to communication, an RSU has the sensing and computation capability. To support IVCAD, the RSU network must be able to detect and track objects on the road. These intelligent capabilities are not assumed for the base stations in a traditional cellular network, but can help the network communication in the RSU network.

Though the study of specific research issues is beyond the scope of this paper, we discuss the special properties of some research topics in the remainder of this section. Note that our objective is not to give a comprehensive review of the research topics, but to highlight the ones closely related to batched network coding and the RSU network.

1) RATIO ACCESS

The lower RLC, MAC and PHY layers defined in 5G NR are mostly compatible with batched network coding, and hence can be mostly adopted by the RSU network protocol for both RSU-to-RSU and vehicle-to-RSU wireless communications. In particular, the millimeter wave bands are more suitable for RSU-to-RSU communication due to the larger bandwidth and directivity. The RSU-to-vehicle communication may need to be of both directional and omni-directional for different applications. We discuss some research directions in these

layers towards improving the communication throughput and latency based on the properties of batched network coding.

First, there is a range of modulation and coding schemes (MCSs) designed for 5G. Adaptive Modulation and Coding (AMC) mechanisms are used to track the channel state and hence decide the MCS to use. Due to the requirements of TCP/IP, existing AMC algorithms in a cellular network aim for a low outage probability, e.g., 10%. To achieve low outage probability, an MCS for a lower rate and a higher signal transmission power is used. Using batched network coding, however, a low outage probability is not required, and hence the AMC can choose an MCS for a higher rate. For example, suppose that in a scenario the MCS for 100 Mbps has an outage probability 10% and the MCS for 400 Mbps has an outage probability of 50%. The existing AMC would choose 100 Mbps, but the MCS for 400 Mbps can achieve a higher communication throughput. In general, using batched network coding, more flexible AMC and the signal transmission power control can be designed towards achieving higher rates and lower energy consumption.

Second, retransmission is adopted in 5G in multiple layers. The RLC can optionally retransmit the packets and the MAC employs a hybrid-ARQ mechanism to improve the reliability. When the feedback is ideal, using retransmission is optimal for resolving packet loss. However in 5G, the feedback always has the delay of several *transmission time intervals* even for the MAC layer hybrid-ARQ. With batched network coding, the retransmission mechanisms can be optimized jointly with network coding. For example, if a packet loss rate of about 50% is expected, then for each batch with N packets received, $2N$ packets generated by RLNC can be transmitted. As network coding does not need to wait for the feedback about the packet loss, it can achieve a lower latency. It is not necessary to completely eliminate retransmission, which is still useful when feedback is available.

In general, these lower-layer functions can be incorporated in the inner code of batched network coding, and hence the design of these lower-layer functions is to maximize the capacity of the end-to-end batch channel formed by the inner code [22]. It is known that achieving an inner code with a high hop-by-hop reliability is not optimal for a multihop network [70], [71].

2) TRANSPORT AND NETWORK LAYERS

Now we discuss the mechanisms that are traditionally in the transport and network layers of the TCP/IP protocol stack. We have shown how to use a BATS code on a single path as in Fig. 8. However, the single path approach is vulnerable to link failures. If one of the three links fails in the network in Fig. 8, the communication from node s to node t fails. A single point of failure is one of the major concern of using multihop wireless networks.

Actually, a BATS code can readily be used in a more general topology [72]. For the network example shown in

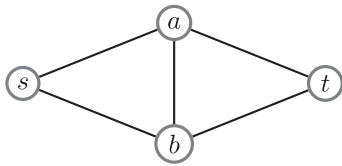


FIGURE 11. A network of four nodes denoted as s , a , b and t . All the network links illustrated by solid lines are bidirectional. There are four acyclic paths from the source node s to the destination node t : $s \rightarrow a \rightarrow t$, $s \rightarrow b \rightarrow t$, $s \rightarrow b \rightarrow a \rightarrow t$, and $s \rightarrow a \rightarrow b \rightarrow t$.

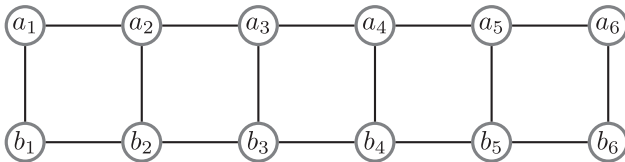


FIGURE 12. This network illustrates the case that the RSUs are deployed on both sides of a road, and form a grid topology to avoid a single point of failure. Nodes a_1 to a_6 are on the one side of road and nodes b_1 to b_6 are on the other side.

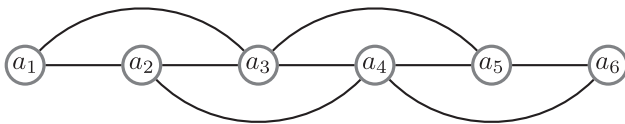


FIGURE 13. This network illustrates the case that the RSUs are deployed on one side of a road. In this topology, each node connects directly to its neighbors within a two-hop distance.

Fig. 11, four acyclic paths exist from the source node s to the destination node t . The traditional routing algorithms usually choose a single path for the communication from the source node s to the destination node t subject to a shortest path or lowest latency criterion. A BATS code can use all the available links in the network simultaneously: The batches generated at the source node s can be transmitted on both outgoing links; Node a can transmit its received batches from node s on both outgoing links after applying network coding; Node b can transmit its received batches from both node s and node a after applying network coding. Node t receives batches from both of its incoming links; and it only needs to receive a sufficient total number of batches to decode the message packets.

Benefiting from the capability of a BATS code for communication beyond a single path, we can design the topology of an RSU network to avoid a single point of failure. In Figs. 12 and 13, we illustrate two examples. In Fig. 12, the RSUs are deployed on both sides of a road, and form a grid topology that can tolerate any single link failure. In Fig. 13, though the RSUs are only deployed on one side of the road, it is also possible to form a topology more complicated than a line to avoid a single point of failure.

For communication in the RSU network as structured above, some resource allocation problems must be considered.

The wireless links in the RSU network need to share some common frequency and time resources. The cellular network has a good solution for the RSU-to-vehicle communication resource management. Using batched network coding, how to further manage the resource allocation involving the RSU-to-RSU links is a new research issue. The problem is related to the IAB network in 5G [89], but has a larger scale and more complicated network topology.

When there are multiple communication flows in a network, a flow control mechanism is required to adjust the network resource allocation. Even for single-path communication, due to network coding, the end-to-end control mechanism used in TCP is not valid. In addition to control the end-to-end information rate, the communication cost on each network link should also be controlled [86]. To support the more complicated network topology for a BATS code flow, a new flow control mechanism is required.

3) MOBILITY

We consider the mobility of vehicles for roadside sensing and roadside control, respectively. Suppose that a road segment is covered by a number of RSUs, which jointly process the sensing data. For roadside sensing IVCAD, the sensing data can be transmitted to the self-driving vehicles moving towards this road segment by broadcasting the information from all the nearby RSUs. Using a BATS code, each RSU can generate batches independently. In other words, multiple RSUs can simultaneously transmit independent information of the same data without explicit coordination. A vehicle on the road just keep receiving the packets from the RSUs when moving along the road. Benefiting from the rateless property of the BATS code, a vehicle can decode the sensing data from the packets received from multiple RSUs. Therefore, for roadside sensing, the mobility of vehicles has no effect on how the roadside infrastructure broadcasts the information.

For roadside control IVCAD, the sensing data is not directly transmitted to the vehicles. Instead, the RSU network should use its sensing capability to keep track of the vehicles to be controlled. Using the sensing data on the road, the RSU network generates the driving instruction for each vehicle, and then transmits the driving instruction to the specific vehicle. As the driving path of the vehicle is known by the RSU network, the RSU network can store the driving instruction message to be transmitted to the vehicle on specific RSUs. The rateless property of the BATS code enable a vehicle to decode the driving instruction from the packets received from multiple RSUs.

V. CONCLUSION

We provided our thinking about the developments of wireless communication technologies towards a city-wide infrastructure-vehicle cooperative autonomous driving system, where vehicles rely on the roadside infrastructure for sensing and/or decision making. We argue that an excessive

amount of investment is not required for the deployment and maintenance of new network communication facilities to support the roadside infrastructure, which has a critical communication demand in terms of latency, reliability and throughput. When the roadside units are deployed with sufficient density, which is necessary for the roadside infrastructure to sense the road environment, the neighboring RSUs can directly communicate with each other to form a wireless mesh network, called an RSU network.

The RSU network can be a scalable solution for infrastructure-vehicle cooperative autonomous driving without relying on the existing wireline and cellular network infrastructures. However, the existing network communication techniques including TCP/IP and 5G cannot effectively enable the RSU network. We discussed why network coding must be applied to support the RSU network on a large scale, and introduced a network communication approach based on an efficient network coding scheme called BATS code. Though various components of 5G can readily be used with network coding, we discussed possible improvements of network and communication techniques for better working with network coding, which may motivate further researches towards the next generation of wireless network standards, e.g., 6G.

REFERENCES

[1] Z. Bai, G. Wu, X. Qi, Y. Liu, K. Oguchi, and M. J. Barth, "Infrastructure-based object detection and tracking for cooperative driving automation: A survey," in *Proc. IEEE Intell. Veh. Symp.*, 2022, pp. 1366–1373.

[2] R. van der Horst and S. de Ridder, "Influence of roadside infrastructure on driving behavior: Driving simulator study," *Transp. Res. Rec.*, vol. 2018, no. 1, pp. 36–44, Jan. 2007.

[3] S. Liu, B. Yu, J. Tang, and Q. Zhu, "Towards fully intelligent transportation through infrastructure-vehicle cooperative autonomous driving: Challenges and opportunities," in *Proc. ACM/IEEE 58th Des. Autom. Conf.*, 2021, pp. 1323–1326.

[4] S. Liu, B. Yu, J. Tang, Y. Zhu, and X. Liu, "Communication challenges in infrastructure-vehicle cooperative autonomous driving: A field deployment perspective," *IEEE Wireless Commun.*, vol. 29, no. 4, pp. 126–131, Aug. 2022.

[5] J. Ordonez-Lucena, P. Ameigeiras, D. Lopez, J. J. Ramos-Munoz, J. Lorca, and J. Folgueira, "Network slicing for 5G with SDN/NFV: Concepts, architectures, and challenges," *IEEE Commun. Mag.*, vol. 55, no. 5, pp. 80–87, May 2017.

[6] 3rd Generation Partnership Project (3GPP), "Release 14 Description; Summary of Rel-14 Work Items," 3GPP, Sophia Antipolis, France, Tech. Rep. TR 21.914, 2017.

[7] 3rd Generation Partnership Project (3GPP), "Study on physical layer enhancements for NR ultra-reliable and low latency case (URLLC)," 3GPP, Sophia Antipolis, France, Tech. Rep. TR 38.824, 2019.

[8] 3rd Generation Partnership Project (3GPP), "Study on enhancement of ultra-reliable low-latency communication (URLLC) support in the 5G Core network (5GC)," 3GPP, Tech. Rep. TR 23.725, 2019.

[9] 3rd Generation Partnership Project (3GPP), "User Equipment (UE) radio transmission and reception for enhanced NR sidelink," 3GPP, Sophia Antipolis, France, Tech. Rep. TR 38.785, 2022.

[10] X. Xiong, J. Sha, and L. Jin, "Optimizing coordinated vehicle platooning: An analytical approach based on stochastic dynamic programming," *Transp. Res. Part B: Methodological*, vol. 150, pp. 482–502, Aug. 2021.

[11] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, "Network information flow," *IEEE Trans. Inf. Theory*, vol. 46, no. 4, pp. 1204–1216, Jul. 2000.

[12] S.-Y. R. Li, R. W. Yeung, and N. Cai, "Linear network coding," *IEEE Trans. Inf. Theory*, vol. 49, no. 2, pp. 371–381, Feb. 2003.

[13] R. Koetter and M. Medard, "An algebraic approach to network coding," *IEEE/ACM Trans. Netw.*, vol. 11, no. 5, pp. 782–795, Oct. 2003.

[14] Y. Wu, "A trellis connectivity analysis of random linear network coding with buffering," in *Proc. IEEE Int. Symp. Inf. Theory*, 2006, pp. 768–772.

[15] A. F. Dana, R. Gowaikar, R. Palanki, B. Hassibi, and M. Effros, "Capacity of wireless erasure networks," *IEEE Trans. Inf. Theory*, vol. 52, no. 3, pp. 789–804, Mar. 2006.

[16] T. Ho et al., "A random linear network coding approach to multicast," *IEEE Trans. Inf. Theory*, vol. 52, no. 10, pp. 4413–4430, Oct. 2006.

[17] R. W. Yeung, "Avalanche: A network coding analysis," *Commun. Inf. Syst.*, vol. 7, no. 4, pp. 353–358, 2007.

[18] D. S. Lun, M. Médard, R. Koetter, and M. Effros, "On coding for reliable communication over packet networks," *Phys. Commun.*, vol. 1, no. 1, pp. 3–20, Mar. 2008.

[19] S. Yang and R. W. Yeung, "Batched sparse codes," *IEEE Trans. Inf. Theory*, vol. 60, no. 9, pp. 5322–5346, Sep. 2014.

[20] B. Tang, S. Yang, Y. Yin, B. Ye, and S. Lu, "Expander chunked codes," *EURASIP J. Adv. Signal Process.*, vol. 2015, no. 1, pp. 1–13, Dec. 2015.

[21] B. Tang and S. Yang, "An LDPC approach for chunked network codes," *IEEE/ACM Trans. Netw.*, vol. 26, no. 1, pp. 605–617, Feb. 2018.

[22] S. Yang and R. W. Yeung, "Network communication protocol design from the perspective of batched network coding," *IEEE Commun. Mag.*, vol. 60, no. 1, pp. 89–93, Jan. 2022.

[23] C. Badue et al., "Self-driving cars: A survey," *Expert Syst. Appl.*, vol. 165, Mar. 2021, Art. no. 113816.

[24] S. Royo and M. Ballesta-Garcia, "An overview of LiDAR imaging systems for autonomous vehicles," *Appl. Sci.*, vol. 9, no. 19, Sep. 2019, Art. no. 4093.

[25] J. Vargas, S. Alsweiss, O. Toker, R. Razdan, and J. Santos, "An overview of autonomous vehicles sensors and their vulnerability to weather conditions," *Sensors*, vol. 21, no. 16, Aug. 2021, Art. no. 5397.

[26] M. Papageorgiou et al., "ITS and traffic management," *Handbooks Oper. Res. Manage. Sci.*, vol. 14, pp. 715–774, 2007.

[27] S. I. Guler, M. Menendez, and L. Meier, "Using connected vehicle technology to improve the efficiency of intersections," *Transp. Res. Part C: Emerg. Technol.*, vol. 46, pp. 121–131, Sep. 2014.

[28] X. Xiong, K. Ozbay, L. Jin, and C. Feng, "Dynamic origin-destination matrix prediction with line graph neural networks and Kalman filter," *Transp. Res. Rec.*, vol. 2674, no. 8, pp. 491–503, Aug. 2020.

[29] N. Oliver, K. Potočnik, and T. Calvard, "To make self-driving cars safe, we also need better roads and infrastructure," *Harvard Bus. Rev.*, Aug. 2018. [Online]. Available: <https://hbr.org/2018/08/to-make-self-driving-cars-safe-we-also-need-better-roads-and-infrastructure>

[30] J. P. V. Talusan, M. Wilbur, A. Dubey, and K. Yasumoto, "Route planning through distributed computing by road side units," *IEEE Access*, vol. 8, pp. 176134–176148, 2020.

[31] S. Liu and J.-L. Gaudiot, "Autonomous vehicles lite self-driving technologies should start small, go slow," *IEEE Spectr.*, vol. 57, no. 3, pp. 36–49, Mar. 2020.

[32] S. Masi, P. Xu, P. Bonnifait, and S. Ieng, "Augmented perception with cooperative roadside vision systems for autonomous driving in complex scenarios," in *Proc. IEEE Int. Intell. Transp. Syst. Conf.*, 2021, pp. 1140–1146.

[33] Y. He, L. Ma, Z. Jiang, Y. Tang, and G. Xing, "VI-eye: Semantic-based 3D point cloud registration for infrastructure-assisted autonomous driving," in *Proc. 27th Annu. Int. Conf. Mobile Comput. Netw.*, 2021, pp. 573–586.

[34] S. Shi et al., "VIPS: Real-time perception fusion for infrastructure-assisted autonomous driving," in *Proc. 28th Int. Conf. Mobile Comput. Netw.*, 2022, pp. 133–146.

[35] L. Le, A. Festag, R. Baldessari, and W. Zhang, "Vehicular wireless short-range communication for improving intersection safety," *IEEE Commun. Mag.*, vol. 47, no. 11, pp. 104–110, Nov. 2009.

[36] S. Gao, E. Frejinger, and M. Ben-Akiva, "Adaptive route choice models in stochastic time-dependent networks," *Transp. Res. Rec.*, vol. 2085, no. 1, pp. 136–143, Jan. 2008.

[37] A. Maxmen, "Self-driving car dilemmas reveal that moral choices are not universal," *Nature*, vol. 562, no. 7728, pp. 469–469, Oct. 2018.

- [38] S. Liu, L. Liu, J. Tang, B. Yu, Y. Wang, and W. Shi, "Edge computing for autonomous driving: Opportunities and challenges," *Proc. IEEE*, vol. 107, no. 8, pp. 1697–1716, Aug. 2019.
- [39] H. Hasrouny, A. E. Samhat, C. Bassil, and A. Laouiti, "VANet security challenges and solutions: A survey," *Veh. Commun.*, vol. 7, pp. 7–20, Jan. 2017.
- [40] A. Guerna, S. Bitam, and C. T. Calafate, "Roadside unit deployment in internet of vehicles systems: A survey," *Sensors*, vol. 22, no. 9, Apr. 2022, Art. no. 3190.
- [41] Y. Hu, S. Fang, Z. Lei, Y. Zhong, and S. Chen, "Where2comm: Communication-efficient collaborative perception via spatial confidence maps," in *Proc. 36th Conf. Adv. Neural Inf. Process. Syst.*, 2022.
- [42] N. Nikookaran, G. Karakostas, and T. D. Todd, "Combining capital and operating expenditure costs in vehicular roadside unit placement," *IEEE Trans. Veh. Technol.*, vol. 66, no. 8, pp. 7317–7331, Aug. 2017.
- [43] I. F. Akyildiz and X. Wang, "A survey on wireless mesh networks," *IEEE Commun. Mag.*, vol. 43, no. 9, pp. S23–S30, Sep. 2005.
- [44] 3rd Generation Partnership Project (3GPP), "Release 16 Description; Summary of Rel-16 Work Items," 3GPP, Sophia Antipolis, France, Tech. Rep. TR 21.916, 2020.
- [45] 3rd Generation Partnership Project (3GPP), "Release 17 Description; Summary of Rel-17 Work Items," 3GPP, Sophia Antipolis, France, Tech. Rep. TR 21.917, 2022.
- [46] C. Li, J. Jose, and X. Wu, "Distributed-fountain network code (DFNC) for content delivery in vehicular networks," in *Proc. 10th ACM Int. Workshop Veh. Inter-Netw., Syst., Appl.*, 2013, pp. 31–40.
- [47] A. Tassi et al., "Agile data offloading over novel fog computing infrastructure for CAVs," in *Proc. IEEE Veh. Technol. Conf.*, 2019, pp. 1–6.
- [48] E. Tasdemir, C. Lehmann, D. Nophut, F. Gabriel, and F. H. P. Fitzek, "Vehicle platooning: Sliding window RLNC for low latency and high resilience," in *Proc. IEEE Int. Conf. Commun. Workshops*, 2020, pp. 1–6.
- [49] S. Pandi, F. Gabriel, J. A. Cabrera, S. Wunderlich, M. Reisslein, and F. H. P. Fitzek, "PACE: Redundancy engineering in RLNC for low-latency communication," *IEEE Access*, vol. 5, pp. 20477–20493, 2017.
- [50] S. Wunderlich, F. Gabriel, S. Pandi, F. H. P. Fitzek, and M. Reisslein, "Caterpillar RLNC (CRLNC): A practical finite sliding window RLNC approach," *IEEE Access*, vol. 5, pp. 20183–20197, 2017.
- [51] D. E. Lucani et al., "Fulcrum: Flexible network coding for heterogeneous devices," *IEEE Access*, vol. 6, pp. 77890–77910, 2018.
- [52] V. Nguyen, E. Tasdemir, G. T. Nguyen, D. E. Lucani, F. H. P. Fitzek, and M. Reisslein, "DSEF Fulcrum: Dynamic sparsity and expansion packets for fulcrum network coding," *IEEE Access*, vol. 8, pp. 78293–78314, 2020.
- [53] E. Tasdemir et al., "SpaRec: Sparse systematic RLNC recoding in multi-hop networks," *IEEE Access*, vol. 9, pp. 168567–168586, 2021.
- [54] E. Tasdemir, V. Nguyen, G. T. Nguyen, F. H. P. Fitzek, and M. Reisslein, "FSW: Fulcrum sliding window coding for low-latency communication," *IEEE Access*, vol. 10, pp. 54276–54290, 2022.
- [55] P. T. Compta, F. H. P. Fitzek, and D. E. Lucani, "Network coding is the 5G key enabling technology: Effects and strategies to manage heterogeneous packet lengths," *Trans. Emerg. Telecommun. Technol.*, vol. 6, no. 1, pp. 46–55, Jan. 2015.
- [56] P. T. Compta, F. H. P. Fitzek, and D. E. Lucani, "On the effects of heterogeneous packet lengths on network coding," in *Proc. Eur. Wireless Conf.*, 2014, pp. 1–6.
- [57] M. Taghouti, D. E. Lucani, J. A. Cabrera, M. Reisslein, M. V. Pedersen, and F. H. P. Fitzek, "Reduction of padding overhead for RLNC media distribution with variable size packets," *IEEE Trans. Broadcast.*, vol. 65, no. 3, pp. 558–576, Sep. 2019.
- [58] M. Taghouti et al., "Implementation of network coding with recoding for unequal-sized and header compressed traffic," in *Proc. IEEE Wireless Commun. Netw. Conf.*, 2019, pp. 1–7.
- [59] B. Schütz and N. Aschenbruck, "Packet-preserving network coding schemes for padding overhead reduction," in *Proc. IEEE Conf. Local Comput. Netw.*, 2019, pp. 447–454.
- [60] C. de Alwis, H. Kodikara Arachchi, A. Fernando, and A. Kondo, "Towards minimising the coefficient vector overhead in random linear network coding," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2013, pp. 5127–5131.
- [61] D. Silva, "Minimum-overhead network coding in the short packet regime," in *Proc. IEEE Int. Symp. Netw. Coding*, 2012, pp. 173–178.
- [62] D. Gligoroski, K. Kralevska, and H. Øverby, "Minimal header overhead for random linear network coding," in *Proc. IEEE Int. Conf. Commun. Workshop*, 2015, pp. 680–685.
- [63] P. A. Chou, Y. Wu, and K. Jain, "Practical network coding," in *Proc. Annu. Allerton Conf. Commun., Control, Comput.*, 2003, pp. 40–49.
- [64] D. Silva, W. Zeng, and F. R. Kschischang, "Sparse network coding with overlapping classes," in *Proc. IEEE Workshop Netw. Coding Theory, Appl.*, 2009, pp. 74–79.
- [65] A. Heidarzadeh and A. H. Banihashemi, "Overlapped chunked network coding," in *Proc. IEEE Inf. Theory Workshop*, 2010, pp. 1–5.
- [66] Y. Li, E. Soljanin, and P. Spasojevic, "Effects of the generation size and overlap on throughput and complexity in randomized linear network coding," *IEEE Trans. Inf. Theory*, vol. 57, no. 2, pp. 1111–1123, Feb. 2011.
- [67] S. Yang and R. W. Yeung, "Coding for a network coded fountain," in *Proc. IEEE Int. Symp. Inf. Theory*, 2011, pp. 2647–2651.
- [68] K. Mahdavian, M. Ardakani, H. Bagheri, and C. Tellambura, "Gamma codes: A low-overhead linear-complexity network coding solution," in *Proc. IEEE Int. Symp. Netw. Coding*, 2012, pp. 125–130.
- [69] P. Maysounkov, N. J. A. Harvey, and D. S. Lun, "Methods for efficient network coding," in *Proc. Allerton Conf. Comm., Control, Comput.*, 2006, pp. 482–491.
- [70] S. Yang, J. Wang, Y. Dong, and Y. Zhang, "On the capacity scalability of line networks with buffer size constraints," in *Proc. IEEE Int. Symp. Inf. Theory*, 2019, pp. 1507–1511.
- [71] S. Yang and J. Wang, "Upper bound scalability on achievable rates of batched codes for line networks," in *Proc. IEEE Int. Symp. Inf. Theory*, 2020, pp. 1629–1634.
- [72] S. Yang and R. W. Yeung, *BATS Codes: Theory and Practice, Ser. Synthesis Lectures on Communication Networks*. San Rafael, CA, USA: Morgan & Claypool Publishers, 2017.
- [73] H. H. F. Yin, H. W. H. Wong, M. Tahernia, and J. Qing, "Packet size optimization for batched network coding," in *Proc. IEEE Int. Symp. Inf. Theory*, 2022, pp. 1584–1589.
- [74] X. Xu, Y. L. Guan, Y. Zeng, and C. C. Chui, "Quasi-universal BATS code," *IEEE Trans. Veh. Technol.*, vol. 66, no. 4, pp. 3497–3501, Apr. 2017.
- [75] X. Xu, Y. Zeng, Y. L. Guan, and L. Yuan, "BATS code with unequal error protection," in *Proc. IEEE Int. Conf. Commun. Syst.*, 2016, pp. 1–6.
- [76] X. Xu, Y. Zeng, Y. L. Guan, and L. Yuan, "Expanding-window BATS code for scalable video multicasting over erasure networks," *IEEE Trans. Multimedia*, vol. 20, no. 2, pp. 271–281, Feb. 2018.
- [77] J. Yang, Z.-P. Shi, C.-X. Wang, and J.-B. Ji, "Design of optimized sliding-window BATS codes," *IEEE Commun. Lett.*, vol. 23, no. 3, pp. 410–413, Mar. 2019.
- [78] J. Qing, H. H. F. Yin, and R. W. Yeung, "Enhancing the decoding rates of BATS codes by learning with guided information," in *Proc. IEEE Int. Symp. Inf. Theory*, 2022, pp. 37–42.
- [79] Q. Zhou, S. Yang, H. H. F. Yin, and B. Tang, "On BATS codes with variable batch sizes," *IEEE Commun. Lett.*, vol. 21, no. 9, pp. 1917–1920, Sep. 2017.
- [80] M. Luby, "LT codes," in *Proc. IEEE 43rd Annu. Symp. Found. Comput. Sci.*, 2002, pp. 271–280.
- [81] B. Tang, S. Yang, B. Ye, S. Guo, and S. Lu, "Near-optimal one-sided scheduling for coded segmented network coding," *IEEE Trans. Comput.*, vol. 65, no. 3, pp. 929–939, Mar. 2016.
- [82] X. Xu, Y. L. Guan, and Y. Zeng, "Batched network coding with adaptive recoding for multi-hop erasure channels with memory," *IEEE Trans. Commun.*, vol. 66, no. 3, pp. 1042–1052, Mar. 2018.
- [83] Z. Zhou, C. Li, S. Yang, and X. Guang, "Practical inner codes for BATS codes in multi-hop wireless networks," *IEEE Trans. Veh. Technol.*, vol. 68, no. 3, pp. 2751–2762, Mar. 2019.
- [84] H. H. F. Yin, B. Tang, K. H. Ng, S. Yang, X. Wang, and Q. Zhou, "A unified adaptive recoding framework for batched network coding," *IEEE J. Sel. Topics Inf. Theory*, vol. 2, no. 4, pp. 1150–1164, Dec. 2021.
- [85] S. Yang, S.-W. Ho, J. Meng, and E.-H. Yang, "Capacity analysis of linear operator channels over finite fields," *IEEE Trans. Inf. Theory*, vol. 60, no. 8, pp. 4880–4901, Aug. 2014.

[86] Y. Dong, S. Jin, Y. Chen, S. Yang, and H. H. F. Yin, "Utility maximization for multihop wireless networks employing BATS codes," *IEEE J. Sel. Topics Inf. Theory*, vol. 2, no. 4, pp. 1120–1134, Dec. 2021.

[87] H. H. F. Yin and M. Tahernia, "On the design of timeout-based recoders for batched network codes in the MU-MIMO regime," *IEEE J. Sel. Areas Inf. Theory*, vol. 2, no. 4, pp. 1150–1164, Dec. 2022.

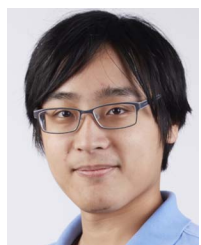
[88] H. H. F. Yin, R. W. Yeung, and S. Yang, "A protocol design paradigm for batched sparse codes," *Entropy*, vol. 22, no. 7, Jul. 2020, Art. no. 790.

[89] M. Pagin, T. Zugno, M. Polese, and M. Zorzi, "Resource management for 5G NR integrated access and backhaul: A semi-centralized approach," *IEEE Trans. Wireless Commun.*, vol. 21, no. 2, pp. 753–767, Feb. 2022.



SHENGCHAO YANG (Member, IEEE) received the B.S. degree from Nankai University, Tianjin, China, in 2001, the M.S. degree from Peking University, Beijing, China, in 2004, and the Ph.D. degree from The Chinese University of Hong Kong (CUHK), Hong Kong, in 2008. He was a Postdoctoral Fellow with the University of Waterloo, Waterloo, On, Canada, from 2008 to 2009 and with CUHK from 2010 to 2012. From 2012 to 2015, he was an Assistant Professor with Tsinghua University, Beijing. Since 2015, he has been with

the Chinese University of Hong Kong, Shenzhen, Shenzhen, China. His research focuses on theoretical and system research related to information. He is a Co-inventor of BATS codes.



HOOVER H. F. YIN (Member, IEEE) received the B.Sc., B.Eng., and Ph.D. degrees from The Chinese University of Hong Kong (CUHK), Hong Kong, in 2013, 2014 and 2019, respectively. From 2019 to 2022, he was a Postdoctoral Fellow with the Institute of Network Coding, CUHK. He is currently a Lecturer with the Department of Information Engineering, CUHK. His research focuses on network coding, computer and network systems, source coding, and recreational mathematics.



RAYMOND W. YEUNG (Fellow, IEEE) received the B.S., M.Eng., and Ph.D. degrees from the Cornell University, Ithaca, NY, USA. He is currently with the Department of Information Engineering, Institute of Network Coding, The Chinese University of Hong Kong (CUHK), Hong Kong. In 1988, he joined the AT&T Bell Laboratories, Murray Hill, NJ, USA. Since 1991, he has been with CUHK, where he is currently a Choh-Ming Li Professor of Information Engineering. He is a Co-founder of the field of network coding and Co-

inventor of BATS codes. He was the recipient of many awards, including the 2021 IEEE Richard W. Hamming Medal and 2022 Claude E. Shannon Award, for his research contributions.



XI XIONG received the B.Sc. degree from Jilin University, Changchun, China, and the M.Sc. degree from Tsinghua University, Beijing, China, and the Ph.D. degree in transportation from the New York University. He is currently a Research Scientist with the Shenzhen Research Institute of Big Data (SRIBD), Shenzhen, China. and associated with The Chinese University of Hong Kong, Shenzhen. Before joining the SRIBD, he was a Postdoctoral Visiting Fellow with the Harvard Kennedy School, Cambridge, MA, USA. He was

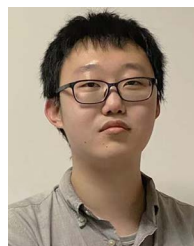
also a Research Assistant with the NYU C2SMART Department of Transportation Center and the Center for Urban Science And Progress. His research interest include developing analytical and computational approaches to improve transportation efficiency and resiliency in increasingly connected cities.



YANLONG HUANG received the B.Eng. degree from the South China University of Technology, Guangzhou, China, in 2017, and the M.Phil. degree from the Chinese University of Hong Kong, Shenzhen, Shenzhen, China, in 2022. He is currently a Research Assistant with the Chinese University of Hong Kong, Shenzhen. His research focuses on mobile computing and networking.



LINGFEI MA received the B.S. degree in electronic of science and technology from the Beijing Institute of Technology, Beijing, China, in 2017. She is currently working toward the M.Phil. degree in computer and information engineering with the Chinese University of Hong Kong, Shenzhen, Shenzhen, China. Her research interests include decentralized object tracking and storage in blockchains.



MO LI received the B.Eng. degree from the University of New South Wales, Sydney, NSW, Australia, in 2020. He is currently working toward the Ph.D. degree with The Chinese University of Hong Kong, Hong Kong. His research interests include network coding and joint channel coding-modulation.



CHAORYANG TANG received the B.Eng. degree from Jilin University, Changchun, China, in 2020. He is currently a Research Assistant with the Chinese University of Hong Kong, Shenzhen, Shenzhen, China. His research interests include network coding and 6G communication.