Computer Networks 105 (2016) 111–123

Contents lists available at ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet

Virtual overhearing: An effective way to increase network coding opportunities in wireless ad-hoc networks

Ling Fu Xie[a], Peter Han Joo Chong[b,*], Ivan Wang-Hei Ho[c], Henry C.B. Chan[d]

[a] Faculty of Electrical Engineering and Computer Science, Ningbo University, China
[b] Department of Electrical and Electronic Engineering, Auckland University of Technology, Auckland, New Zealand
[c] Department of EIE, The Hong Kong Polytechnic University, Hong Kong
[d] Department of Computing, The Hong Kong Polytechnic University, Hong Kong

ARTICLE INFO

Article history:
Received 19 February 2015
Revised 25 April 2016
Accepted 30 May 2016
Available online 1 June 2016

Keywords:
Network coding
Overhearing
Virtual overhearing
Coding-aware routing

ABSTRACT

Overhearing is of great importance to wireless network coding in that it can be exploited to obtain the side information needed for packet decoding. Recently, a new technique called virtual overhearing (VOH) was proposed to allow a node to obtain the packet sent by another node that is multiple hops away for free. This can overcome the limitation of overhearing and be used to discover more coding opportunities. In this paper, we take advantage of VOH and propose two modes of exploiting VOH to increase coding opportunities in wireless ad-hoc networks. First, we make use of VOH to increase the chance of finding a route with coding opportunities for a new incoming flow. Second, and more importantly, we make use of VOH to *create* coding opportunities between two established flows which are currently unmixable. Note that most previous studies only attempt to find coding opportunities rather than create them. Based on these two modes of VOH usage, we design two routing protocols: distributed coding-aware routing with virtual overhearing (DCAR-VOH), and its enhanced version DCAR-VOH+. DCAR-VOH implements only the first mode of usage, whereas DCAR-VOH+ incorporates both modes of usage. Our extensive simulations indicate that VOH provides an effective way to discover coding opportunities, resulting in improved network performance. The positive effect of the second mode of usage stands out especially.

1. Introduction

Overhearing is of great importance to network coding in wireless networks. In the past decade, network coding [1] emerges as a promising tool to effectively boost wireless network capacity via packet encoding/mixing [2-4]. One important form of network coding is the inter-flow network coding [6-8], which encodes packets from different flows using XOR [6] or random linear combination [5] and serves those flows simultaneously with the coded packet(s). To decode the coded packet, some node(s) on one flow must be able to obtain the packet(s) or the side information [9] from the other flows; overhearing is generally exploited for the side information acquisition, as illustrated in Fig. 1(a). Many previous works [13-18] on inter-flow network coding exploit overhearing for network coding opportunity discovery. For example, DCAR [13] defines two conditions for an intermediate node to become an encoding node for two flows: (1) this node must be an intersection node, i.e., a common node on the paths of the two flows; and (2) with respect to this node, a downstream node of each of the flows must be able to *overhear* an upstream node of the other flow.

Though overhearing provides an appealing way for the side information acquisition in network coding, it only takes effect in the *one-hop* neighborhood of a node and thus is restrictive in coding opportunity discovery. Fig. 1(b) presents a general scenario in multi-hop wireless ad-hoc networks to illustrate how overhearing fails to bring about coding opportunities. It can be observed from the figure that all the previous works [13-18] could not find any coding opportunity at node M, the intersection node of flow 1 (f_1) and flow 3 (f_3). This is because none of the downstream nodes of f_3, node E or G, can overhear packets of f_1.

Recently, a new technique called virtual overhearing (VOH) [9] was proposed to overcome the limitation of overhearing for discovering more coding opportunities. VOH enables a node to obtain a packet sent by another node that can be *multiple* hops away for free. Thus, with VOH, a node can *virtually* overhear another node that is far away. In fact, given the establishment of f_1 and f_2 in Fig. 1(b), VOH can be applied to allow all upstream nodes, i.e., nodes *G* and *F*, of node *H* on f_2 to obtain the packet (e.g., P_1) sent by node *A* on f_1 at no cost, details of which will be illustrated

* Corresponding author.
E-mail addresses: xielingfu@nbu.edu.cn (L.F. Xie), peter.chong@aut.ac.nz (P.H.J. Chong), ivanwh.ho@polyu.edu.hk (I.W.-H. Ho), cshchan@comp.polyu.edu.hk (H.C.B. Chan).

http://dx.doi.org/10.1016/j.comnet.2016.05.023
1389-1286/

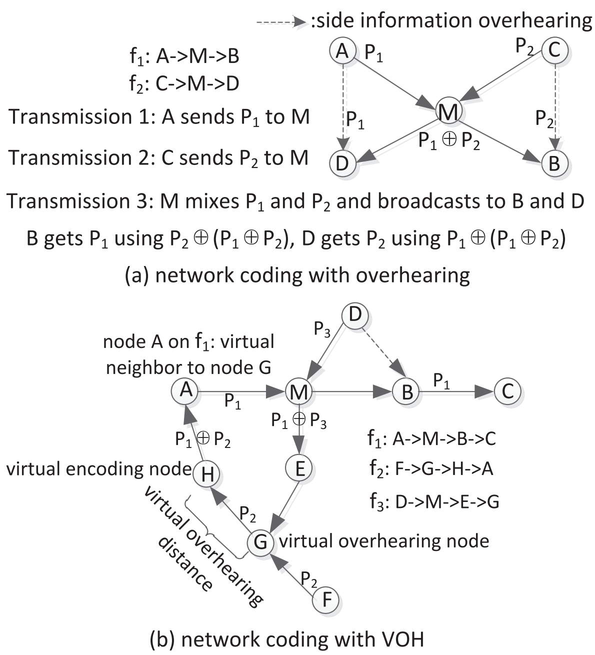


Fig. 1. Inter-flow network coding examples.

in the next section where VOH is reviewed. Thus, with VOH in Fig. 1(b), the intersection node, node *M*, of f_1 and f_3 is now enabled to encode packets from the two flows, because node *B* on f_1 and node *G* on f_3 can obtain the packets P_3 and P_1 respectively and both can decode the coded packet, i.e., $P_1 \oplus P_3$, by node *M*.

The authors in [9] mainly investigate how the extra coding opportunities brought about by VOH improve network performance in a simple practical network, but they fail to consider how to make better use of VOH in a general network, e.g., in multi-hop wireless ad-hoc networks. In this paper, we take advantage of VOH and propose the following two modes of VOH usage to increase coding opportunities. Refer to Fig. 1(b) again.

Increasing the chance of finding a route with coding opportunities for a new incoming flow: Consider the situation when f_3 in Fig. 1(b) is the last flow to enter the network. Here, f_3 could be made aware of the existing VOH between node *G* on f_2 and node *A* on f_1, and take it into consideration in the route discovery process. By doing so, f_3 could find a route with coding opportunities, thanks to VOH. Actually, this routing strategy falls into network coding-aware routing [12], which aims to find coding opportunities for a new incoming flow with existing flows before its establishment. It is shown in many previous works [13-18] that coding-aware routing yields more coding opportunities and thus benefits network performance.

Allowing any new emerging of VOH (caused by new flow establishment) to create coding opportunities between two established flows which are currently unmixable: Consider the situation when f_1 is the last flow to enter the network. The establishment of f_1 will then enable VOH between node *G* on f_2 and node *A* on f_1, which *creates* coding opportunities for f_1 and f_3 at node *M*. It is similar if f_2 is the last flow to be established.

Our work in this paper is based on the two modes of VOH usage, and compared with [9], it has the following major contributions.

- First, we propose distributed coding-aware routing with virtual overhearing (DCAR-VOH) to implement the first mode of VOH usage. DCAR-VOH allows an incoming flow to consider existing VOH for coding opportunity discovery during its route setup.
- Second, we further propose an enhanced version of DCAR-VOH, namely DCAR-VOH+, to additionally implement the second mode of VOH usage. DCAR-VOH+ allows a third flow to create coding opportunities for two currently unmixable flows. This makes it superior to many prior network coding schemes [13-18] in general, because if two existing flows are not mixable in those prior schemes, they will never be mixable regardless of the setup of other flows.
- Third, we propose and implement an adaptive encoding mechanism in both DCAR-VOH and DCAR-VOH+. With this mechanism, the packet mixing at the encoding node is controlled such that the decoding of the coded packets could be guaranteed.
- Fourth, we conduct extensive computer simulations to study the performance of the two routing schemes in various network conditions, and show that DCAR-VOH+ stands out in most situations.

The rest of this paper is organized as follows. Section 2 reviews VOH and describes the mechanisms for its discovery. The designs of DCAR-VOH and DCAR-VOH+ are introduced in Sections 3 and 4, respectively. Section 5 gives the summary of the system implementation and analyzes the protocol complexity. Section 6 presents our simulation results, and Section 7 concludes this paper and discusses possible directions for future work.

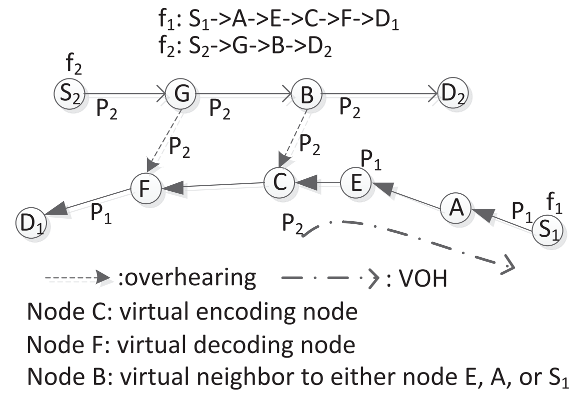


Fig. 2. Conditions of VOH.

2. VOH and its discovery

In this section, we first briefly review the conditions of VOH and illustrate how it works, and then we introduce five key definitions related to VOH. Finally, we describe the mechanisms for VOH discovery.

2.1. Conditions of VOH

For one node, say node ***A***, to virtually overhear another node, say node ***B***, the conditions are generalized in [9], which are restated as follows.

(1) Nodes ***A*** and ***B*** must be on the paths of two established flows, say f_1 and f_2, respectively.
(2) There must exist one downstream node, say node ***C***, of node ***A*** on f_1 that is able to directly overhear the packet sent by node ***B*** on f_2.
(3) There must further exist one downstream node, say node ***F***, of node ***C*** on f_1 that is able to directly overhear the packet of f_2.

Once the three conditions are met, node ***A*** on f_1 can virtually overhear node ***B*** on f_2 and obtain the packet of f_2 sent by node ***B*** for free, as will be illustrated later. Fig. 2 presents a typical scenario where node ***A*** on f_1 virtually overhears node ***B*** on f_2. Let us examine how the three conditions are met in Fig. 1(b) for node G on f_2 to virtually overhear node A on f_1. First, given the establishment of f_1 and f_2, condition (1) is met. Second, node H, i.e., a downstream node of node G on f_2, can directly overhear node A on f_1, hence condition (2) is also met. Third, node A, i.e., a downstream node of node H on f_2, can directly overhear itself on f_1 (here, we say one node can overhear itself in the sense that it knows any packet that it has sent), hence condition (3) is met. Therefore, node G on f_2 can virtually overhear node A on f_1.

We now illustrate in Fig. 1(b) how the packet, say P_1, of f_1 sent by node A can be obtained by node G for free. The idea is to allow node H on f_2 to assist as follows. Instead of sending a packet P_2 from node F to node A all the way on f_2, node H will encode P_2 with the overheard packet, P_1, from f_1 to form $P_1 \oplus P_2$. Then node H broadcasts $P_1 \oplus P_2$ to nodes A and G. Upon receiving $P_1 \oplus P_2$, node A could still decode P_2 from $P_1 \oplus P_2$ and its own native packet P_1; and upon overhearing $P_1 \oplus P_2$, node G could decode P_1 from $P_1 \oplus P_2$ and the previously stored P_2. This way, node G obtains or virtually overhears the packets sent by node A on f_1. In fact, similar to node H, node G can also encode P_1 with any packet of f_2 and broadcast the coded packet to let node F obtain P_1. In particular, it is desirable for node G to obtain f_1's packets, as node G, in the context of network coding at node M, needs f_1's packets to decode the coded packets of f_3.

To make use of VOH in the routing protocol design, we first introduce the following five definitions related to VOH. Refer to Fig. 2 again. Node ***C*** is termed a *virtual encoding* node on f_1, and node ***F*** is termed a *virtual decoding* node to node ***C*** on f_1; node ***A*** is termed a *virtual overhearing* node on f_1; node ***B*** on f_2 is termed a *virtual neighbor* to node ***A*** on f_1; and the distance (or the number of hops) between nodes ***A*** and ***C*** on f_1 is called *the virtual overhearing distance* between node ***A*** and its virtual neighbor node ***B***. Fig. 1(b) also illustrates these new terms. Obviously, VOH starts at the virtual encoding node, and the virtual decoding node is responsible for decoding the packet coded by the virtual encoding node.

2.2. Virtual neighbor discovery

In this subsection, we propose a *distributed* mechanism for a node on a flow to discover all potential virtual neighbors from other flows. In order to do that, two requirements are imposed.

First, every node on an existing flow records both the entire path consisting of the node IDs of all the nodes on that flow and the flow ID of that flow. Note that the flow ID is uniquely represented by the structure of (*source ID, flow sequence number*), where the flow sequence number is generally assigned in an increasing order to the subsequent flows generated from a source. The entire path can be directly obtained because source routing is normally adopted by network coding-aware routing [13-18]. Second, once a new flow is established, every node except the destination node on that flow broadcasts the new flow ID to its neighbors. We employ Hello message broadcast to carry this new flow ID information. Upon receiving this broadcast, each neighbor not on the same flow creates a local record in the form of (*neighbor_flow_id, hello_source*) where *neighbor_flow_id* is the new flow ID in the Hello message and *hello_source* is the ID of the node sending this Hello message. We refer to this record as the overhearing record (OHR). The virtual neighbor discovery involves two steps:

Step 1: Reporting OHR to the upstream nodes – This step is triggered by one of the following two situations:

(1) A node on some existing flow(s) creating a new OHR – If a node on some existing flow(s) creates a new OHR, then, for each existing flow, it generates a packet containing the new OHR and that existing flow ID, *my_flow_id*, and forwards that packet to all of its upstream nodes on that existing flow. Upon receiving this packet, each upstream node extracts and stores the information locally in the form of (*neighbor_flow_id, my_flow_id, hello_source, downstream_node*) where the *downstream_node* is the node sending this packet. We refer to this information as the reported overhearing record (ROHR). Note that the step in this situation will allow a node on some existing flow(s) to find some virtual neighbors from a newly established flow.

(2) A node with some previously stored OHRs appearing on a newly established flow – If a node on a newly established flow has previously stored OHRs, it creates a packet to contain all of those OHRs together with the new flow ID, and forwards this packet to its upstream nodes along the new flow. Upon receiving that packet, each upstream node creates an ROHR for each OHR contained in the packet. The step in this situation will allow a node on a newly established flow to discover some virtual neighbors from existing flows.

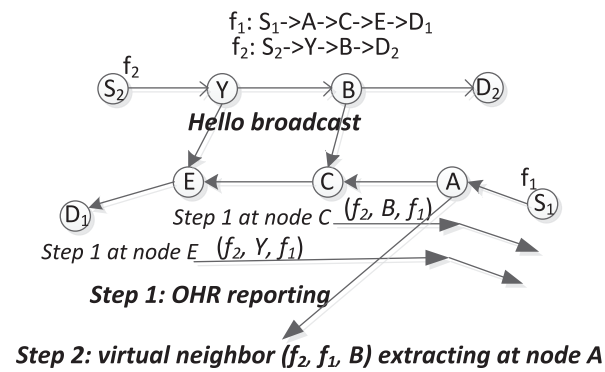


Fig. 3. Virtual neighbor discovery.

Step 2: Extracting virtual neighbors from ROHRs – After Step 1, a node may have stored some ROHRs. Then, it tries to find some virtual neighbors from those ROHRs. Among those ROHRs, if there are two records with the same first two elements, i.e., *neighbor_flow_id* and *my_flow_id*, a new virtual neighbor is found according to the VOH conditions and the definition in Section 2.1. A virtual neighbor record (VNR) in the form of (*neighbor_flow_id, my_flow_id, VN_id*) is then created for the newly found virtual neighbor, where the *VN_id* is the virtual neighbor's ID, i.e., the *hello_source* stored in one of the two ROHRs.

Fig. 3 shows an example of the virtual neighbor discovery process in the first situation. Suppose f_2 is a newly established flow. Then, after nodes C and E on f_1 receive Hello messages from nodes B and Y on f_2, respectively, each of them creates a new OHR locally and forwards that OHR to the upstream nodes on f_1. After node A on f_1 receives the two OHRs from its downstream nodes, it creates two ROHRs, (f_2, f_1, B, C) and (f_2, f_1, Y, E). With these two ROHRs, node A knows that node B on f_2 is a new virtual neighbor and records it as a new VNR in the form of (f_2, f_1, B) locally. In Section 5, we will show the two steps for virtual neighbor discovery can be done without incurring extra overhead in terms of the number of control packets.

3. DCAR-VOH

In this section, we describe how DCAR-VOH makes use of VOH during the establishment of a new incoming flow to find a route with coding opportunities. The route discovery, the route selection, and the VOH initiation are the three main steps in DCAR-VOH.

3.1. Route discovery

When a source intends to set up a flow to a destination, it needs to collect all possible paths to the destination and select the best one among them. We employ source routing for this purpose.

First, to collect those paths, the source broadcasts a route request (RREQ) packet. The RREQ packet consists of the source ID, the destination ID, and a flow sequence number.

Next, upon receiving an RREQ, a node updates its local record of the sequence number for this source and further broadcasts this RREQ, if it is not the destination and the sequence number in that RREQ is larger than the one it previously received from this source. Before broadcasting, the node appends its ID in the RREQ packet so as to eventually form the entire path. It also appends all of its one-hop neighbors and VNRs in this packet. The one-hop neighbors are directly obtained from the Hello messages and the VNRs are obtained from the mechanism in Section 2.2.

Once this RREQ reaches its destination, the destination then sends a route reply (RREP) packet back to the source along the opposite path recorded in the RREQ packet.

During the forwarding of the RREP back to the source, each intermediate node needs to execute the following two tasks.

(1) Coding opportunity identification: If this intermediate node is on some existing flows, it needs to identify whether it could be an encoding node between the new incoming flow and any of those existing flows. Here we treat the virtual neighbor in the same way as the one-hop neighbor. Thus, we adopt the same two conditions as in DCAR mentioned in Section 1 to identify the coding opportunity between two flows. However, a higher priority is given to *direct* overhearing than to VOH. This means that if the network coding between two flows can be simultaneously supported by direct overhearing and VOH, direct overhearing will be used. This is because network coding with VOH is less reliable, as will be seen from the simulation studies.

(2) Channel congestion state (CCS) reporting: Each intermediate node is required to report its channel congestion state [13] in the RREP so that the source can evaluate the returned routes and choose the best one. We adopt the metric, the modified interface queue length (MIQ) defined in DCAR, to reflect the channel congestion state. The MIQ length at one node is the required number of transmissions for that node to send all packets in the interface queue, and this is calculated based on the coding graph [13]. This coding graph is built based on the coding opportunities identified in task (1) for two flows. The channel congestion state of one node is simply the sum of the MIQ length at this node and all of its neighbors. In order to know the up-to-date MIQ lengths of all neighbors, every node is required to insert its current MIQ length in the Hello message.

3.2. Route selection

Once all RREP packets are returned, the source first evaluates each route by summing up the recorded MIQ lengths of all intermediate nodes in the reply. Then, it chooses the route with the smallest sum for the new incoming flow. During the packet transmission process, this selected path will be appended in the packet header.

3.3. VOH initiation

Once an intermediate node knows that it is an encoding node for a new flow, and that the network coding is supported by VOH, it initiates the VOH process. The initiation is executed once the encoding node receives the first packet of the new flow. For example, in Fig. 1(b), once node M receives a packet from f_3 for the first time, it will inform node H on f_2 to start mixing the overheard packet from f_1 and the packet from f_2 so that the decoding node of f_3, node G, can obtain the packet from f_1 later.

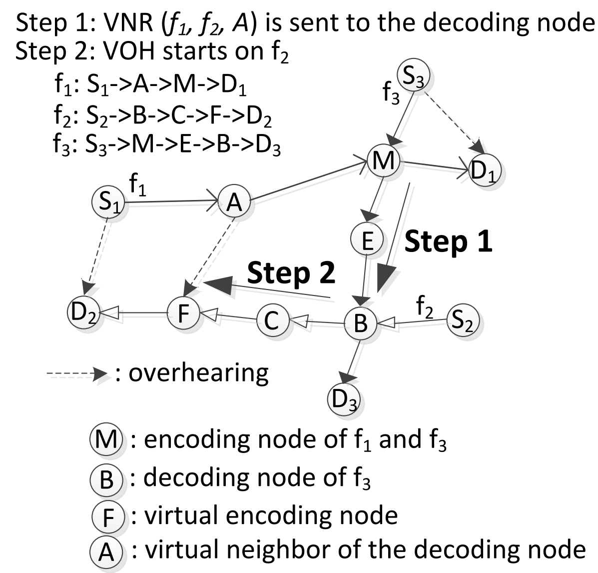


Fig. 4. VOH initiation.

Two steps are involved in the VOH initiation as shown in Fig. 4.

Step 1: *The encoding node informs the decoding node about its virtual neighbor record*. In the prior coding opportunity identification, the encoding node came to know the decoding node of the new flow and its virtual neighbor. Now, the encoding node starts sending a packet carrying this VNR to this decoding node. In Fig. 4, for example, the encoding node M of f_1 and f_3 knows from the VNR (f_1, f_2, A) in the RREP that node B is the decoding node of f_3 and node A is its virtual neighbor. Then, node M sends this VNR to node B once it receives the first packet from f_3.

Step 2: *The decoding node informs a virtual encoding node to activate the VOH process*. Upon receiving the VNR, (*neighbor_flow_id, my_flow_id, VN_id*), in Step 1, the decoding node then informs the corresponding virtual encoding node on flow, *my_flow_id*, to start mixing the overheard packet from flow, *neighboring_flow_id*. The process works as follows. The decoding node sends a packet containing the information of *neighbor_flow_id* and *VN_id* to its first downstream node on flow, *my_flow_id*. This packet will be continuously forwarded to the virtual encoding node, which can directly overhear the node, *VN_id*, on the flow *neighbor_flow_id*. Then, this virtual encoding node will start mixing the overheard packets from flow *neighbor_flow_id*. In Fig. 4, for example, the decoding node B will send a packet to the virtual encoding node F on f_2 to let it mix the packets overheard from node A. All nodes on flow *my_flow_id* between the virtual encoding node and the decoding node of the new flow will further mix the packets from flow *neighbor_flow_id* so that those packets can finally reach the decoding node.

Note that the virtual encoding node only chooses the packets that are overheard after the VOH 'start' command in Step 2 for mixing. In addition, we incorporate the following two simple mechanisms into DCAR-VOH to improve its design.

(1) '*The selective VOH*' mechanism: This is used to assist the virtual encoding node in selecting packets for mixing. If the decoding node mentioned in Step 2 lacks a certain packet, say P_1, for decoding a coded packet, it will piggyback P_1's ID onto the packets of flow *my_flow_id* to request the vir-

tual encoding node to choose P_1 for mixing. In Fig. 1(b), for example, if node G receives $P_1 \oplus P_3$ before it virtually overhears P_1, it then appends the ID of P_1 to f_2's packets (e.g., P_2) to let the virtual encoding node H first choose P_1 for mixing. In case of multiple requests, the virtual encoding node first serves the latest one.

(2) '*The adaptive encoding*' mechanism: This is used to control the packet encoding at the encoding node to ensure the decodability of the coded packets. In Fig. 1(b), if f_2 has low flow rate or throughput, it is likely that the rate of node G receiving coded packets of f_3 is much faster than the rate of node G virtually overhearing f_1's packets. This will result in many coded packets being un-decodable, and the decoding failure is actually detrimental to network performance, as will be seen from the simulation study in Section 6. To solve this problem, an adaptive encoding strategy is implemented in DCAR-VOH. Specifically, when a decoding node, e.g., node G in Fig. 1(b), detects that there are N_{UN} packets accumulated and un-decoded, it sends a packet to the encoding node, e.g., node M in Fig. 1(b), to deactivate the packet mixing. Later, once most coded packets are successfully decoded, the encoding node will be informed to resume the packet mixing. This way, the detrimental effect of low VOH rate could be greatly mitigated or even eliminated.

3.4. Discussion on flow deactivation

We now discuss the situations when flow deactivation occurs in DCAR-VOH. First, we consider the termination of the flow, e.g., f_2 in Fig. 1(b), participating in the VOH process. In this case, network coding supported by VOH should be deactivated as quickly as possible. To achieve this, choosing a relatively small N_{UN} is a feasible choice. Second, we consider the termination of the flow, e.g., f_1 or f_3 in Fig. 1(b), participating in network coding. In this case, no more coded packets will be generated, and hence, after requesting all the needed packets for decoding, a decoding node in DCAR-VOH could inform a virtual encoding node (e.g., node H in Fig. 1(b)) to shut down the VOH process.

4. DCAR-VOH+

In this section, we introduce DCAR-VOH+, an enhanced version of DCAR-VOH. It is designed on top of DCAR-VOH, and it allows any new emerging of VOH to create coding opportunities for two *unmixable* established flows.

We first present an example in Fig. 5 to show the advantage of the second mode of VOH usage. As will be seen from the simulation study, this example is actually typical under some practical network settings. In Fig. 5 three flows form a symmetric structure, and among them, f_3 is the last flow entering the network. Obviously, with the first mode of VOH usage (or in DCAR-VOH), only node C is identified as an encoding node for f_1 and f_3. If we further apply the second mode of VOH usage, nodes A and F will also be identified as encoding nodes. Specifically, the setup of f_3 enables nodes A and C to virtually overhear node C on f_3 and node F on f_2, respectively, which then create coding opportunities between f_2 and f_3 at node F and between f_1 and f_2 at node A, respectively. Hence, to benefit from the second mode of VOH usage, DCAR-VOH+ is proposed.

DCAR-VOH+ is based on DCAR-VOH, but incorporates the following two mechanisms.

(1) *Flow overhearing status recording at the non-encoding intersection node*: For each non-encoding intersection node of two flows, say f_1 and f_2, if all downstream nodes of f_1 (f_2) can neither directly nor virtually overhear the upstream node(s) of f_2 (f_1), then this intersection node is required to record this situation. This kind of information is referred to as the flow overhearing status, and it can be obtained during the route discovery for a new flow in DCAR-VOH when the coding opportunity identification is executed by the intersection node. In Fig. 1(b), for example, in the absence of f_2, the intersection node, node M, of f_1 and f_3 will make a record that none of the downstream nodes of f_3 could directly or virtually overhear the upstream nodes of f_1.

(2) *New VNR reporting*: As mentioned earlier, with the virtual neighbor discovery mechanism, a node may create some new VNRs after a new incoming flow is established. A new VNR creation means that the flow overhearing status might be changed. In DCAR-VOH+, once a node on some existing flow(s) creates a new VNR, it sends a packet to report that VNR to the upstream nodes on each existing flow. This packet carries the existing flow ID, the node ID, and the new VNR, (*neighbor_flow_id, my_flow_id, VN_id*). Consider the case in Fig. 1(b) that f_2 or f_3 is the last established flow. After the establishment of f_2 or f_3, node A on f_1 is detected by node G on f_2 as a new virtual neighbor. Then, node G reports this new VNR (f_1, f_2, A) to all of its upstream nodes on f_3, including node M.

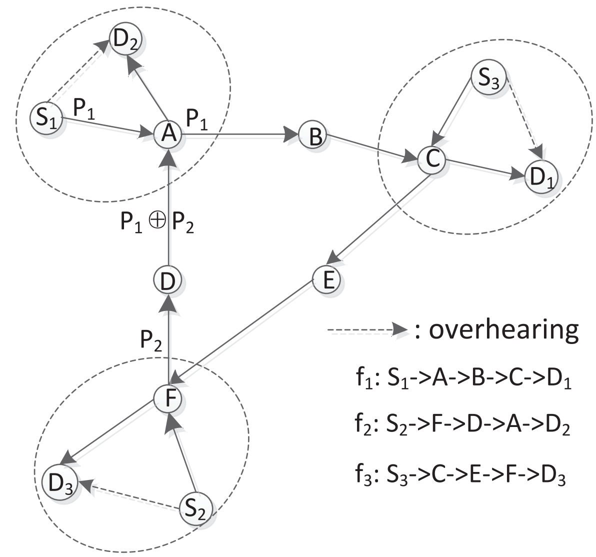


Fig. 5. Coding opportunity creation in DCAR-VOH+.

Upon receiving the reported VNR, a node knows that some downstream node of the existing flow indicated in the packet can virtually overhear some upstream node of flow *neighbor_flow_id* contained in the VNR. Hence, with the acquisition of this information, a non-encoding intersection node of two flows may update its recorded flow overhearing status between the two flows, and then determine whether new coding opportunities arise between the two flows. Let us continue with the example described in (2). Once node M receives the newly reported VNR (f_1, f_2, A) from node G, it knows that some downstream node of f_3 now can virtually overhear node A on f_1, i.e., one of its upstream nodes on f_1, and that the flow overhearing status between f_1 and f_3 can be updated. Thus, node M finally knows that it now becomes an encoding node of f_1 and f_3. Similarly, with the two proposed mechanisms (1) and (2), nodes A and F in Fig. 5 will be additionally found as the encoding nodes in DCAR-VOH+.

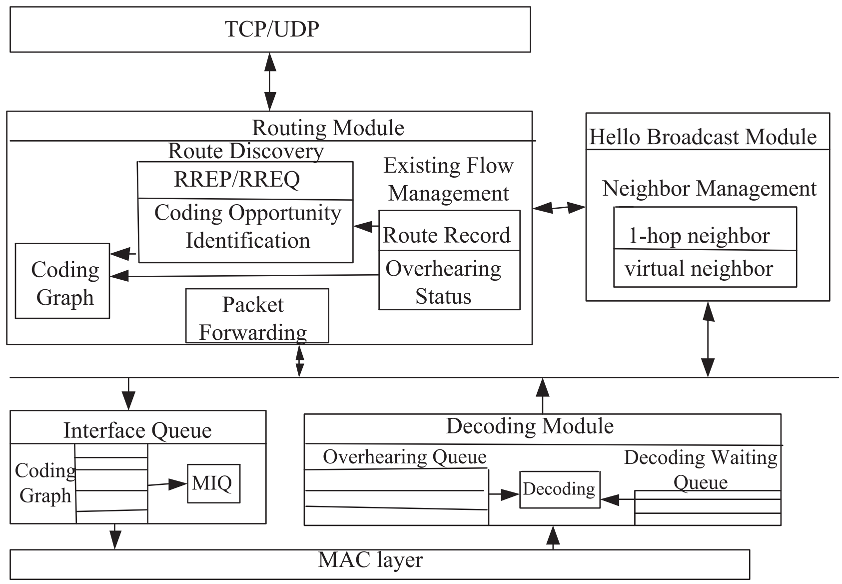


Fig. 6. The architecture of DCAR-VOH+.

5. System implementation and complexity of DCAR-VOH+

5.1. Architecture of DCAR-VOH+

We here present the system implementation of DCAR-VOH+ in NS-2. Fig. 6 shows how DCAR-VOH+ is implemented at each node. DCAR-VOH+ is composed of four major modules: the routing module, the hello broadcast module, the decoding module, and the interface queue module.

First, it can be seen from Fig. 6 that the routing module is used for each node to manage flows, responsible for the packet forwarding, the flow route discovery, the coding opportunity identification, and the overhearing status recording between any two established flows. Second, the hello broadcast module is to manage the neighbor information, including the one-hop neighbors and the virtual neighbors. Third, two queues reside in the decoding module: the overhearing queue and the decoding waiting queue. The former is used to store both directly overheard packets and virtually overheard packets. The latter is uniquely used in DCAR-VOH+ to store the coded packets, of which the decoding is supported by VOH. This is because it is possible in DCAR-VOH+ that the coded packet arrives at its decoding node earlier than the packet to be virtually overheard for decoding. For example, in Fig. 1(b), if node H is congested, packet P_1, which needs to be virtually overheard by node G, may arrive at node G later than the coded packet $P_1 \oplus P_2$. Hence, with this decoding waiting queue, a coded packet in DCAR-VOH+ could be decoded later. Here, the waiting time for this packet to be decoded is referred to as the decoding delay for this packet. Lastly, the interface queue module is used to store the packets waiting for a transmission. Each time the interface queue module is about to send a data packet to the MAC layer, it refers to the coding graph maintained by the routing module to see whether any other queued packet can be mixed with that data packet.

In DCAR-VOH+, the overhearing queue and the decoding waiting queue are managed as follows. First, for the decoding waiting queue, a simple strategy is applied, that is, in case of queue overflow, the coded packet entering the queue earliest will be dropped first. Second, for the overhearing queue, a flow-based management is adopted. In DCAR-VOH+, if there is coding opportunity for a flow, all packets from this flow in the overhearing queue might need to participate in VOH process or be used for packet decoding. Thus, priority is given to those packets of such a flow in the overhearing queue. In particular, those packets will not be removed until the queue is full.

5.2. Complexity of DCAR-VOH+

We now analyze the communication complexity of DCAR-VOH+ in terms of the number of control messages. We find that DCAR-VOH+ has the same communication complexity as DCAR. The reasons are as follows.

First, the virtual neighbor discovery can be performed without any additional control packets. On one hand, the Hello message is used in both DCAR and DCAR-VOH+, and the broadcast of a new flow ID in DCAR-VOH+ is carried out in the Hello message transmission. On the other hand, a new created OHR can be piggybacked onto the data packet and then overheard by the upstream nodes of a flow. In Fig. 3, when node C sends f_1's packets, it can piggyback the new created OHR onto f_1's packets, and let its upstream node on f_1 overhear the OHR. Later, node C's upstream nodes that overhear the new OHR can do the same to let all node C's upstream nodes on f_1 be informed of this OHR. Note that this method also applies to both the new VNR reporting for coding opportunity creation and the packet encoding control in the adaptive encoding mechanism. Second, the communication complexity of the flow route discovery in DCAR-VOH+ is the same as that in DCAR, because both adopt the same route request and route reply as DSR. Third, the VOH initiation also does not need any new control packet. In Fig. 4, the information carried in the two steps can actually be piggybacked onto the data packets, e.g., the VNR information in Step 1 can be piggybacked onto f_3's packets. This method also applies to the side information requesting in the selective VOH mechanism. Therefore, DCAR-VOH+ has the same communication complexity as DCAR in terms of the number of control packets.

6. Simulation results and discussion

This section evaluates the performance of the two proposed routing protocols, DCAR-VOH and DCAR-VOH+, in different network scenarios by simulation in NS-2. The two well-known network coding schemes, COPE [6] and DCAR [13], are used as benchmarks. In all simulations, IEEE 802.11 standard is adopted in the MAC layer. The data transmission rate is fixed at 1 Mbps. The radio transmission range is 100 m, and the carrier sensing range is 250

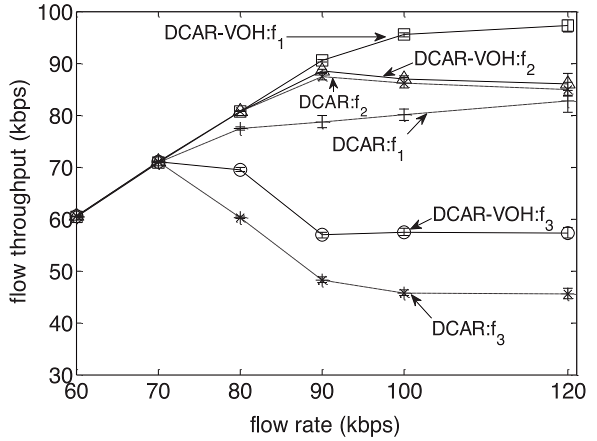


Fig. 7. Flow throughput under CBR traffic.

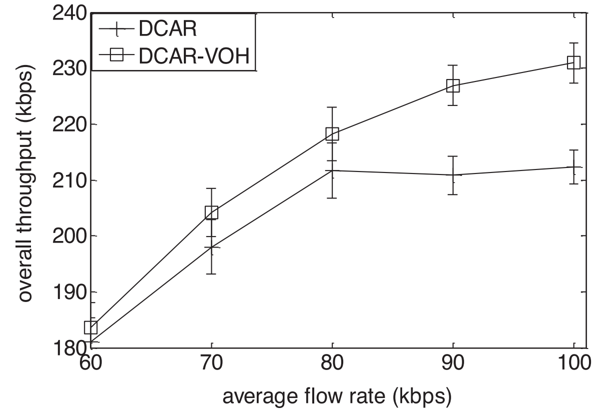


Fig. 8. Overall throughput under bursty traffic.

m. The lengths of the interface queue, the overhearing queue, and the decoding waiting queue are set at 50 packets, 500 packets, and 1500 packets, respectively. The threshold N_{UN} for encoding deactivation is set at 200 packets. The Hello broadcast interval is 1 s. The User Datagram Protocol (UDP) is used for each flow. We run each simulation for 250 s, and for each simulation setting, we have at least 10 runs or replicates to obtain the average result(s). We also plot the 90% confidence interval for the simulation results shown in the figures of this section.

6.1. Performance of DCAR-VOH in the illustrative topology

This subsection studies how DCAR-VOH responds to several important network parameters, including the traffic type, the flow rate, the link quality, and the virtual overhearing distance, in the illustrative topology of Fig. 1(b). For this study, we assume the three flows, f_1, f_2, and f_3, enter the network in time order so that node M is identified as an encoding node in DCAR-VOH.

A. Impact of the traffic type

We first study the impact of the traffic type on DCAR-VOH. Both the constant-bit-rate (CBR) traffic and the bursty traffic are considered for the source nodes in Fig. 1(b).

Case of CBR traffic: For this study, we assume all the three flows have the same flow rate. Fig. 7 shows the throughput of each flow in DCAR-VOH and DCAR at different flow rates. First, we see from Fig. 7 that at a high flow rate of 90 kbps to 120 kbps, the throughput of f_1 and f_3 in DCAR-VOH is about 21% and 26% higher than that in DCAR, respectively. Supported by VOH, network coding at node M between f_1 and f_3 improves the throughput of both f_1 and f_3 in DCAR-VOH. Especially, we find in this study that most coded packets of f_3 are decoded shortly after arriving at the decoding node G (or, the adaptive encoding is not triggered at all). By contrast, DCAR cannot find any coding opportunity in Fig. 1(b), thus resulting in a lower throughput. Second, at a low flow rate (<70 kbps), DCAR and DCAR-VOH perform at the same level for the throughput of f_1 and f_3 because both can handle the light traffic. Third, at all flow rates, the throughput of f_2 in DCAR-VOH remains essentially the same as in DCAR. This shows that VOH does not affect the throughput of f_2, similar to the observation in [9]. Therefore, regarding the overall network throughput, we see that DCAR-VOH improves DCAR at a high flow rate by about 15%.

Case of bursty traffic: we here consider Pareto On/Off traffic [10] as the bursty traffic to study the performance of DCAR-VOH. The key parameters in this traffic model are as follows. The Pareto shape parameter is set to 1.5, and both the average "On" time and "Off" time are set to 500 ms. In this study, we also assume all the three flows have the same rate during burst.

From the simulation results, we find that at medium to high flow rates during burst, DCAR-VOH still outperforms DCAR, similar to the case under CBR traffic. For example, Fig. 8 shows the overall throughput of DCAR-VOH and DCAR at different average flow rates, from which we see that DCAR-VOH outperforms DCAR by about 9% at a high average flow rate of 100 kbps. Note that in our Pareto traffic the average flow rate equals a half of the flow rate during burst. This throughput improvement, however, is not as high as that under CBR traffic, where the improvement is about 15% at the flow rate of 100 kbps. The reason is as follows. Given a short time interval, the possibility that both node A and node D, i.e., the source nodes of f_1 and f_3, have packets to send is low under bursty traffic, whereas the two nodes always have packets to send under CBR traffic. Consequently, the chance for the encoding node M of f_1 and f_3 to perform network coding during that time interval is lower under bursty traffic. Indeed, our study shows that at the average flow rate of 100kbps, the number of the coded packets at node M is 11 per second on average under bursty traffic, whereas it is 14 under CBR traffic. Evidently, the lower improvement in DCAR-VOH in Fig. 8 results from the reduced coding opportunities at node M.

The authors in [19] propose to delay a non-coded packet transmission at an intermediate node for more coding opportunities, and they show that a small delay benefits TCP flows. Aiming to increase coding opportunities at node M under bursty traffic, we have conducted simulations with different delay times at node M. We find that the overall throughput gain from this strategy is limited, $<2\%$, for UDP flows. The reason is that the randomness of bursty UDP traffic is not affected by network coding at all whereas TCP traffic (e.g., the flow rate) can be affected. For more coding opportunities under bursty UDP traffic, we may consider applying traffic regulation (e.g., to redistribute the traffic more evenly in time as in CBR traffic) at the source nodes in DCAR-VOH. Hereafter, for all our simulation studies, we use CBR traffic.

B. Impact of the flow rate of f_2

We now study how the flow rate of f_2 affects the performance of DCAR-VOH in Fig. 1(b). To see the effect clearly, we deactivate the adaptive encoding mechanism in DCAR-VOH. For this study, both f_1 and f_3 are fixed at a high flow rate, 110 kbps, to enable plenty of coding opportunities at node M, and the flow rate of f_2 varies from 20 kbps to 100 kbps.

Fig. 9 compares the flow throughput between DCAR-VOH and DCAR. First, we see that regardless of the flow rate of f_2, the throughput of f_1 in DCAR-VOH is higher than that in DCAR by

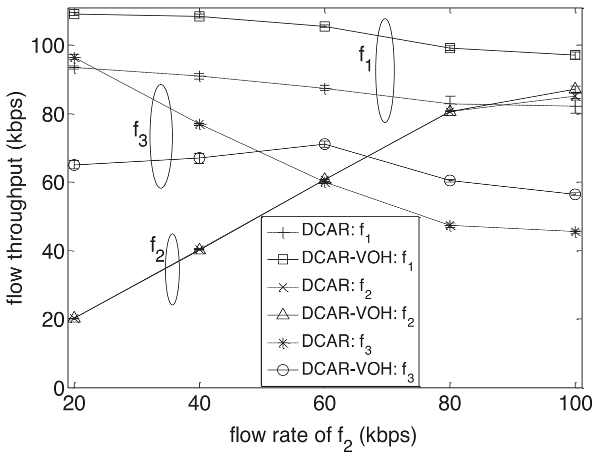


Fig. 9. Flow throughput at different flow rates of f_2.

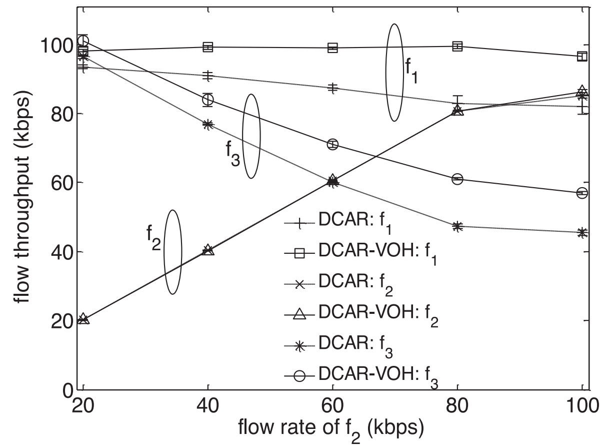


Fig. 10. Performance of DCAR-VOH with the adaptive encoding mechanism.

about 23%. This is because with network coding at node *M* in DCAR-VOH, whenever the channel is occupied by node *M*, the packets of f_1 can be transmitted immediately, whereas in DCAR those packets cannot be sent immediately because they need to compete with packets of f_3. Second, similar to Fig. 7, VOH has no impact on the throughput of f_2. Third, interestingly, we see that (1) at a low flow rate of f_2, the throughput of f_3 in DCAR is higher than that in DCAR-VOH; (2) but at a high flow rate of f_2, the situation is just the opposite. The reasons are as follows. Due to high flow rates of f_1 and f_3, most packets of f_3 received at node *G* in DCAR-VOH are coded packets. To decode them, node *G* needs to virtually overhear the packets of f_1. Unfortunately, many coded packets cannot be decoded at a low flow rate of f_2, because fewer packets of f_1 are virtually overheard by node *G*. Table 1 shows that up to 70% of the coded packets of f_3 cannot be decoded at the low flow rate of f_2, 20 kbps. Thus, the decoding failure decreases the throughput of f_3 in DCAR-VOH. By contrast, at a high flow rate of f_2, as shown in Table 1, node *G* successfully decodes most coded packets of f_3 because node *G* can virtually overhear the packets of f_1 quickly at a high flow rate of f_2. Hence, at a high flow rate of f_2, the throughput of f_3 in DCAR-VOH benefits from the coding opportunities at node *M*.

Table 1 also shows the size requirement of the decoding waiting queue and the average decoding delay of the decoded packets in DCAR-VOH. Here, the required queue size is the minimal queue size for not dropping any coded packet in the decoding waiting queue. We see from the table that a higher flow rate of f_2 speeds up the VOH process and the packet decoding at node *G*, yielding a low decoding delay, and a low requirement for the decoding waiting queue size as well. Especially, we can find that at a high flow rate of f_2 (from 80 kbps to 100 kbps) the average decoding delay is rather limited compared to the average packet delivery delay in DCAR-VOH. Also, note that at a low flow rate, 20 kbps, of f_2, the decoding delay decreases. This is because at low flow rates of f_2, only a part of coded packets are decoded and their decoding is usually done in short time due to 'the selective VOH' mechanism.

In fact, if the adaptive encoding mechanism is activated, DCAR-VOH will benefit from f_2, even at low flow rates of f_2. Fig. 10 shows the achieved flow throughput after the mechanism activation. Let us focus on f_3 for simplicity. First, at low flow rates of f_2, f_3 in DCAR-VOH now achieves higher throughput than in DCAR. The reason is as follows. With the packet encoding controlled at node *M*, most coded packets in DCAR-VOH now can be successfully decoded, e.g., our studies show that no more than 50 coded packets of f_3 are un-decodable at node *G* when the flow rate of f_2 is 40 kbps. In this situation, the packet encoding at node *M* becomes beneficial to f_3. Second, at high flow rates of f_2 (from 60 to 100 kbps) in Fig. 10, the throughput of f_3 in DCAR-VOH is the same as that in Fig. 9, because the adaptive encoding mechanism is not triggered at all in this situation.

C. Impact of the virtual overhearing distance

Now we study how the virtual overhearing distance impacts DCAR-VOH in Fig. 1(b). To see the effect, we also deactivate the adaptive encoding mechanism. Note that in Fig. 1(b) the virtual overhearing distance between node *G* and node *A* on f_1 is 1 hop. To have different virtual overhearing distances for our study, we first properly move node *F* to let it be a neighbor of node *E*, and then we choose node *F* as the destination node of f_3 to have a two-hop virtual overhearing distance. Intuitively, a longer virtual overhearing distance incurs a longer decoding delay at node *G*, thus decreasing the performance of DCAR-VOH. However, the study in Fig. 11 shows that the impact on DCAR-VOH from increasing the virtual overhearing distance is very limited.

We evaluate the performance of DCAR-VOH under the above two virtual overhearing distances with the following network settings. We fix f_1 and f_3 at a high flow rate, 110 kbps, and vary the

Table 1
The statistics of the packet decoding at node *G* on f_3.

Flow rate of f_2	Flow rate of f_1 = Flow rate of f_3 = 110 kbps; Simulation time = 250 s			
	Undecoded packet ratio[†]	Average decoding delay	Average packet delivery delay	Required Decoding waiting queue size
20 kbps	2684 / 3841 = 70%	0.36 s	1.03 s	2684 packets
40 kbps	1432 / 3817 = 37%	0.52 s	1.47 s	1432 packets
60 kbps	162 / 3661 = 4%	0.76 s	1.71 s	167 packets
80 kbps	11 / 3378 = 0.3%	0.22 s	2.01 s	23 packets
100 kbps	14 / 3289 = 0.4%	0.19 s	3.21 s	24 packets

[†]undecoded packet ratio = the number of undecoded packets over the number of coded packets in total

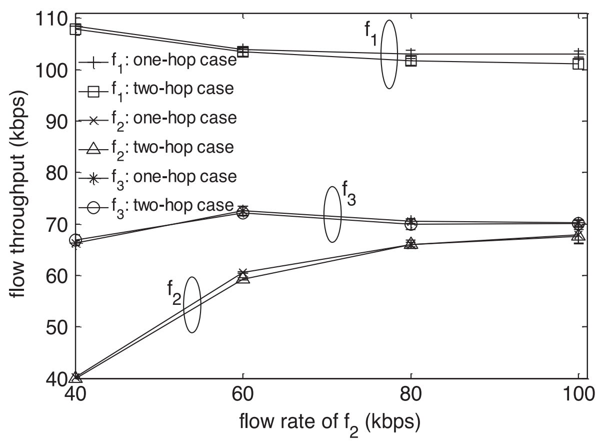


Fig. 11. Flow throughput under different virtual overhearing distances.

flow rate of f_2 from low to high values. Fig. 11 shows the flow throughput in DCAR-VOH for this study, from which we see that the throughput of f_1, f_2, or f_3 is essentially the same under the two virtual overhearing distances. Let us focus on explaining why f_3 achieves the same throughput under different virtual overhearing distances.

Table 2 shows the statistics of the packet decoding in DCAR-VOH. We see from the table that at any given flow rate of f_2, the average decoding delay incurred under the two-hop virtual overhearing distance is always larger than that under the one-hop case. Nonetheless, the impact of this longer delay is very limited, because the un-decoded packet ratio, a parameter that is critical to the performance of DCAR-VOH, in the two cases are very close. Also, we see that the increase of the required decoding waiting queue size in the two-hop case is limited compared with the one-hop case. Hence, Table 2 shows that overall speaking, the virtual overhearing distance has limited impact on DCAR-VOH. In fact, a similar observation is obtained if the adaptive encoding mechanism is activated in this study.

D. Impact of the link quality

Lastly, we study how DCAR-VOH responds to the link quality in Fig. 1(b). The simulation is configured as follows. Node *G* is chosen as the destination node of f_3. Both the flow rates of f_1 and f_3 are fixed at a high value, 100 kbps, and the flow rate of f_2 is also fixed at a relatively high value, 80 kbps. In 802.11 MAC, the maximal transmission attempts for a data packet is 4 by default [20]. For simplicity, all links are assumed to have the same packet delivery probability (PDP).

We focus on the throughput of f_2 and f_3 in DCAR-VOH. We find from the simulation results (not shown as figures here) that (1) at any PDP, VOH does not impact f_2, i.e., f_2 achieves the same throughput in both DCAR-VOH and DCAR, and (2) the throughput improvement of f_3 in DCAR-VOH over DCAR is limited as PDP becomes lower. The reason for (2) is that at a lower PDP, more coded packets of f_3 in DCAR-VOH cannot be decoded at node *G* due to the loss of the side information, as shown in Table 3. Note that the adaptive encoding mechanism is not disabled in this study. The side information loss occurs when node *H* fails to overhear the packet sent by node *A* on f_1 or when node *G* fails to overhear the coded packet sent by node *H*. The loss in the former case (in the latter case) is termed the overhearing loss (the VOH loss). Table 3 shows that the overhearing loss is the major cause for (2). This is because if node *H* fails to overhear the packet of f_1, it will miss it forever; whereas if node G fails to overhear a coded packet, e.g., $P_1 \oplus P_2$ in Fig. 1(b), from node *H*, it will keep informing node *H* to mix P_1 from f_1 until it obtains P_1, thanks to 'the selective VOH' mechanism. Hence, to improve DCAR-VOH under lossy links, we may only need to improve the reliability of direct overhearing, e.g., from node *A* to node *H* in Fig. 1(b). This is actually a common problem as it also exists in traditional network coding design [11]. The authors in [9] actually have made the similar observations of (1) and (2).

Table 3
Statistics related to the performance of f_3 in DCAR-VOH under lossy links.

PDP	Simulation time = 250 s		
	Number of coded packets	Number of undecoded packets at node *G*	
		Overhearing loss (packets)	VOH loss (packets)
0.95	1392	61	14
0.90	1384	115	24
0.85	1364	132	33

Therefore, our studies in this section under various network conditions show that if the overhearing loss is resolved (e.g., through the transmission rate adaptation [11]), more benefits in DCAR-VOH can be achieved by performing VOH on a flow particularly with high throughput.

6.2. DCAR-VOH+ in the illustrative topology

We now study how DCAR-VOH+ can improve DCAR-VOH in the scenario of Fig. 5. The three flows in Fig. 5 are randomly generated at different times with the same flow rate. We assume all the links in Fig. 5 are reliable.

Fig. 12 plots the overall throughput of DCAR-VOH+, DCAR-VOH, and DCAR. Not surprisingly, DCAR-VOH+ is the best performer among them. At a high flow rate (i.e., > 80 kbps), DCAR-VOH+ improves the overall throughput over DCAR and DCAR-VOH by 14% and 9%, respectively. The main reason for this improvement is that all the three intersection nodes in Fig. 5, i.e., nodes *A, C*, and *F*, are identified as encoding nodes in DCAR-VOH+, whereas only *one* node (node *A, C* or *F*, depending on the establishment order of the three flows) can be an encoding node in DCAR-VOH and no encoding node is identified in DCAR. Hence, DCAR-VOH+ benefits from more coding opportunities that are identified.

Table 2
The statistics of the packet decoding at node *G* under different virtual overhearing distances.

Flow rate of f_2	Flow rate of f_1 = flow rate of f_3 = 110 kbps; Simulation time = 250 s					
	Undecoded packet ratio		Average decoding delay		Required decoding waiting queue size	
	one-hop case	two-hop case	one-hop case	two-hop case	one-hop case	two-hop case
40 kbps	38.2%	40.4%	0.51 s	0.84 s	1515 packets	1648 packets
60 kbps	6.8%	7.0%	1.15 s	1.5 s	271 packets	303 packets
80 kbps	3.5%	5.7%	0.77 s	1.68 s	135 packets	240 packets
100 kbps	1.3%	1.4%	0.58 s	0.95 s	57 packets	70 packets

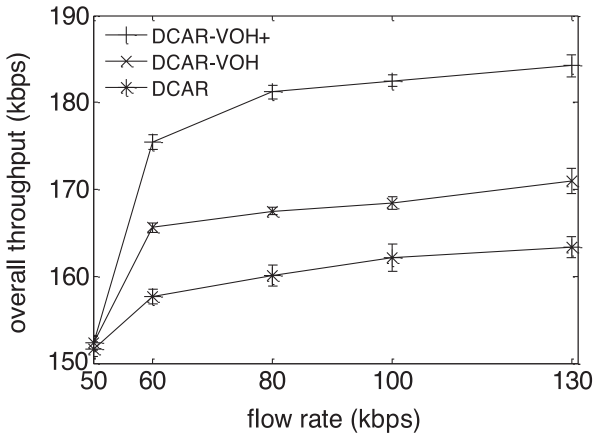


Fig. 12. Overall throughput of the three flows in Fig. 5.

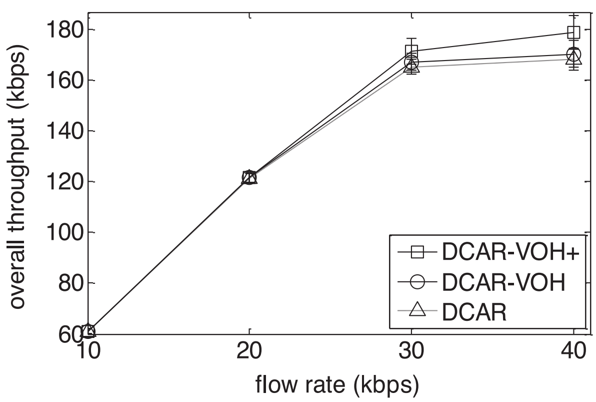


Fig. 13. Overall throughput in the random network.

6.3. DCAR-VOH+ in random networks with random traffic

In this simulation, we are interested in studying how DCAR-VOH+ can improve DCAR-VOH and DCAR in a random network, i.e., under random traffic in a random topology. Thirty nodes are randomly placed in a 400 m × 400 m network, and six flows are randomly generated at different times with equal flow rates. All the flows are assumed to have different source nodes (and the destination nodes as well), and the length of each flow is at least two hops as both VOH and network coding cannot be performed on a one-hop flow.

Fig. 13 plots the overall throughput of these three schemes, where we see that DCAR-VOH+ outperforms the other two schemes and improves the performance of DCAR by about 6% at high flow rates. Due to the second mode of VOH usage, DCAR-VOH+ can find the most coding opportunities, thus improving network performance. By contrast, the first mode of VOH usage in DCAR-VOH finds very few extra coding opportunities in a random network compared with DCAR, yielding essentially the same performance level as DCAR.

6.4. DCAR-VOH+ under traffic with certain patterns

Apart from the study under random traffic flows in Section 6.3, here we study two typical traffic patterns, the inter-cluster traffic and the three-node cyclic traffic, which will favor both DCAR-VOH and DCAR-VOH+.

Case of the inter-cluster traffic: This traffic pattern assumes in a network that (1) there are a number of separate clusters, with each being formed by some nearby nodes, and (2) the source and the destination of each flow are selected from two different clusters. Fig. 5 shows an example of the inter-cluster traffic. The nodes in each dashed circle in Fig. 5 form a cluster, and each flow is established from one cluster to another. We consider the inter-cluster traffic in the random network to study DCAR-VOH+ and DCAR-VOH. The network is configured as follows. In a 400 m × 400 m random network, four cluster regions are formed at four corners of the network area. We assume that each cluster has a relatively small region, a 150 m × 150 m square, to represent some traffic hotspot, and that any two cluster regions are beyond one hop, similar to the case in Fig. 5. A total of 30 nodes are uniformly distributed in the network and the nodes located in one cluster region form a cluster. Then, six inter-cluster flows are randomly generated in the network at different times. Fig. 14 compares the overall throughput of the four protocols including COPE, and we have the following observations.

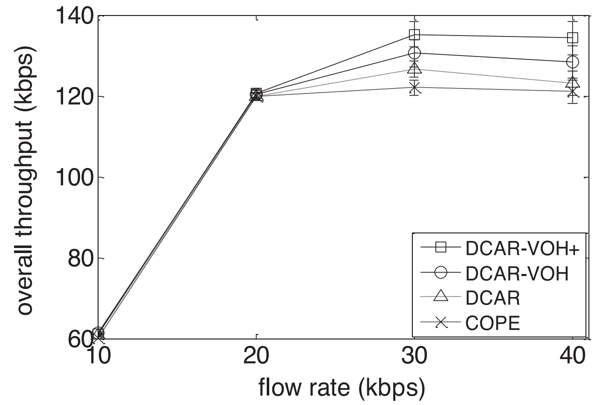


Fig. 14. Overall throughput in the random network with inter-cluster traffic.

First, DCAR outperforms COPE by 3% and DCAR-VOH outperforms DCAR by 5%. Obviously, the reason for this is that VOH enables DCAR-VOH to identify more coding opportunities than DCAR in the inter-cluster traffic. To see how DCAR-VOH can find more coding opportunities, we refer to Fig. 5 where three inter-cluster flows are generated. Two features exist in this traffic pattern: (1) for two nodes within one cluster on two flows, the possibility that one can directly overhear the other is high due to the small area; and (2) for two nodes in different clusters on two flows, the possibility that one can directly overhear the other becomes low. Referring to the network coding conditions in DCAR, we see that it is difficult in this scenario to meet these conditions due to feature (2), as shown in Fig. 5. By contrast, with VOH existing between any two intersecting flows in Fig. 5, it is highly possible that DCAR-VOH can find coding opportunities under this inter-cluster traffic. For example, if f_1 and f_2 are the first two established flows, then, f_3, during its establishment, will find coding opportunities at node 3 with f_1. Therefore, the inter-cluster traffic favors DCAR-VOH in identifying coding opportunities, and thus DCAR-VOH outperforms DCAR.

Second, DCAR-VOH+ improves the performance of DCAR by 10%. As illustrated earlier in Fig. 5, all the three intersection nodes are encoding nodes in DCAR-VOH+ whereas only one among the three is identified as an encoding node in DCAR-VOH. Hence, the inter-cluster traffic favors DCAR-VOH+ more than DCAR-VOH.

Case of the three-node cyclic traffic: This traffic pattern considers three nodes to form a traffic group and send packets in a cyclic form. Suppose in a traffic group the three nodes are nodes *A*, *B*, and *C*, then, a cyclic traffic pattern is formed when packets are sent from node *A* to node *B*, from node *B* to node *C*, and from node *C* to node *A*. To study the performance of DCAR-VOH+ and DCAR, we consider this cyclic traffic in a 5 × 5 grid topology as shown in Fig. 15. Each node in this topology can directly commu-

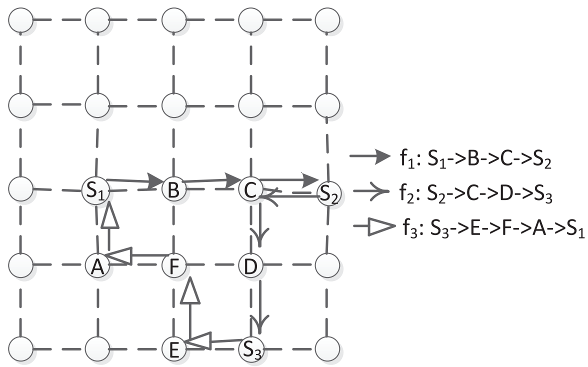


Fig. 15. 5 × 5 grid topology.

Table 4
Throughput of DCAR and DCAR-VOH+ for different numbers of traffic groups.

Total number of traffic groups	Flow rate	DCAR	DCAR-VOH+	Improvement
1	100 kbps	227.5 kbps	255.8 kbps	12.4%
2	50 kbps	203.6 kbps	219.1 kbps	7.6%
3	36 kbps	194.1 kbps	205.4 kbps	5.8%

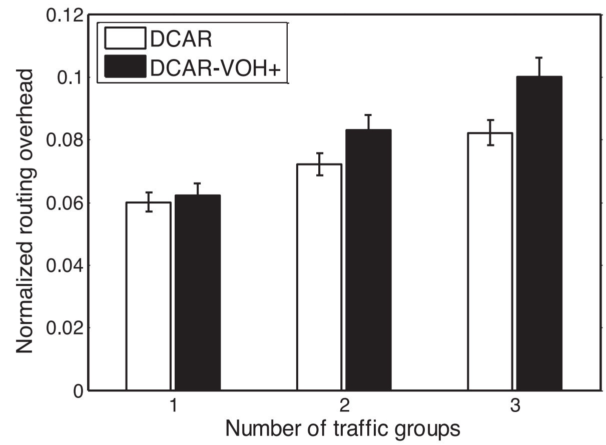


Fig. 16. Normalized routing overhead in DCAR-VOH+ and DCAR.

nicate with its eastern, southern, western, and northern neighbors, if any. We will generate different numbers of such traffic groups in total to study the scalability of DCAR-VOH+ in Fig. 15. In each generated group, the three nodes are randomly chosen from the grid network in Fig. 15.

Table 4 compares the overall throughput between DCAR-VOH+ and DCAR when the network is saturated. First, DCAR-VOH+ improves the performance of DCAR for any number of traffic groups. This is because DCAR-VOH+ can find more coding opportunities under the three-node cyclic traffic. Especially, if there is an intermediate node relaying traffic for any two nodes of a traffic group, then it must be an encoding node in DCAR-VOH±. Consider a scenario in Fig. 15 where there is an intermediate node, say node C, serving as a relay node for both the source nodes S_1 and S_2. In DCAR, node C may not be an encoding node. However, node C must be an encoding node in DCAR-VOH±: with the flow established from node S_3 to node S_1, node S_3 can virtually overhear node S_1 and thus, node C is allowed to mix the packets from nodes S_1 and S_2. This is why the cyclic traffic favors DCAR-VOH±. Second, as the total number of traffic groups (or the number of flows) increases, the performance improvement of DCAR-VOH+ over DCAR decreases. In fact, this trend has been observed in COPE [6], and the reason is that as the network gets more congested due to the injection of more traffic groups or flows, the coding opportunities are reduced and thus, the positive effect of network coding diminishes.

Here, we present an evaluation of the routing overhead in DCAR-VOH+ and DCAR. The routing overhead includes all control packets (i.e., those mentioned in Section 5) except those used in the MAC layer. In our implementation in NS2, the MAC header has a fixed size, 58 bytes, and the node address is composed of 4 bytes. For a fair comparison, we count the overhead in terms of the number of bytes that are transmitted in total, and we look at the normalized routing overhead (defined as the ratio of the total overhead in bytes over the total data bytes received at destinations) [13].

Fig. 16 compares the normalized routing overhead between DCAR-VOH+ and DCAR under the three-node cyclic traffic in the network of Fig. 15. We see that with more flows or traffic groups, the normalized overhead in DCAR-VOH+ increases faster than in DCAR. However, as can be seen from Table 4, the increase of the overhead still brings performance gains in DCAR-VOH+. Moreover, Fig. 16 also shows that the overhead relative to the received data packets is actually very low in both schemes.

7. Conclusions and areas for future improvement

In this paper, we took advantage of a recently proposed new technique called virtual overhearing (VOH) to benefit the routing performance in multi-hop wireless ad-hoc networks. We presented two modes of utilizing VOH and designed two network coding-aware routing protocols in this paper. The first protocol, DCAR-VOH, makes a new incoming flow aware of existing VOH during the route discovery and allows it to find a route with coding opportunities. The second protocol, DCAR-VOH+, makes further use of VOH to *create* coding opportunities between two established flows which are currently unmixable. As a result, there are many more coding opportunities in DCAR-VOH+ compared with DCAR-VOH. Furthermore, we proposed an adaptive encoding mechanism for the two routing protocols to ensure all coded packets could be decoded.

Extensive simulation results showed that in different network scenarios both DCAR-VOH and DCAR-VOH+ can discover or create extra coding opportunities compared with the two well-known network coding schemes, DCAR and COPE, resulting in improved network performance. In particular, a network with either the inter-cluster traffic or the three-node cyclic traffic favors both DCAR-VOH+ and DCAR-VOH, and with the two modes of VOH usage implemented, DCAR-VOH+ generally outperforms DCAR-VOH.

To further improve DCAR-VOH and DCAR-VOH+, the future work can be as follows.

Incorporating the VOH rate into the routing metric: Here, the VOH rate refers to the rate that a decoding node virtually overhears packets, and it determines how many coded packets can be decoded. In both DCAR-VOH and DCAR-VOH+, the route quality is solely determined by the modified interface queue (MIQ) length of each node, which does not take the VOH rate into account. As shown in our studies, a low VOH rate could be detrimental or only yields limited performance gains. Hence, a better routing metric should take the VOH rate into account.

Improving packet decodability in case of low VOH rate: The adaptive encoding mechanism in DCAR-VOH and DCAR-VOH+ controls the packet encoding to fit a low VOH rate. This begets the question of how to overcome the drawback of a low VOH rate and improve the packet decodability at a decoding node. To tentatively solve this problem, we may let a virtual encoding node, e.g., node H in Fig. 1(b), additionally forward some overheard packets to a decoding node, e.g., node G in Fig. 1(b), to enhance its ability of decoding packets. Obviously, this is achieved at the expense of *additional* traffic injected to the network. We can expect that this tentative solution will work effectively in the situation where the *additional* injected traffic does not compete with existing flows for the use of network resources. However, more investigations are needed for this tentative solution, and we leave it as our future work.

Reducing the decoding delay in DCAR-VOH and DCAR-VOH+: Tables 1 and 2 show that the decoding delay is not small, especially in case of multihop virtual overhearing distance. This poses a challenge for DCAR-VOH and DCAR-VOH+ to render time-sensitive applications. However, we find that it is possible to reduce the decoding delay. Observing that from the packets stored in the interface queue, an encoding node, e.g., node *M* in Fig. 1(b), is able to predict what packets will be encoded in subsequent transmissions, we can design a mechanism to let a virtual encoding node, e.g., node *H* in Fig. 1(b), be informed beforehand of the packets to be encoded. This way, the coded packets, after arriving at the decoding node, are more likely to be decoded immediately. We leave the study of this mechanism as another focus of our future work.

Acknowledgement

This work was supported in part by the Early Career Scheme under Project 25200714 established under the University Grant Committee of the Hong Kong Special Administrative Region, China; by the National Natural Science Foundation of China under Projects 61401384 and 61571250; and by The Hong Kong Polytechnic University under Projects 4-ZZCZ, G-YBK6, and G-YN17.

References

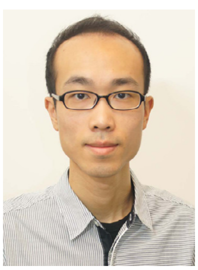
[1] R. Ahlswede, N. Cai, S. Li, R.W. Yeung, Network information flow, IEEE Trans. Inf. Theor. 46 (4) (2000) 1204–1216.
[2] C. Fragouli, J.L. Boudec, J. Widmer, Network coding: an instant primer, in: Proceedings of ACM SIGCOMM, 2006, pp. 63–68.
[3] C. Fragouli, D. Katabi, A. Markopoulou, M. Medard, H. Rahul, Wireless network coding: opportunities & challenges, in: Proceeding of MILCOM, 2007, pp. 1–8.
[4] P.A. Chou, Y. Wu, Network coding for the internet and wireless networks, IEEE Sig. Process. Mag. 24 (5) (2007) 77–85.
[5] S. Li, R.W. Yeung, N. Cai, Linear network coding, IEEE Trans. Inf. Theor. 49 (2) (2003) 371–381.
[6] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Medard, J. Crowcroft, XORs in the air: practical wireless network coding, Proceeding of ACM SIGCOMM, Italy, 2006.
[7] S. Omiwade, R. Zheng, C. Hua, Butterflies in the mesh: lightweight localized wireless network coding, Proceeding of the Workshop on Network Coding, Theory and Applications, 2008.
[8] A. Eryilmaz, D. Lun, Control for inter-session network coding, in: Proceedings of the Workshop on Network Coding, Theory and Applications, 2007.
[9] L.F. Xie, P.H.J. Chong, Y.L. Guan, Performance analysis of network coding with virtual overhearing in wireless networks, IEEE Trans. Vehicular Technol. 64 (5) (2015) 2051–2061.
[10] K. Fall, K. Varadhan, The ns manual, The VINT Project, UC Berkeley, LBL, USC/ISI, and Xerox PARC, 2011.
[11] T. Kim, et al., Realizing the benefits of wireless network coding in multi-rate settings, IEEE/ACM Trans. Netw. 21 (3) (2013) 950–962.
[12] S. Sengupta, S. Rayanchu, S. Banerjee, Network coding-aware routing in wireless networks, IEEE/ACM Trans. Network. 18 (4) (2010) 1158–1170.
[13] J. Le, C.S. Lui, D. Chiu, DCAR: distributed coding-aware routing in wireless networks, IEEE Trans. Mob. Comput. 9 (4) (2010) 596–608.
[14] M. Jhang, S. Lin, W. Liao, C2AR: coding and capacity aware routing for wireless ad hoc networks, in: Proceeding of ICC, 2010, pp. 1–5.
[15] L.F. Xie, P.H.J. Chong, S.C. Liew, Y.L. Guan, CEO: consistency of encoding and overhearing in network coding-aware routing, IEEE Wireless Commun. Lett. (2013) 187–190.
[16] Y. Peng, Y. Yang, X. Lu, X. Ding, Coding-aware routing for unicast sessions in multi-hop wireless networks, in: Proceeding of Globecom, Miami, USA, 2010, pp. 1–5.
[17] B. Guo, H. Li, C. Zhou, Y. Cheng, Analysis of general network coding conditions and design of a free-ride-oriented routing metric, IEEE Trans. Vehicular Technol. 60 (4) (2011) 1714–1727.
[18] Z. Zhou, L. Zhou, Network joint coding-aware routing for wireless ad hoc networks, in: Proceeding of WCNIS, 2010, pp. 17–21.
[19] Y. Huang, M. Ghaderi, D. Towsley, W. Gong, TCP performance in coded wireless mesh networks, in: Proceeding of IEEE SECON, 2008, pp. 179–187.
[20] X. Li, P. Kong, K. Chua, TCP performance in IEEE-802.11-based ad hoc networks with multiple wireless lossy links, IEEE Trans. Mob. Comput. 6 (12) (2007) 1329–1342.

Ling Fu Xie received his B. Eng. and M. Eng. in communications engineering from University of Electronic Science and Technology of China in 2006 and 2009, respectively, and his PhD degree from Nanyang Technological University, Singapore, in 2014. From 2014 to 2015, he worked as a post-doctoral fellow in The Hong Kong Polytechnic University, Hong Kong. In Oct. 2015, he joined the Faculty of Electrical Engineering and Computer Science of Ningbo University, Ningbo, China. His research interests mainly include protocol design and performance analysis in mobile networks, wireless network coding, and physical-layer network coding.

Peter H. J. Chong received the B.Eng. (with distinction) in electrical engineering from the Technical University of Nova Scotia, Halifax, NS, Canada, in 1993, and the M.A.Sc. and Ph.D. degrees in electrical engineering from the University of British Columbia, Vancouver, BC, Canada, in 1996 and 2000, respectively.

Between July 2000 and January 2001, he worked in the Advanced Networks Division at Agilent Technologies Canada Inc., Vancouver, BC, Canada. From February 2001 to May 2002, he was with the Radio Communications Laboratory at Nokia Research Center, Helsinki, Finland, and was involved in research on WCDMA and standardization for HSDPA. During his stay in Finland, he has taught part of a graduate course in WCDMA at the Helsinki University of Technology, Helsinki, Finland. From 2002 to 2016, he was with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore, as an Associate Professor (Tenured). He was an Assistant Head of Division of Communication Engineering between 2011 and 2013. Between 2013 and 2016, he was a Director of Infinitus, Centre for Infocomm Technology. In April 2016, he joins Auckland University of Technology, New Zealand, as a Full Professor and becomes the Head of Department of Electrical and Electronic Engineering. He has visited Tohoku University, Japan, as a Visiting Scientist in 2010 and Chinese University of Hong Kong (CUHK), Hong Kong, between 2011 and 2012. He is currently an Adjunct Faculty of CUHK.

He was a Technical Program Committee Chair for Mobility Conference 2005 and 2006, a Chair of Mobility Conference 2007 and 2008, a TPC Co-Chair of IEEE International Conference on Networks (ICON) 2012 and General Chair of International Conference on Information, Communications, and Signal Processing (ICICS) 2013. He served as a Guest Editor of *Journal of Internet Technology* in 2006, *International Journal of Ad Hoc and Ubiquitous Computing* in 2007 and lead Guest Editor of *IEEE Communications Magazine* for the Feature Topic on 'Technologies in Multihop Cellular Network' in 2007 and *IEEE Wireless Communications* for the Feature Topic on 'Technologies for Green Radio Communication Networks' in 2011. He is an Editorial Board Member of Security and Communication Networks, International Journal of Wireless Communications and Networking, Wireless Sensor Network, and an Editor of Far East Journal of Electronics and Communications, and KSII Transactions on Internet and Information Systems. His research interests are in the areas of mobile communications systems including radio resource management, multiple access, MANETs, multihop cellular networks and green radio networks.



Ivan Wang-Hei Ho received the BEng and MPhil degrees in information engineering from The Chinese University of Hong Kong in 2004 and 2006, respectively, and the PhD degree in electrical and electronic engineering from Imperial College London, United Kingdom, in 2010.

He is currently a research assistant professor at the Department of Electronic and Information Engineering, The Hong Kong Polytechnic University. He was on the Mobile Environmental Sensing System Across a Grid Environment (MESSAGE) project funded by EPSRC and the Department for Transport, United Kingdom, and the International Technology Alliance (ITA) project funded by the US Army Research Laboratory and United Kingdom Ministry of Defence during his PhD studies. In 2007, he spent a summer with the IBM T.J. Watson Research Center, Hawthorne, New York. After his PhD graduation, he was with the System Engineering Initiative at Imperial College London as a postdoctoral research associate. In September 2010, he cofounded P2 Mobile Technologies Limited at Hong Kong Science Park and served as the chief R&D engineer. The MeshRanger series wireless mesh embedded system primarily invented by him won the Silver Award in Best Ubiquitous Networking in Hong Kong ICT Awards 2012. He is also the holder of a US patent.

His research interests are in wireless communications and networking, specifically in vehicular ad hoc networks and intelligent transportation systems, physical-layer network coding, and wireless mesh networks. His publications can be found at http://www.eie.polyu.edu.hk/~whho.

Henry C. B. Chan received his B.A. and M.A. degrees from the University of Cambridge, England, and his Ph.D. degree from the University of British Columbia, Canada. In August 1998, he joined The Hong Kong Polytechnic University (PolyU), where he is now an associate professor in the Department of Computing. His research interests include networking/communications, cloud computing, Internet technologies, and electronic commerce. He was the Chair (2012) of the IEEE Hong Kong Section, and the Chair (2008-2009) of the IEEE Hong Kong Section Computer Society Chapter.

He was the recipient of the 2015 IEEE Computer Society Computer Science and Engineering Undergraduate Teaching Award ("for outstanding contributions to computing education through teaching, mentoring students, and service to the education community"). At PolyU, he received three President's Awards and five Faculty Awards.