Cluster-based Cooperative Data Forwarding with Multi-radio Multi-channel for Multi-flow Wireless Networks

Cherry Ye Aung[1], G. G. Md. Nawaz Ali[1] and Peter Han Joo Chong[2]
[1]School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore
[2]Department of Electrical and Electronic Engineering, Auckland University of Technology, New Zealand
[email: cherryaung@ntu.edu.sg, nawaz.ali@ntu.edu.sg and peterhjchong@gmail.com]
*Corresponding author : Peter Han Joo Chong

Received April 26, 2016; revised August 4, 2016; revised September 22, 2016; accepted October 17, 2016; published December 31, 2016

Abstract

Cooperative forwarding has shown a substantial network performance improvement compared to traditional routing in multi-hop wireless network. To further enhance the system throughput, especially in the presence of highly congested multiple cross traffic flows, a promising way is to incorporate the multi-radio multi-channel (MRMC) capability into cooperative forwarding. However, it requires to jointly address multiple issues. These include radio-channel assignment, routing metric computation, candidate relay set selection, candidate relay prioritization, data broadcasting over multi-radio multi-channel, and best relay selection using a coordination scheme. In this paper, we propose a simple and efficient cluster-based cooperative data forwarding (CCDF) which jointly addresses all these issues. We study the performance impact when the same candidate relay set is being used for multiple cross traffic flows in the network. The network simulation shows that the CCDF with MRMC not only retains the advantage of receiver diversity in cooperative forwarding but also minimizes the interference, which therefore further enhances the system throughput for the network with multiple cross traffic flows.

Keywords: Cooperative forwarding, opportunistic routing, coordination, multi-radio multi-channel, multi-flow.

http://dx.doi.org/10.3837/tiis.2016.12.002 **ISSN : 1976-7277**

1. Introduction

The capacity of wireless ad hoc networks is mainly limited by co-channel interference and unreliable wireless channels. Co-channel interference can be minimized by using multi-radios [2]. But it requires a proper channel assignment to assign different channels to use in different nodes. A number of works focusing on channel assignment have been done for Traditional deterministic Routings (TRs), such as DSR and AODV, for multi-radio multi-channel wireless ad hoc networks. However, it has been proved that those TRs perform poorly under unreliable and lossy wireless channels due to interference and multipath fading [13]. It has been shown that cooperative forwarding, also known as opportunistic routing, can highly increase the reliability of data communications under lossy wireless channels [14], [16]. In order to improve the system capacity, a promising way is to employ the cooperative forwarding with multi-radio multi-channel (MRMC) functions.

The basic idea of cooperative forwarding is to make use of receiver diversity (i.e. spatial diversity) and provide reliable communications over long, weak wireless links. The fundamental design components of cooperative forwarding include 1) routing metric computation; 2) candidate relay set (CRS) selection; 3) candidate relays prioritization (CRP); 4) data broadcast to candidate relays; and 5) best relay selection using a coordination scheme. Source selects CRS and conducts CRP before forwarding. Timer-based coordination is mostly used in the existing works for transmission scheduling and duplicate transmission avoidance [14], [16], [17]. During the coordination phase, the candidate nodes set a waiting delay proportional to its priority as per CRP. Nodes have to listen higher priority nodes transmission and discard its schedule if the same packet is being forwarded by the higher priority node.

From this coordination nature, we observe that the following problems may arise under multiple traffic flows, especially when the same CRS is being used for different flows.

1. The forwarding node being unable to hear the higher priority node transmission successfully due to the traffic coming from another source and therefore wrongly believes that the higher priority node did not receive the packet, resulting in duplicate transmissions.
2. Having multiple schedules for multiple flows and inaccuracy in scheduling due to the unavailable medium, resulting in forwarding the same packet by different forwarding nodes at the same time.
3. All forwarding nodes being unable to receive the data packet from the source/sender node due to collisions, resulting in unsuccessful data delivery.
4. Unnecessarily waiting for the higher priority node transmission when it is the only node who receives the packet correctly.

Given the advancement of wireless radios equipped with multiple interfaces working on multiple RF channels, it is a promising way to incorporate the multi-radio multi-channel (MRMC) capability into cooperative forwarding to further enhance the system throughput, especially in the presence of highly congested multiple cross traffic flows. A number of works on joint channel assignment and routing in MRMC wireless networks had been proposed for traditional routing mechanisms [3]–[7]. In traditional routing, data forwarding takes place between a sender and a pre-selected next-hop node. No diversity is involved. In channel assignment for traditional routing, a pair of nodes uses orthogonal channel to other neighbouring pairs or a set of nodes for one flow uses orthogonal channel to other nearby flows to minimize the interference and maximize the throughput. However, in cooperative forwarding, it has been said that "multi-carrier utilization reduces interference, but it also reduces the spatial diversity gain offered by cooperative forwarding, since the number of relays operating over a common channel decreases" [17]. In [11], it is also stated that incorporating MRMC into cooperative forwarding can be considered as a tradeoff between multiplexing and spatial diversity and what choice the neighbouring nodes should make is nontrivial. In [10], it is shown that traditional channel assignment does not work well in cooperative forwarding. Therefore, in cooperative forwarding, one has to jointly address multiple issues: radio-channel assignment, CRS selection, CRP, data broadcasting over MRMC, and best relay selection using a coordination scheme.

From these observations, we note that no cooperative forwarding algorithm that can work under multi-flow network where the same CRS is being used for different flows has been proposed. In addition, no performance study on the efficacy of cooperative forwarding under such environment has been conducted. In this paper, we propose a simple and efficient cluster-based cooperative data forwarding (CCDF) algorithm for multi-flow network. We conduct performance evaluation under multiple cross traffic flows and observe that network throughput is reduced and end-to-end delay is increased with the incresing number of traffic flows. To solve this issue, a solution called CCDF with multi-radio multi-channel capability is proposed. We show that by taking the advantage of clustering, mutli-radio mutli channel capability can be easily integrated with the cooperative forwarding mechanism.

From the performance evaluation, we observe that the proposed solution can achieve both multiplexing and spatial diversity. Specifically, CCDF with multi-radio mutli-channel capability reduces about 97% of the delay and achieves about 200% of throughput improvement compared to the single-radio single-channel (SRSC) case. In addition, we also compare the performance of CCDF-MRMC and CCDF-SRSC with the end-to-end routing AODV-MRMC and AODV-SRSC protocols. We observe a significant improvement in terms of the packet delivery ratio, end-to-end delay, throughput, and transmission count for all different traffic flow cases. To further evaluate the effectiveness of proposed cooperative forwarding algorithm, we compare the performance of CCDF-SRSC with an exisitng cooperative forwaring protocol under single-flow network. From the performance study, we note that CCDF provides slightly better throughput performance and lower end-to-end delay over the existing scheme.

The rest of the paper is organized as follows. Section 2 reviews related works on cooperative forwarding in MRMC networks. The network model used in this paper is discussed in Section 3. Section 4 presents the routing metric. The proposed data forwarding scheme with several protocol details is presented in Section 5. Integration of proposed data forwarding with MRMC capability is discussed in Section 6. Section 7 evaluates the efficacy of the proposed scheme with single-radio single-channel and multi-radio multi-channel under multiple cross traffic flows. The concluding remarks are provided in Section 8.

2. Related Work

A number of works on joint channel assignment and routing in MRMC wireless networks have been proposed [3]–[7]. Existing channel assignment schemes are based on assigning a frequency on a link between two nodes or on a series of connected link between source and destination. Whereas for cooperative forwarding, a new channel assignment strategy, which allows broadcast transmission to achieve receiver diversity while mitigating interference and providing multiple concurrent transmissions, is required.

Most existing cooperative forwarding protocols were proposed for single-radio single-channel wireless network [14], [16], [17]. Consequently, research in MRMC cooperative forwarding is still in the very beginning stage [17]. A linear-programming based study on end-to-end throughput of MRMC cooperative forwarding is discussed in [11]. It shows the performance improvement achieved using MRMC cooperative forwarding over the MRMC traditional routing. A single-radio multi-channel cooperative forwarding protocol is proposed in [10]. It uses only one radio and the channels are uniformly assigned to all nodes. The CRS per channel is formed and joint channel and CRS selection is performed to minimize the interference. The forwarding nodes switch the channel as per the destination's channel. However, it did not discuss a strategy on how to guarantee the receiver diversity when the candidate nodes are operating on different channel from the source or sender. A similar protocol but with multiple radios is proposed in [8]. It uses one radio for control purpose and the others for data communications. Random channel assignment is used for each radio interface. Although it ensures at least one candidate receives the data packet, it does not guarantee receiver diversity at most of the potential candidate nodes due to random channel assignment and channel switching. In [12], it studies the possible concurrent transmissions using multi-radio multi-channel. It assigns the same channel to all nodes in a flow and all forwarding nodes belong to the flow, which makes the nodes unavailable for other flows especially when a number of cross flow traffic are present. In [9], it extends a cooperative forwarding protocol to incorporate multi-radio multi-channel. It performs workload-aware channel assignment to candidate nodes and focuses on throughput maximization for single flow. Multiple traffic flows are not considered.

3. Network Model

We consider a network with *C* clusters and *K* orthogonal channels. The algorithm makes use of clustering to support 1) distributed agent-based coordinaion so as to handle multpile cross traffic flows; and 2) cooperative forwarding in multi-radio multi-channel network. In this paper, we focus on cooperative data forwarding under multiple cross traffic flows and then maximizing its throughput using MRMC features. So, for clustering, we assume that the network is already composed of clusters and nodes are organized by clusters according to their geographical/logical relationships. We aslo assume that each cluster comprises member nodes and a head and the link between member and its head has non-zero bi-directional link quality.

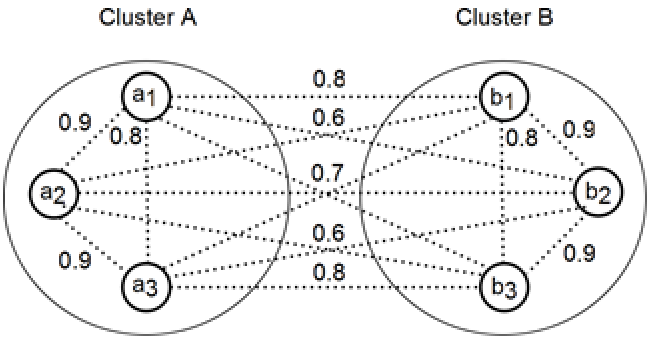


Fig. 1. Link delivery probability *p*.

4. Cluster-to-Cluster ETX (*C2CETX*)

In CCDF algorithm, nodes conduct live-updating link qualities to monitor the unreliable and probabilistic nature of wireless channel and to compute the cluster path for data forwarding.

IP header		
neighbour node ID n_j	p with n_j	p_{rev} with n_j
…		

Fig. 2. Hello packet header format.

IP header	
next-hop cluster ID	destination cluster ID
data relay node ID	ACK-sender ID
…	

Fig. 3. Data packet header format.

Each node broadcasts a hello message periodically, at an average period of λ (we use one second in the performance evaluation). Nodes calculate bi-directional link qualities of the links connecting to the direct contact nodes. This link quality denoted as p is calculated as $p = p_{fwd} * p_{rev}$, where p_{fwd} and p_{rev} are the link quality of forward and reverse links. The reverse link quality p_{rev} with a contact node is calculated based on the received hello packet ratio during the last ω seconds from that node. A node calculates p_{rev} using the equation shown below.

$$p_{rev}(t) = \frac{r(t-\omega,t)}{\omega/\lambda}$$

where $p_{rev}(t)$ denotes the reverse link quality at time t, $r(t-\omega,t)$ denotes the number of hello received from the contact node during the last ω seconds, and ω/λ denotes the number of hello the contact node sent during the last ω seconds.

The forward link quality p_{fwd} with a contact node is obtained as follows. In addition to the IP header in the hello message, nodes also add the contact IDs and the reverse link qualities with each of the contacts. These reverse link qualities in the hello message are indirectly telling each of the contact node the forward link quality from each of them. From this p_{fwd} and above discussed p_{rev}, nodes obtain the bi-directional link qualities p with each of the contact nodes as shown in **Fig. 1**. The illustration of hello packet header format is given in **Fig. 2**.

From the measured link delivery probabilities, nodes compute the link delivery probabilities to neighbouring clusters. The illustration with three clusters named A, B, and C is shown in **Fig. 4**. Clusters A, B, and C act here as a source cluster, forwarding cluster, and destination cluster, respectively. A node in cluster B computes its link delivery probability to cluster C as follows.

$$p_{b_i,C} = 1 - \prod_{j=1}^{|Z_C|} (1 - p_{b_i,c_j}), \forall b_i \in Z_B \tag{1}$$

where Z_B and Z_C refer to set of nodes in cluster B and cluster C. From $\boldsymbol{p_{b_i,C}}$, the node computes the expected transmission count [19] to cluster C as follows.

$$ETX_{b_i,C} = \frac{1}{p_{b_i,C}} \tag{2}$$

Nodes periodically update the measured ETX-to-cluster values to its head. From these members update, head calculates the cluster-to-cluster ETX (*C2CETX*) as follows.

$$C2CETX_{B,C} = \sum_{e=1}^{2^{|Z_B|-1}} p(e) \min\left\{x_{b_1}(e)ETX_{b_1,C}, x_{b_2}(e)ETX_{b_2,C}, \ldots, x_{b_{|Z_B|}}(e)ETX_{b_{|Z_B|},C},\right\} \tag{3}$$

where $C2CETX_{B,C}$ refers to ETX from cluster B to cluster C. $p(e)$ refers to probability of each event and is computed as follows. Note that χ is the set of all possible events, i.e. $|\chi| = 2^{|Z_B|-1}$.

$$p(e) = \frac{1}{2^{|Z_B|-1}}, e \in \chi \tag{4}$$

The event e here refers to nodes in cluster B receive the data packet from cluster A successfully. $x_{b_i}(e)$ is the binary variable which is defined as "1" if b_i is selected in event e or "0" otherwise. We assume that the probability of each event is evenly distributed and the probability that all nodes fail to receive the data packet from cluster A is zero. Therefore, the calculation in (3) serves as a rough, conservative estimate of the cluster-to-cluster transmission count.

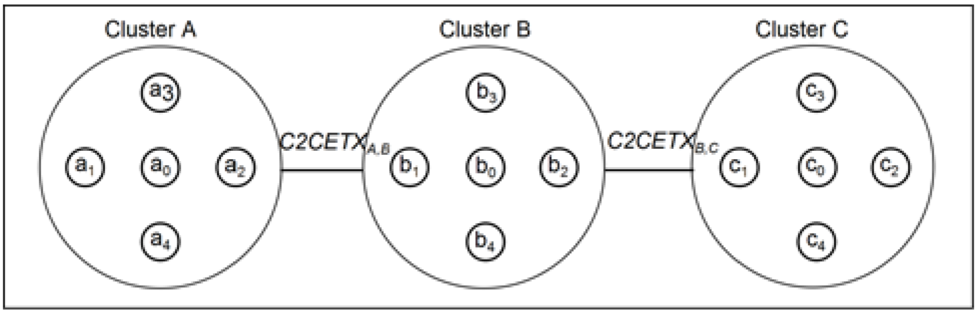


Fig. 4. Network with three clusters and *C2CETX* between clusters.

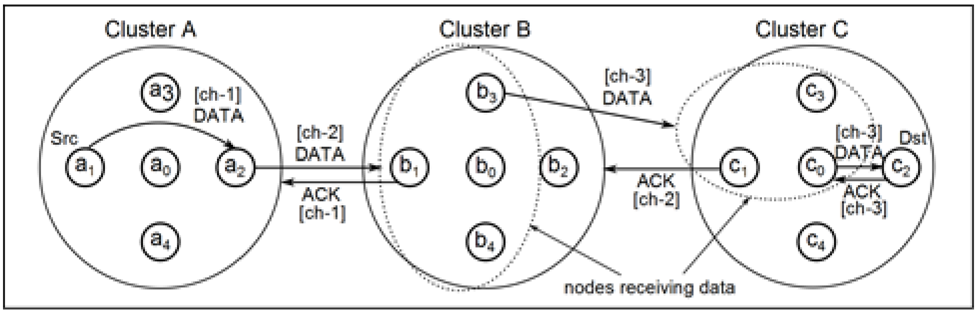


Fig. 5. Illustration of data forwarding in CCDF.
For CCDF-MRMC, please refer to the channel number indicated in the bracket.

5. Cluster-based Cooperative Data Forwarding (CCDF)

In this section, we discuss the details of the CCDF data forwarding algorithm. We first describe an overview of the forwarding procedure. We then elaborate the details of the coordination mechanism which is called distributed agent-based coordination. We also present the additional components of the protocol that complement data forwarding which include network layer acknowledgement, packet buffers (primary and temporary), augmentation for data resiliency, and handling link loss. We also provide an example illustration of CCDF data forwarding in **Fig. 5**. The discussion based on the example is also provided at the end of this section.

Each cluster head maintains a table with the *C*2*CETX* to all other clusters in the network. It then updates its members the consolidated *C*2*CETX* periodically. Cluster *A* calls cluster *B* neighbour cluster if there is at least a node n_i in its cluster with non-zero $p_{n_i,B}$ value.

Cluster head dynamically selects a gateway node to each contact cluster based on the highest link delivery probability to the cluster. In CCDF, nodes which have direct connectivity to a neighbour cluster are called as bridge node to the cluster. So, a gateway node can be considered as one of the bridge nodes which has the strongest link quality to the neighbour cluster. Note that different nodes can be assigned as gateway to different contact clusters or a node can be assigned as gateway to more than one contact clusters. Cluster head will conduct reselection if there is a change in link delivery probability. This information is also added in the hello message. Therefore, member nodes are aware of gateway nodes to each contact cluster.

We focus on the inter-cluster traffic flow in this paper as this type of traffic has to be traversed over multiple hops and has higher chance to be affected by other cross traffic flows. Suppose the source cluster is *A* and the destination cluster is *D*. A source node a_i chooses a cluster path with the lowest *C2CETX* value and performs the data forwarding as follows. Note that a source node chooses a next-hop cluster only if none of the member nodes in the cluster has a non-zero link delivery probability to cluster *D*.

If the source node finds that the source cluster *A* has direct connectivity to the destination cluster, it first checks if it has direct connection to the destination cluster. If it does not have direct connectivity, it will then find a bridge node among its reachable members to the destination cluster. If it finds that it does not have direct connectivity to any bridge node, it will then get a member node as a relay and forward the data packet to the gateway via the relay node. Upon receiving the data packet, the bridge node or the gateway node broadcasts the packet to the destination cluster directly.

If the source node finds that there is no direct connectivity to the cluster *D*, it forwards the data packet as follows. It first checks if it has connectivity to the next-hop cluster. Suppose cluster *B* is the next-hop cluster. If it has connectivity to cluster *B*, it will broadcast the data packet to cluster *B* directly. Otherwise, it will forward the packet to a bridge node or the gateway node via a relay if it does not have connectivity to any bridge node. The bridge node or the gateway node will then broadcast the packet to cluster *B*. Upon receiving the data packet, cluster *B* performs distributed agent-based coordination as follows and further forwards the packet. The illustration of various data forwarding paths from the source/sender cluster to the next-hop or the destination cluster is shown in **Fig. 6**.

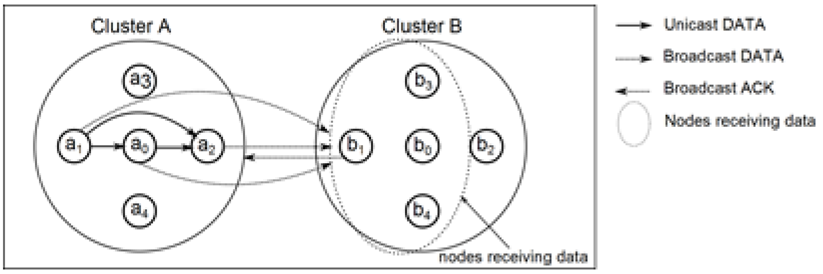


Fig. 6. Illustration of various data forwarding paths from the source/sender cluster to the next-hop or the destination cluster.

5.1 Distributed Agent-based Coordination

We use cluster head as an agent for making decision on selecting a candidate out of all receivers to further forward the data packet. Upon receiving the data packet, nodes update head about the reception information *(senderID, packetID)*. We assume a packet has its unique ID. As we consider multiple traffic flows coming from any directions, the sender ID associated to each flow is attached in the reception information. Each node holds the reception for a period of time to append all the possible incoming traffic and informs head once timeout. Upon receiving the reception information, head first selects the next-hop cluster for each flow. Suppose cluster C is selected as a next-hop cluster for a flow f_i. Head then selects a node which has highest $p_{n_i,C}$ value. If it finds that none of the receivers has connectivity to next-hop cluster, it selects a node which has highest $p_{n_i,n_{gw}}$ value to the gateway node. After making the decision for all the receptions, head broadcasts its decision to the member nodes. For reliability, head sends the decision for three times so that all receivers receive the decision successfully and process the data packet accordingly. This decision re-send also notifies other nodes who has late reception, so that those nodes can promptly discard the packet without further processing.

Upon receiving the decision, nodes which are not selected as forwarding node discard the packet. Therefore, all the receivers not selected as forwarding node need not require any scheduling and processing for the data packet, resulting in higher probability of resource availability for other flows. In addition, the node selected as the forwarding candidate can immediately perform further data forwarding. In the follwing, we discuss the detail of the time taken to perform distributed agent-based coordination.

5.1.1 Coordination time

In the existing cooperative forwarding protocols [14], [16], the forwarding candidates schedule the time-to-forward the data packet based on its forwarding priority/rank assigned by the source. The coordination time at each forwarding node is calculated as

$$t_{b_r} = (r-1)t_\delta + rt_x \tag{6}$$

where t_δ refers to the minimum time interval required for suppression so as to avoid duplicate transmissions, t_x is the delay jitter, and $r = 1$ being the highest priority node (highest rank). The t_δ is calculated as the amount of time to complete a fragment transmission. A fragment may consist of multiple packets. Therefore, one can compute t_δ as

$number\_of\_packets\_per\_fragment \times per\_packet\_forwarding\_time$. In SOAR [16], it limits the maximum number of packets queued on the wireless card to 3 and t_δ is set as 45*ms* (i.e. 15*ms* for each packet).

In CCDF, using distributed agent-based coordination, the time required for coordination is

$$t_{agt} = t_{REP} + t_{REQ} + t_x \tag{7}$$

where t_{agt} is the time required for distributed agent-based coordination, t_{REQ} denotes the time taken for the agent to receive decision request from all potential forwarding nodes, and t_{REP}, the time taken for decision reply. In CCDF, the agent sets the timer from the time it receives the first decision request till the threshold value T_{DEC}. Once timeout, it makes decision for all the request it has received within the time frame. Therefore $t_{REQ} = t_{REQ_1} + T_{DEC}$. Late request for the same packet will be discarded by the agent. A node which has not yet sent out the request but receives the decision from the agent will also drop the outgoing request.

In CCDF, there is a holding period at each candidate node before it reports the reception information to the agent. A holding period, denoted by t_h, is the period of time during which the node tries to consolidate all incoming reception information at the moment so as to have less frequent reporting and more efficient coordination. Hence, if candidate nodes receive another flow packets during t_h, only one request for each candidate is required. We use cumulative request and reply so as to minimize the control overhead. Having this holding period is well in line with the existing cooperative forwarding protocols. In cooperative forwarding, sender sends data in bulk transfer (i.e., a fragment of packets) and candidate nodes start coordination for further forwarding at the end of the fragment transmission. Note that the coordination time in CCDF is not affected by the number of flows, but by the number of packets received at the candidate nodes. When a node finds that the request has been sent out, it will insert the newly received packet information in the next request message. This request will be sent when it reaches the maximum packet count or when holding period timeout. Therefore, if the information is happened to be hold at the candidate nodes, this waiting delay will be added into the coordination time for that packet.

In CCDF, upon receiving the reply from the agent, nodes discard the packet in the buffer if the ID is not listed or schedule their transmissions as follows. Nodes first sort the number of packets to be sent at each candidate node in ascending order. For the same number of packets, node with lowest ID will schedule its transmission first followed by the second lowest ID. Having this agent-based scheduling, it brings the following benefits. 1) Nodes do not require to involve in scheduling if they are not assigned as the forwarding node; 2) the forwarding node becomes the custody of the packet to ensure the delivery instead of source; 3) there is no scheduling misalignment, duplicate transmissions, and collisions due to inaccuracy in scheduling; and 4) it shortens the coordination time especially for multi-flow network with high traffic load.

By having forwarding coordinators distributed over the network, no global priority scheduling is required. As a result, it can prevent the following situations. 1) Nodes being unable to hear the higher priority node transmission successfully due to the traffic coming from another source and therefore wrongly believe that the higher priority node did not receive the packet, resulting in duplicate transmissions; 2) having multiple schedules for multiple flows and inaccuracy in scheduling due to the unavailable medium, resulting in forwarding the same packet by different forwarding nodes at the same time; and 3) unnecessarily waiting for the higher priority node transmission when it is the only node who receives the packet correctly.

5.1.2 Receiving cluster diversity

CCDF also allows to take advantage of transmissions reaching nodes in a cluster closer to the destination cluster, other than the nodes in the next-hop cluster. That is, due to dynamic wireless link nature, it is possible that cluster *C* receives the data packet from cluster *A* over long, weak wireless links. To exploit this spatial diversity, cluster *C* also performs agent-based coordination and further forwards the packet to the destination cluster *D*. Upon overhearing the transmission from cluster *C*, the forwarding node in cluster *B* discards the ongoing transmission. It also inspects the interface queue and drops the packet if it has passed the packet to lower layer for transmission.

5.2 Network Layer Acknowledgement

In CCDF, nodes use IEEE 802.11 unicast for intra-flow data transmission and IEEE 802.11 broadcast for inter-flow data transmissions. We use network layer acknowledgement scheme as IEEE 802.11 broadcast mode does not provide any acknowledgement for the broadcast packet. Prior to forwarding the data packet to another cluster, the source/forwarding/gateway node dynamically chooses a node in the receiving cluster as ACK-sender node. It selects the node with which it has the highest link delivery probability. It then inserts the nodeID in the packet header as shown in **Fig. 3** and forwards the packet. Upon receiving the data packet from another cluster, a node sends an ACK if it is assigned as the ACK reply node in the packet header. Due to the dynamic nature of wireless links, different nodes can be assigned as ACK-sender for different data packets. In case of missing acknowledgement from the next-hop cluster, data sender can still be informed about the successful data reception when it overhears the data packet transmission from the next-hop cluster.

CCDF uses cluster-by-cluster acknowledgement. That is, once the source/forwarding node receives the ACK from the next-hop cluster, it discards the packet from its buffer. The forwarding node in the next-hop cluster becomes the custody for the data packet until it has successfully forwarded the packet to its next-hop cluster or the destination cluster.

We use cumulative acknowledgement instead of acknowledgement for individual packet to reduce the control packet overhead and collisions to data packet transmissions. This acknowledgement is transmitted in broadcast mode. Therefore, it informs not only the sender about successful reception of the data packet but also other nodes who are within the communication range. For ACK message reliability, the gateway node re-broadcasts the ACK it receives from next-hop cluster to make sure that the source/forwarding node are acknowledged.

5.3 Packet Buffer

In the proposed protocol, a node maintains two types of buffer to store data packets as follows.

1) Primary packet buffer (buffer): It is used to store the data packet while waiting for the head decision about the selected forwarding node.

2) Temporary packet buffer (temporary buffer): It is used to keep copies of packets which are being forwarded to the next-hop cluster but waiting for acknowledgement to ensure the successful reception. After the node overhears the acknowledgement from the ACK reply node, the corresponding packet copy will be removed from the temporary buffer. Otherwise, the node will retrieve the packet from the temporary buffer and store it in the primary buffer after acknowledgement timeout.

5.4 Augmentation for Data Resiliency

As data packets are sent over long, weak wireless links, CCDF adds an augmentation mechanism to ensure data resiliency. When a cluster finds that if it has a non-zero link delivery value to the destination cluster, it will forward the data packet directly without traversing other clusters. However, sending over the long weak links likely incurs transmission failure with higher probability. Therefore, to strengthen this data delivery, an augmentation scheme is introduced. Suppose a_2 is the sender, c_1 is the ACK-sender selected by a_2 and clusters A and C are the source/sender cluster and the destination cluster, respectively as shown in **Fig. 7**.

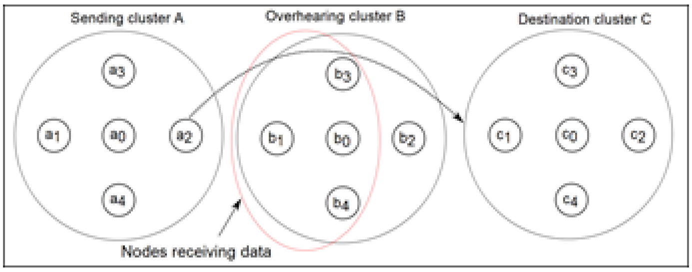


Fig. 7. Illustration of augmentation for data resiliency.

Consider the case where only nodes b_0, b_1, b_3, and b_4 receive the data packet successfully from a_2. Suppose b_0, b_1, b_2, b_3, b_4, and c_1 are listed as neigbours of a_2 and b_0, b_2, b_3, and b_4 are listed as neighbours of c_1 in the most recent hello message of a_2 and c_1, respectively. The hello message header format is given in **Fig. 2**. In this case, it can be observed that only nodes b_0, b_3, and b_4 will participate in the augmentation for data resiliency.

Assume that nodes pariticipating in the augmentation process are denoted as n_k. A node n_k will further check whether 1) its link delivery probability to C, $p_{n_k,C}$, is higher than $p_{a_2,C}$ and 2) its link delivery probability to A, $p_{n_k,A}$, is higher than $p_{c_1,A}$. The link delivery probabilities $p_{a_2,C}$ and $p_{c_1,A}$ can be computed from the neighbour list information included in the hello message of nodes a_2 and c_1, respectively. If the conditions are met, the node will self-evaluate the length of delay as in (8) and will broadcast the packet when the timer expires.

$$Delay(n_k) = \frac{t}{p_{n_k,C}} + x \tag{8}$$

where $p_{n_k,C}$ is the link delivery probability from n_k to cluster C. t refers to the average one-hop delay, i.e, the time taken from the transmitter network layer to the receiver network layer, which includes processing time, queuing time, and transmission time and x is the jitter which follows uniform distribution. When a node n_k overhears transmission from another n_k node or the acknowledgement for the data packet from the ACK-sender, it will discard the ongoing transmission. By having the backup transmissions, the success probability of data delivery from a_2 to the destination cluster C is improved.

5.5 Handling Link Loss

Due to the dynamic and random nature of wireless link, the communication link can be lost while packets are being forwarded. To stop sending out packet if the link is lost, the link layer maintains a bi-directional link delivery information table and passes the packets to interface queue only if the link delivery probability is non-zero. In case of link loss, the link layer will discard the packet.

5.6 Data Forwarding Example

An example illustration about the forwarding in CCDF is given in **Fig. 5**. There are three clusters: *A*, *B* and *C* in this example. Node a_1 from cluster *A* is the source and c_2 from cluster *C* is the destination. Assume none of the nodes in cluster *A* has non-zero delivery probability to cluster *C*. From the *C2CETX* information, a_1 chooses cluster *B* as the next-hop cluster.

Suppose the link delivery probability from a_1 to cluster *B* (i.e., $p_{a_1,B}$) is zero and $p_{a_2,B} > p_{a_0,B} > p_{a_3,B} > p_{a_4,B}$. Cluster head of *A* (assumes a_0 in this example) assigns node a_2 as the gateway node to cluster *B* as it has highest delivery probability to *B* among all other nodes.

As a_1 has no direct connectivity to *B*, it unicasts its data packet to the gateway node a_2. Upon receiving the data packet, a_2 chooses b_1 as the node to reply *ACK* for its upcoming transmission, inserts b_1's ID in the packet header, and broadcasts the packet. Upon receiving the data packet from a_2 (nodes receiving a_2's transmission are shown in the dotted line), nodes in *B* reply ACK if it is assigned as the ACK reply node. Note that the node a_2, which is the current gateway node to *B* will re-broadcast the ACK reply upon receiving so as to ensure that all the nodes in *A* receive the reply. Nodes in *B* will then perform distributed agent-based coordination. Suppose cluster head of *B* (assumes b_0 in this example) selects node b_3 as the forwarding node as it is the best candidate among others. Upon receiving the decision, nodes not selected as the forwarding candidate will then drop the packet. The forwarding node, which is b_3 in this example will then broadcast the data packet to cluster *C*. Here, it chooses c_1 as the ACK reply node for its upcoming transmission.

Upon receiving the data packet from b_3, the ACK reply node, which is c_1 in this example, will first reply the ACK to *B*. As the receiving node knows that the destination, which is c_2 in this example, is a member in its cluster, each receiver (c_0, c_1, and c_3) will self-evaluate the time-to-forward based on its delivery probability to c_3 using (8) and broadcasts once timeout. Upon receiving the data packet, c_3 replies the ACK using broadcast. By overhearing another receiver's transmission or the ACK reply from c_3, the rest of the receivers will cancel their schedule and drop the packet.

6. CCDF with MRMC

Although we can now avoid coordination misalignment and scheduling inaccuracies, there still exists interference and collisions in multiple flow network. As a result, it causes all potential forwarding nodes being unable to receive the data packet from source/sender and hence, resulting in throughput drop and unsuccessful data delivery. In order to suppress potential interferences and collisions and improve the system capacity, a promising way is to incorporate the MRMC capability into CCDF. In the following, we first discuss how radios and channels are being assigned in CCDF and then present the cooperative data forwarding over diverse channels.

6.1 Cluster-based Channel Assignment

We use $K-1$ channels for data transmission/reception and one channel for control purpose. We call $K-1$ channels "data channel" and the one used for sending/receiving control messages "control channel". Each node is equipped with three radios namely $R1$, $R2$, and $R3$. The radio $R1$ is used for control purposes and therefore, it tunes to control channel. The second radio $R2$ is used for receiving incoming packets and the last radio $R3$ is used for transmitting outgoing packets. By allocating the transmit radio and receive radio specifically, it supports multi-radio multi-channel functions for multiple receptions with unlimited number of channels.

Here, channel assignment is required only for radio $R2$. Cluster head chooses a channel for $R2$. Assume each cluster C_i, where $C_i \in C$, has a unique ID, ID_{C_i}; $(0 \le ID_{C_i} \le |C|-1)$. Cluster head of each cluster conducts an initial channel assignment for $R2$ as follows.

$$k_{C_i} = ID_{C_i} (\bmod K-1) \tag{9}$$

where k_{C_i} refers to the channel number of $R2$ at cluster C_i. Member nodes in each cluster set $R2$ to k_{C_i} upon receiving the notification from the head. As we focus on the scheduling and data delivery problem in cooperative forwarding under multi-flow network, we assume that every adjacent cluster in the network uses different channel for $R2$. For the inter-flow interference estimation and channel reassignment, one can use the approach in *iAWARE* protocol [21] [22] and switch the channel for $R2$. As the adjacent cluster selects different channels for $R2$ and the node switches $R3$ as per the next-hop cluster $R2$'s channel for transmission, the probability of having intra-flow interference is low.

6.2 Data Forwarding over Diverse Channels

For intra-cluster data communications, nodes use $R2$ for both transmission and reception. For inter-cluster data communications, nodes use radio $R2$ for reception and $R3$ for transmission. All members in a cluster use the same channel on $R2$. By tuning the same channel on $R2$, the receivers diversity required for cooperative data forwarding is achieved.

Consider clusters A, B, and C are the source, next-hop, and the destination cluster respectively. Let a_i be the source/sender node in cluster A and b_i be the forwarding node selected using distributed agent-based coordination in cluster B. Let $R2$ of clusters A, B, and C are on channels 1, 2, and 3, respectively. The data packet from a_i to cluster C will be forwarded as shown in **Fig. 8**.

As described in **Fig. 8**, when there is no direct connectivity to cluster B, a_i uses $R2$ to reach to gateway node a_{gw}. Then the gateway node uses $R3$ and tunes the channel to the one cluster B is using for receiving. Otherwise, it uses its $R3$ radio and tunes the channel to cluster B's receiving channel. Same concept applies to the second cluster-hop forwarding as well.

Using dedicated radio for transmission and for reception and switching the transmitter radio's channel based on the receivers' channel, concurrent transmissions is achieved while retaining the spatial diversity. In addition, as adjacent clusters are using different receiving channel, the interference caused by other flows coming to the adjacent cluster can be avoided. As discussed in CCDF, having distributed agent-based coordination enables to process multiple incoming flows at the same time. Incorporating MRMC into the CCDF further enhances the system capacity as forwarding candidates selected for each flow will likely be using different channels for their outgoing transmissions. Moreover, CCDF with MRMC works well for unlimited number of channels while requiring only three radios.

```
First cluster-hop forwarding:
if $p_{a_i,B} = 0$ then
    $a_i \xrightarrow{ch2} a_{gw} \xrightarrow{ch3} B$
else
    $a_i \xrightarrow{ch3} B$
end if
Second cluster-hop forwarding:
if $p_{b_i,C} = 0$ then
    $b_i \xrightarrow{ch3} b_{gw} \xrightarrow{ch4} C$
else
    $b_i \xrightarrow{ch4} C$
end if
```

Fig. 8. Data forwarding over diverse channels

6.3 CCDF-MRMC Example

The example illustration of CCDF with MRMC is given in **Fig. 5**. Assume there are four channels *ch*0, *ch*1, *ch*2, and *ch*3 in the network (i.e., K =4). Suppose R_1 of all nodes tune to channel *ch*0. Using (9), nodes in clusters *A*, *B*, and *C* choose *ch*1, *ch*2, and *ch*3 for their *R*2 radio, respectively. Therefore, as shown in **Fig. 5**, when a node sends a data packet to another cluster (i.e., inter-cluster communication), it uses *R*3 radio and tunes the channel of *R*3 to the receiving cluster's *R*2's channel. Note that for the intra-cluster communication, it uses *R*2 radio. It is also important to point that CCDF-MRMC allows channel reuse. That is, if there are only 3 channels in the network (i.e., K =3), cluster *C* will reuse *ch*1 and therefore, its *R*2 radio will be operating on *ch*1. We will show the performance of both cases, i.e., with reuse and without reuse, in the performance evaluation.

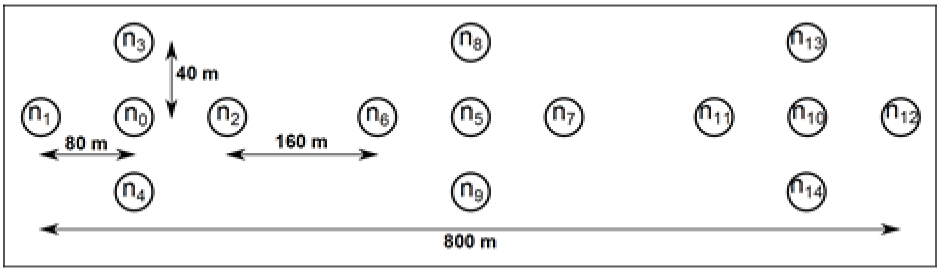


Fig. 9. Network topology

Table 1. Simulation parameters

Parameter	**Scenario I**	Scenario II
Network area (lxl m^2)	80x800	80x800
Number of nodes	15	15
Transmission range	Probabilistic	Probabilistic
Fading model	Nakagami	Nagamai
Traffic rate	10 Packets/sec	100 Packets/sec
Total number of packets	3000	100
Packet size	512 Bytes	1000 Bytes
Traffic type	Constant Bit Rate (CBR)	Constant Bit Rate (CBR)
Number of traffic flows	2, 3, 4, 5, 6	1
Traffic flow (src,dst)	(n_1, n_{12}), (n_{13}, n_3), (n_4, n_{14}), (n_{11}, n_0), (n_2, n_{10}), and (n_1, n_{14})	(n_1, n_7), (n_1, n_{11}), (n_1, n_{10}), and (n_1, n_{12})

7. Performance Evaluation

We use network simulator (NS-2) to evaluate the performance of the proposed scheme. The evaluation covers two main scenarios which include 1) single-radio single-channel (SRSC or 1R1C); and 2) multi-radio multi-channel (MRMC) with three radios three channels (3R3C) and three radios four channels (3R4C) cases. We implement AODV [1] with MRMC capability for performance comparison in MRMC networks. We examine 1) the packet delivery ratio (PDR) which shows the amount of packets delivered over the amount of packets generated, 2) the end-to-end delay which indicates the time taken from the packet generated at the source application layer to the packet delivered at the destination application layer, and 3) the MAC layer transmission count which indicates the number of times a packet is transmitted/forwarded by the nodes over the air to be delivered to the destination. The transmission count here also takes into account the number of retransmissions as well.

The network topology used in our simulation is illustrated in **Fig. 9**. The Nakagami propagation model is used to study the effect of realistic fading environment. Therefore, the transmission range between two nodes is probabilistic. The fading model parameters can be found in [18]. **Fig. 10** shows the probe (HELLO) packet reception ratio over distance. With this reception probability, it is noted that the source and destination pair (n_1, n_{12}) which are at the two ends would take 10 hops end-to-end path length under 100% reception probability. In CCDF, with the given topology, three clusters are formed with nodes n_0, n_1, n_2, n_3, and n_4 in cluster *A*, n_5, n_6, n_7, n_8, and n_9 in cluster B and n_{10}, n_{11}, n_{12}, n_{13}, and n_{14} in cluster *C*, respectively. Note that, the node with the highest packet reception ratio to all the other nodes in the cluster is assigned as the head (i.e. agent for coordination). Therefore, nodes n_0, n_5, and n_{10} become the head of the respective cluster.

We compare the network performance with the end-to-end routing (AODV) for SRSC network and AODV-MRMC for MRMC network. We use scenaio I network model shown in **Table 1**. We study the effect of number of traffic flows on network performance. In addition, we also conduct a performance comparison with an opportunistic routing (OR) scheme. The implementation of OR scheme is discussed in Section 7.2. This study is conducted on single flow traffic in SRSC environment. For this study, we use scenario II network model. To the best of our knowledge, no cooperative forwarding study/mechanism on the same candidate relay set being used for multiple cross traffic flows and under multi-radio multi-channel environment has been conducted/proposed. In this study, we show the performance advantage of using proposed agent-based coordination compared to the commonly used timer-based coordination.

The IEEE 802.11 standard is used in all the simulation studies in this paper. All traffic flows are injected into the network at the same time to study the multi-flow network performance. We collect simulation results using identical traffic model with at least 10 different traffic injection time. Each simulation is run for 900 seconds. The detail simulation parameters are described in **Table 1**.

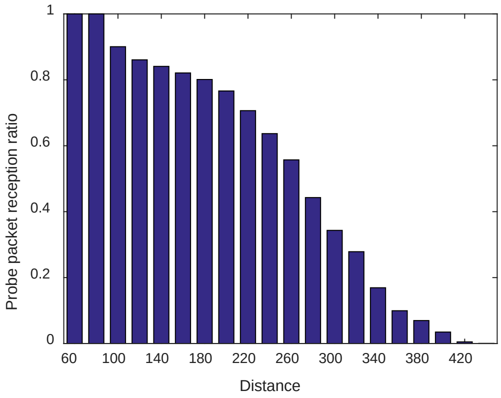


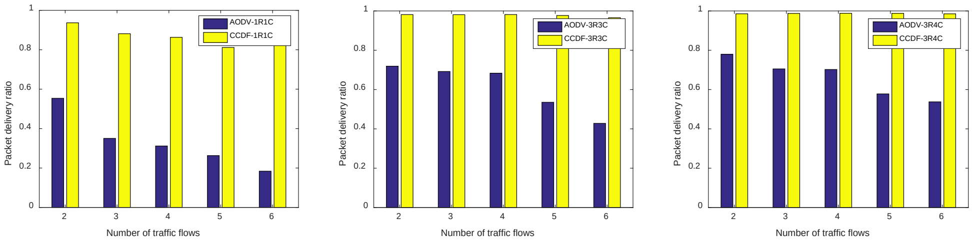
Fig. 10. Probe packet reception ratio over distance.

7.1 AODV with MRMC Implementation

We implement the same radio-channel assignment for the AODV in order to make the performance comparison. In AODV-MRMC, each node is equipped with three radios $R1$, $R2$, and $R3$. The radio $R1$ is used for control purposes and therefore, it tunes to control channel. The second radio $R2$ is used for receiving incoming packets and the last radio $R3$ is used for transmitting outgoing packets. All the control messages are sent over the control channel. Route discovery which includes sending/receiving route request (RREQ) and route reply (RREP) messages are also conducted over the control channel. The node uses the transmitting radio $R3$ to forward the packet. The channel for $R3$ is switched based-on the next-hop receiver's channel, same as in the CCDF-MRMC. The node can receive the data packet at the same time while transmitting as it uses $R2$ for incoming data packet, as per the CCDF-MRMC.

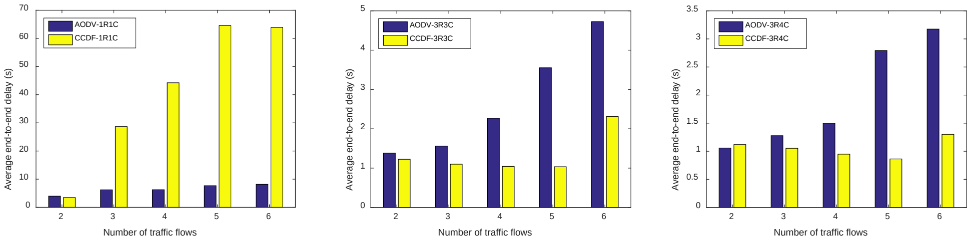
7.2 Opportunsitic Routing Implementation

We impletement opportunistic routing (OR) protocol in NS-2 based on the ExOR mechanism [14]. Candidate relays are selected and prioritized based on the ETX to the destination. Nodes maintain in-memory batch map which records the highest priority candidate ID which is known to have a copy of the packet. Nodes embed the in-memory batch map in the data packet header and broadcast the packet. This embedded information indirectly acknowledge other candidates of which packets were successfully received. Timer-based cooridnation is used for transmission scheduling and duplicate trannsmssion avoidance. As shown in **Table 1** scenario II network model, the traffic source initially generates 100 data packets. Therefore, we set batch size for OR to 100.



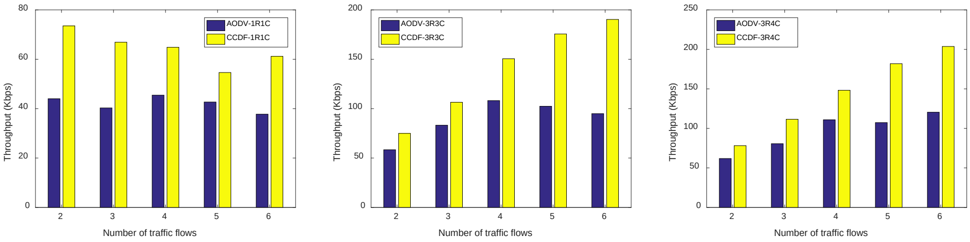
(a) (b) (c)

Fig. 11. Packet delivery ratio: (a) SRSC; (b) 3R3C; and (c) 3R4C.



(a) (b) (c)

Fig. 12. Average end-to-end delay: (a) SRSC; (b) 3R3C; and (c) 3R4C.



(a) (b) (c)

Fig. 13. Throughput: (a) SRSC; (b) 3R3C; and (c) 3R4C.

Table 2. Relative packet delivery ratio improvement of CCDF-MRMC over AODV-MRMC counterpart

	2 flows	3 flows	4 flows	5 flows	6 flows
CCDF-3R3C	136.49	141.77	143.68	182.46	225.59
CCDF-3R4C	126.43	140.03	140.72	170.89	183.27

Table 3. Relative end-to-end delay improvement of CCDF-MRMC over CCDF-SRSC

	2 flows	3 flows	4 flows	5 flows	6 flows
CCDF-3R3C	35.34	3.84	2.36	1.60	3.62
CCDF-3R4C	32.27	3.67	2.14	1.34	2.04

Table 4. Relative end-to-end delay improvement of CCDF-MRMC over AODV-MRMC counterpart

	2 flows	3 flows	4 flows	5 flows	6 flows
CCDF-3R3C	88.61	70.47	45.99	29.14	48.88
CCDF-3R4C	105.76	82.41	63.22	30.91	40.99

Table 5. Relative throughput improvement of CCDF-MRMC over CCDF-SRSC

	2 flows	3 flows	4 flows	5 flows	6 flows
CCDF-3R3C	101.92	159.08	232.08	321.41	311.06
CCDF-3R4C	106.32	166.73	228.59	332.95	332.99

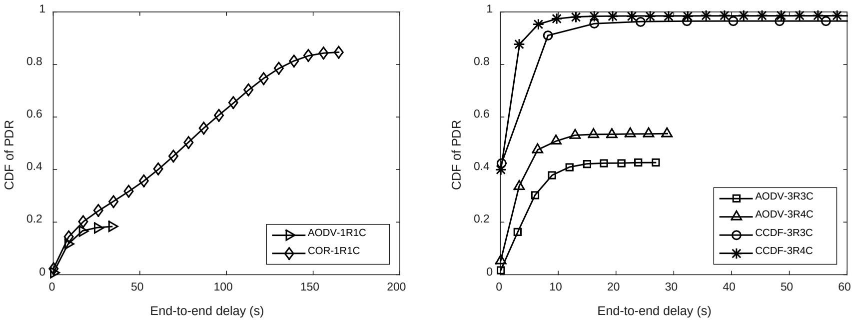
Table 6. Relative throughput improvement of CCDF-MRMC over AODV-MRMC counterpart

	2 flows	3 flows	4 flows	5 flows	6 flows
CCDF-3R3C	128.35	127.90	139.14	171.45	200.46
CCDF-3R4C	126.31	138.32	133.71	169.41	169.04

7.3 Results and Discussion

7.3.1. Simulation Study in Multi-Radio Multi-Channel Multi-Flow Network

For SRSC, the packet delivery ratio, end-to-end delay, throughput, and CDF of PDR over end-to-end delay are shown in **Fig. 11a**, **Fig. 12a**, **Fig. 13a** and **Fig. 14a**. It is observed that CCDF maintains the PDR at 90% regardless of the number of flows whereas in AODV, the PDR is decreasing with increasing number of flows. AODV usually maintains the PDR at almost 100% delivery under perfect network condition (i.e., network links with no loss rates.). However, in lossy environments, the PDR drops to about 50% under light traffic load (i.e. 2 flows) and the drop rate gets higher when more cross traffic flows are injected into the network. It is well known that cooperative forwarding works well under the lossy nature of wireless environments. Therefore, we can see that over 90% PDR is achieved with minimal delay in two-flow networks. However, with more cross traffic flows, it is observed that the throughput of cooperative forwarding is reduced. By having the agent-based distributed coordination in CCDF, the forwarding node selected by the agent works as a custody to ensure the delivery for the packet. As a result, the delivery is maintained at 90% regardless of the number of cross traffic flows. However, it is noted that due to timeout and retransmission for unsuccessful delivery, the delivery delay is increased and the throughput is reduced.



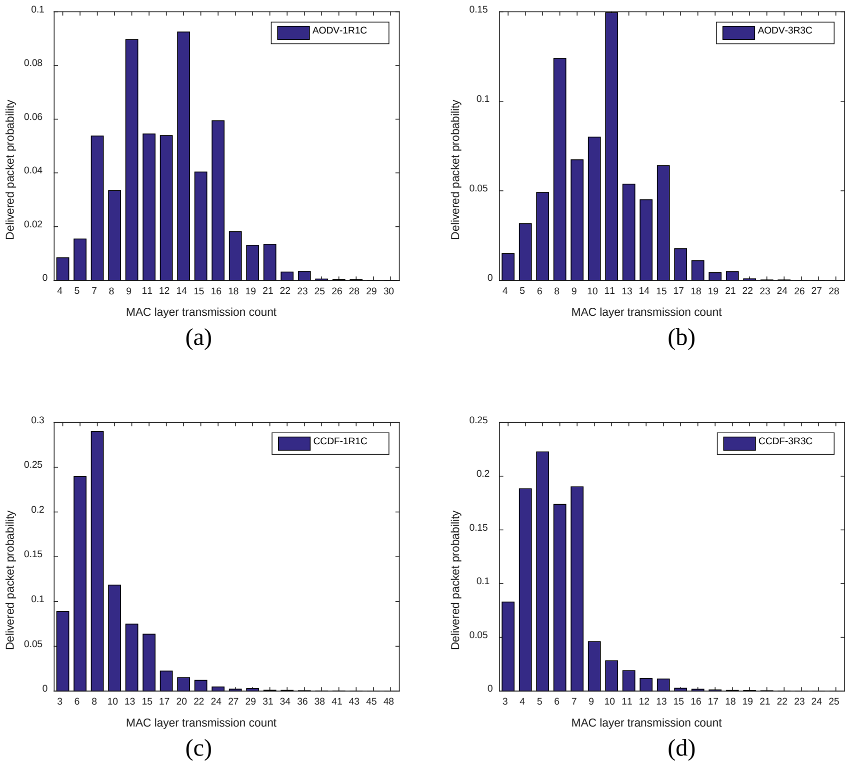
(a) (b)

Fig. 14. CDF of PDR: (a) SRSC and (b) 3R3C and 3R4C

The packet delivery ratio, end-to-end delay, and throughput for 3R3C are shown in **Fig. 11b**, **Fig. 12b**, and **Fig. 13b** and for 3R4C are shown in **Fig. 11c**, **Fig. 12c**, and **Fig. 13c**. The CDF of PDR over end-to-end delay for both MRMC cases is shown in **Fig. 14b**. The relaitve PDR improvement over AODV-MRMC is shown in **Table 2.** The relative delay and throughput improvement over CCDF-SRSC and AODV-MRMC are shown in **Table 3**, **Table**

4, **Table 5**, and **Table 6**, respectively. From the performance evaluation, we observe that the proposed solution can achieve both multiplexing and spatial diversity. Specifically, CCDF with multi-radio mutli-channel capability reduces about 97% of the delay and achieves about 200% of throughput improvement compared to the single-radio single-channel (SRSC) case. This improvement implies that CCDF-MRMC ensures the achievable throughput of cooperative forwarding regardless of the number of traffic flows. It also shows that CCDF-MRMC works well even under the situation where the same set of candidate nodes is being used for multiple flows.

We then study the MAC layer transmission count under light traffic (2 flows) and congested traffic (6 flows) scenarios. The delivered packet probability distribution based on the transmission count incurred under light traffic scenario is shown in **Fig. 15**, and under congested traffic scenario is shown in **Fig. 16**. The average transmission count per delivered packet is shown in **Table 7**. We observe that CCDF makes more transmissions in SRSC compared to MRMC cases. With the given topology, it can be seen that nodes n_5, n_6, n_7, n_8, and n_9 will be assigned as the candidate nodes. Therefore, with more cross flow traffic injected into the network, more collisions, drops, and retransmissions are being made. As a result, 6-flow network makes about 13 transmissions per delivered packet whereas 2-flow network incurs about 9 transmissions. This is in contrast to the AODV case. In AODV, 2-flow network makes about 12 transmissions whereas in 6-flow, it reduces to 11 transmissions. This is because of the higher drop rate and unsuccessful delivery in AODV under more contending traffic scenario.

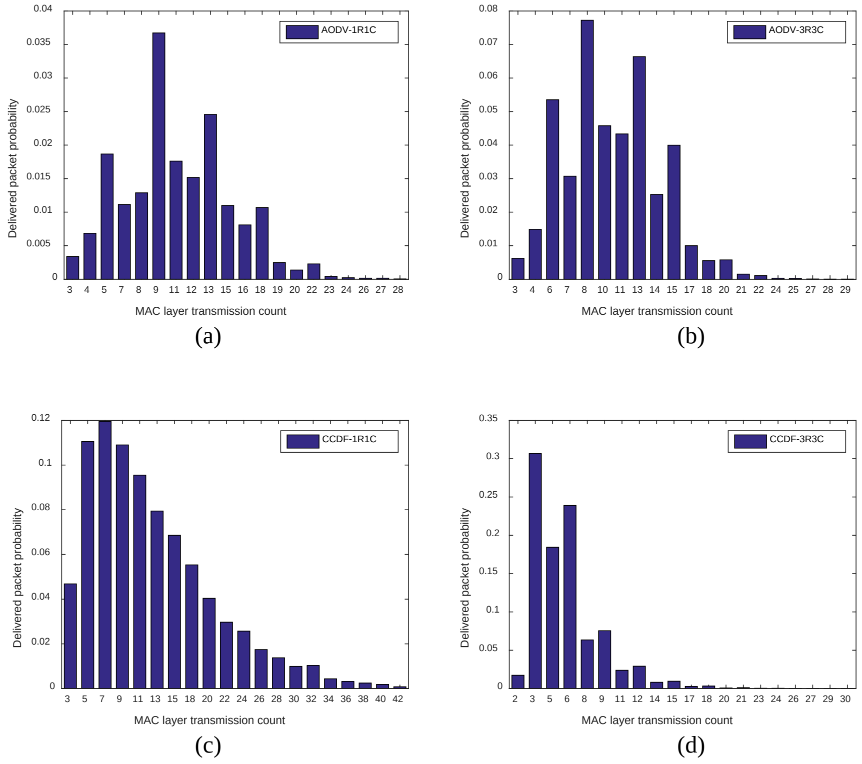


(a) (b)

(c) (d)

Fig. 15. Study on MAC layer transmission count in two-flow network:
(a) AODV-SRSC; (b) AODV-3R3C; (c) CCDF-SRSC; and (d) CCDF-3R3C.

We can see that CCDF-MRMC significantly improves the transmissions required per delivered packet over CCDF-SRSC in both 2-flow and 6-flow networks as shown in **Table 7**. Specifically, in 2-flow network with CCDF-MRMC (i.e., both CCDF-3R3C and CCDF-3R4C), the tranmsission count reduces to 67% and 65% of the one incurred by CCDF-SRSC. We observe more significant improvement in 6-flow network, where it reduces to 47% and 42%. In CCDF, using agent-based coordination, candidate nodes need not require to have multiple schedules for multiple flows. It in fact prevents several potential duplications caused by 1) missing higher priority node transmissions and 2) scheduling inaccuracy due to unavailable medium. In addition, having receiver diversity significantly improves the chance of successful delivery, and it proves that CCDF-MRMC further enhances this capability. It is observed that AODV incurs a large number of tranmsmissions to deliver a packet to the destination in both SRSC and MRMC cases. Specifically, under light traffic load, it incurs about 12, 11, and 10 average transmissions for SRSC, 3R3C, and 3R4C, respectively. Similarly, under congested traffic, it incurs about 11, 10, and 10 average transmissions, respectively. In AODV, although MRMC provides more bandwidth resource, the lossy wireless link environments strongly impact the AODV performance. Although the successful delivery probability is improved in MRMC, it shows that it still incurs frequent losses and retransmissions during the packet delivery.



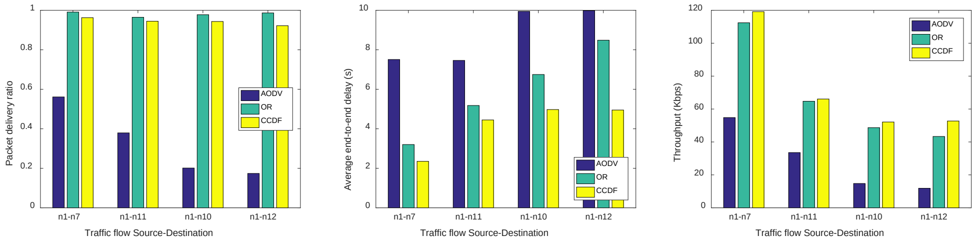
(a) (b)

(c) (d)

Fig. 16. Study on MAC layer transmission count in six-flow network:
(a) AODV-SRSC; (b) AODV-3R3C; (c) CCDF-SRSC; and (d) CCDF-3R3C.

Table 7. Average MAC layer transmission count per delivered packet

Flow count	AODV-1R1C	AODV-3R3C	AODV-3R4C	CCDF-1R1C	CCDF-3R3C	CCDF-3R4C
2 flows	12.077	10.567	10.133	8.905	5.988	5.832
6 flows	10.906	10.369	9.721	12.977	6.104	5.566



(a) (b) (c)

Fig. 17. Comparisons with AODV and OR in scenario II network model:
(a) Packet delivery radio; (b) Average end-to-end delay; and (c) Throughput.

7.3.2. Comparison with Opportunistic Routing

In this study, source node n_1 sends a single batch of 100 packets to the destinaiton node. We study the performance by varying different hop counts between the source node and the destination node. Specifically, we choose node n_7, n_{11}, n_{10}, and n_{12} as the destination node. Packet delivery raito, average end-to-end delay, and throuhgput performance are shown in **Fig. 17a**, **Fig. 17b**, and **Fig. 17c**. It can be observed that CCDF achieves packet delivery ratio almost the same as OR but with much lower end-to-end delay. By using agent-based coordination, nodes need not require to wait for the higher priority candidates' forwardings. Nodes are able to schedule the forwarding immediately upon receiving the decision from the agent and forward the data packet whenever the medium is available. It can also be seen that for cases with destination nodes n_{11}, n_{10}, and n_{12}, which are generally located in the same cluster in CCDF, similar delivery delay is observed regardless of the hop count from the source. However, in OR, we observe incresing end-to-end delay which is due to additional coordination time incurred when more forwarding candidates are involved. For the end-to-end routing AODV, it is noted that the delivery performance and the throughput drop significantly with the increasing number of hops between the source node and the destination node. We observe that this performance drop in end-to-end routing is caused by probabilistic wireless channels, non-cooperative data forwarding, and high traffic bursts.

7.3.3. Overhead

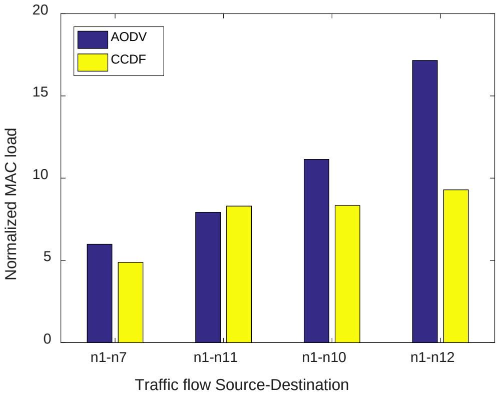


Fig. 18. Normalized MAC loads of AODV and CCDF in scenario II network model.

We then study the normalized control message load of AODV and CCDF with varying hop counts between the source node and the destination node. This performance shows the number of routing, Address Resolution Protocol (ARP), and IEEE 802.11 unicast acknowledgement (ACK) packets transmitted at the MAC layer for each delivered data packet. In CCDF, the routing control packets include decision request and decision reply required for distributed agent-based coordination and cumulative acknowledgement for IEEE 802.11 broadcast packets. Note that CCDF with MRMC does not use any additional control packets. In AODV, the routing control packets include route request, route reply, and route error messages. This control load performance does not include the hello message overhead as the message is sent periodically throughout the simulation. Note that hello message is used in all protocols (AODV, OR, and CCDF) for neighbour discovery and link metric computation. For OR, it does not incur any additional control overhead as the batch map is embedded in the data packet header.

As shown in **Fig. 18**, it can be observed that the overhead of CCDF in (n_1, n_7) case is very low. In this case, the source and destination clusters are the neighbours. Thus, it does not need to traverse another cluster, resulting in no agent-based coordination process involved. Therefore, the routing overhead due to agent-based coordination process are not incurred. In (n_1, n_{11}), (n_1, n_{10}), and (n_1, n_{12}) cases, although there is an increase in the overhead, the increment is minimal and does not vary with the increasing hop count. In CCDF agent-based coordination, the decision request and decision reply control messages are used within the cluster only. However, in AODV, the routing control messages are sent in the network-wide manner, resulting a significant increase in overhead with increasing hop count between the source node and the destination node.

8. Conclusion

We have proposed a cluster-based cooperative data forwarding for single-radio single-channel and multi-radio multi-channel (MRMC) for multi-hop wireless network. The cluster-to-cluster expected transmission count (*C2CETX)* metric is introduced to compute cluster path for forwarding. Data packet is broadcasted over the cluster path. Distributed agent-based coordination is proposed to select the most suitable candidate dynamically. With the agent-based coordination, it is shown that cooperative forwarding need not require to have global candidate relay set selection and prioritization. To further enhance the system throughput, the CCDF-MRMC mechanism which include radio-channel assignment and cooperaitve data forwarding over MRMC is proposed. The network simulation shows that cooperative forwarding using agent-based coordination achieves significant performance improvement over the cooperative forwarding using timer-based coordination. It also shows that the CCDF with MRMC not only retains the advantage of receiver diversity in cooperative forwarding but also minimizes the interference, which therefore maximizes the system throughput for the network with multiple cross traffic flows.

References

[1] C. Perkins, E. Belding-Royer, and S. Das, “Ad hoc on-demand distance vector (AODV) routing," No. RFC 3561. 2003. Article (CrossRef Link)

[2] M. Kodialam and T. Nandagopal, “Characterizing the capacity region in multi-radio multi-channel wireless mesh networks," in *Proc. of the 11th annual international conference on Mobile computing and networking*, pp. 73-87, ACM, 2005. Article (CrossRef Link)

[3] A. P. Subramanian, M. M. Buddhikot and o. Miller, “Interference aware routing in multi-radio wireless mesh networks," in *Proc. of* the *2nd IEEE Workshop on Wireless Mesh Networks*, Reston, VA, pp. 55-63, 2006. Article (CrossRef Link)

[4] M. Alicherry, R. Bhatia, and L. E. Li, “Joint channel assignment and routing for throughput optimization in multi-radio wireless mesh networks," in *Proc. of the 11th annual international conference on Mobile computing and networking*, pp. 58-72, ACM, 2005. Article (CrossRef Link)

[5] A. Raniwala, K. Gopalan, and T.-c. Chiueh, “Centralized channel assignment and routing algorithms for multi-channel wireless mesh networks," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 8, no. 2, pp. 50-65, 2004. Article (CrossRef Link)

[6] R. Draves, J. Padhye, and B. Zill, “Routing in multi-radio, multi-hop wireless mesh networks," in *Proc. of the 10th annual international conference on Mobile computing and networking*, pp. 114-128, ACM, 2004. Article (CrossRef Link)

[7] K. N. Ramachandran, E. M. Belding-Royer, K. C. Almeroth, and M. M. Buddhikot, “Interference-aware channel assignment in multi-radio wireless mesh networks.," in *Proc. of INFOCOM*, vol. 6, pp. 1-12, 2006. Article (CrossRef Link)

[8] Wen Jingrong, Wu Muqing, Lv Bo and Wang Dongyang, “Opportunistic cooperative routing in multi-radio multi-channel wireless sensor networks," in *Proc. of the IEEE International Conference on Communications Workshops (ICC)*, Budapest, pp. 276-280, 2013. Article (CrossRef Link)

[9] F. Wu and N. Vaidya, “Workload-aware opportunistic routing in multi-channel, multi-radio wireless mesh networks," in *Proc. of the 9th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON)*, Seoul, pp. 344-352, 2012. Article (CrossRef Link)

[10] A. Zubow, M. Kurth and J. P. Redlich, “An Opportunistic Cross-Layer Protocol for Multi-Channel Wireless Networks," in *Proc. of the IEEE 18th International Symposium on Personal, Indoor and Mobile Radio Communications*, Athens, pp. 1-5, 2007. Article (CrossRef Link)

[11] K. Zeng, Z. Yang and W. Lou, “Opportunistic Routing in Multi-Radio Multi-Channel Multi-Hop Wireless Networks," in *IEEE Transactions on Wireless Communications*, vol. 9, no. 11, pp. 3512-3521, November 2010. Article (CrossRef Link)

[12] H. Shiming, Z. Dafang, X. Kun, Q. Hong and Z. Ji, “A Simple Channel Assignment for Opportunistic Routing in Multi-radio Multi-channel Wireless Mesh Networks," in *Proc. of the Seventh International Conference on Mobile Ad-hoc and Sensor Networks (MSN)*, Beijing, pp. 201-208, 2011. Article (CrossRef Link)

[13] D. Aguayo, J. Bicket, S. Biswas, G. Judd, and R. Morris, “Link-level measurements from an 802.11 b mesh network," in *Proc. of ACM SIGCOMM Computer Communication Review*, vol. 34, no. 4, pp. 121-132, 2004. Article (CrossRef Link)

[14] S. Biswas and R. Morris, “Exor: opportunistic multi-hop routing for wireless networks," in *Proc. of ACM SIGCOMM Computer Communication Review*, vol. 35, pp. 133-144, ACM, 2005. Article (CrossRef Link)

[15] S. Chachulski, M. Jennings, S. Katti, and D. Katabi, “Trading structure for randomness in wireless opportunistic routing," vol. 37. ACM, 2007. Article (CrossRef Link)

[16] E. Rozner, J. Seshadri, Y. Mehta and L. Qiu, “SOAR: Simple Opportunistic Adaptive Routing Protocol for Wireless Mesh Networks," in *Proc. of IEEE Transactions on Mobile Computing*, vol. 8, no. 12, pp. 1622-1635, Dec. 2009. Article (CrossRef Link)

[17] N. Chakchouk, “A Survey on Opportunistic Routing in Wireless Communication Networks," in *Proc. of IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2214-2241, Fourthquarter 2015. Article (CrossRef Link)

[18] Q. Chen, F. Schmidt-Eisenlohr, D. Jiang, M. Torrent-Moreno, L. Delgrossi, and H. Hartenstein, “Overhaul of IEEE 802.11 modeling and simulation in ns-2," in *Proc. of the 10th ACM Symposium on Modeling, analysis, and simulation of wireless and mobile systems*, pp. 159-168, 2007. Article (CrossRef Link)

[19] D. S. De Couto, D. Aguayo, J. Bicket, and R. Morris, “A high-throughput path metric for multi-hop wireless routing," in *Proc. of Wireless Networks*, vol. 11, no. 4, pp. 419-434, 2005. Article (CrossRef Link)

[20] Z. Wang, Y. Chen, and C. Li, “CORMAN: A novel cooperative opportunistic routing scheme in mobile ad hoc networks,” in *Proc. of IEEE Journal on Selected Areas in Communications*, vol. 30, pp. 289-296, 2012. Article (CrossRef Link)

[21] A. P. Subramanian, M. M. Buddhikot, and S. Miller, “Interference aware routing in multi-radio wireless mesh networks,” in *Proc. of the 2nd IEEE Workshop on Wireless Mesh Networks*, pp. 55-63, IEEE, 2006. Article (CrossRef Link)

[22] Buddhikot, Milind M., Scott C. Miller, and Anand Prabhu Subramanian. "Interference aware routing in multi-radio wireless mesh networks," U.S. Patent No. 8,532,023. 10 Sep. 2013.

Cherry Ye Aung received B.Eng. from Yangon Technological University, Myanmar, in 2009 and M.Sc. from Nanyang Technological University, Singapore, in 2010, both in Electrical Engineering and electronic engineering. She then joined ASM Pacific Technology, Singapore, as a Software Engineer from August 2010 to November 2011. In 2011, she joined School of EEE, Nanyang Technological University, Singapore, as a Ph.D. student. Her research interests are protocol design and development in cooperative opportunistic networks and V2X communication system development in vehicular networks.

G. G. Md. Nawaz Ali is currently a Postdoctoral research fellow with the School of Electrical and Electronic Engineering, Nanyang Technological University (NTU), Singapore. He received his PhD in the Department of Computer Science, City University of Hong Kong in 2013. He received his B.Sc. degree in Computer Science and Engineering from Khulna University of Engineering & Technology, Bangladesh in 2006. He is a member of IEEE and IEEE VTS. His current research interests include wireless broadcasting, mobile computing, network coding, and ad hoc networking with a focus on vehicular ad hoc networking.

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 2000 and 2001, he worked at Agilent Technologies Canada Inc., Canada. From 2001 to 2002, he was at Nokia Research Center, 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 and a Director of Infinitus, Centre for Infocomm Technology, between 2013 and 2016. From April 2016, he joined the Department of Electrical and Electronic Engineering, Auckland University of Technology, New Zealand as a Full Professor and Head of Department. 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 lead Guest Editor of *IEEE Communications Magazine* in 2007 and *IEEE Wireless Communications* in 2011. He is an Editorial Board Member of Security and Communication Networks and an Editor of 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 vehicular communications networks.