Rate-Adaptive Coding-Aware Multiple Path Routing for Wireless Mesh Networks

Yan Yan 1,2  Zhuang Zhao 2  Baoxian Zhang 2,3  Hussein T. Mouftah 4  Jian Ma 5
1 Institution of Computing Technology, Chinese Academy of Sciences, Beijing 100080, China
2 CCCE, Graduate University of Chinese Academy of Sciences, Beijing 100049, China
3 Key Lab of WSN, Shanghai Institute of Microsystem and Information Technology of Chinese Academy of Sciences, China
4 SITE, University of Ottawa, Ottawa, ON KIN 6N5, Canada
5 Nokia Research Center, Beijing 100176, China
Email: yanyan@ict.ac.cn, {zhaozh, bxzhang}@gucas.ac.cn, mouftah@site.uottawa.ca, jian.j.ma@nokia.com

Abstract—Network coding has been considered as an effective strategy for improving the performance of Wireless Mesh Networks (WMNs) by encoding multiple packets into a single transmission. Existing work shows that integration of network coding and routing at the network layer can achieve good performance in terms of network throughput and packet delay. In this paper, we propose a rate-adaptive coding-aware multiple path routing mechanism for WMNs. The main design objective is to improve the network performance via traffic splitting for maximizing the coding opportunities in the network. Simulation results are used to verify the effectiveness of our proposed mechanism.

Index Terms—Network coding, multiple path routing, on-demand routing, wireless mesh networks.

I. INTRODUCTION

Network coding has recently been proposed as an effective strategy to obtain high throughput in Wireless Mesh Networks (WMNs). Network coding obtains coding gain by mixing multiple packets in a single transmission and it can benefit from the inherent broadcast nature of wireless medium in WMNs. Fig. 1 illustrates the basic idea of network coding in wireless network. In Fig. 1, node A and node B want to exchange packets via a common relay node R. A sends packet p1 to B and B sends packet p2 to A. After R receives both packets p1 and p2, it can create a new packet “p1 xor p2,” and then broadcast it to the air. Upon the receipt of the xor-ed packet, both A and B can decode their interested packet. The number of total transmissions in this example is reduced from four to three.

Network coding, however, is possible only when different flows form certain coding structures. Traditional shortest path routing selects paths without considering coding opportunities. Thus, it creates coding opportunities in an opportunistic way. Although some work [2]-[7] has been carried out to address the issue of provide more coding opportunities in routing, how to effectively combine network coding and routing is still not well explored. In particular, most existing network coding mechanisms are based on the assumption that all flows coded together have the same data rate, which is untrue for most applications. If the flows coded together have different data rates, the benefit of network coding cannot be fully realized.

In this paper, we focus on the design of coding-aware rate-adaptive on-demand multiple path routing mechanism. We argue that further gain is possible by making routing decisions with the awareness of coding opportunities and appropriately splitting traffic of a flow to maximize the total coding opportunities in the network. The contributions of this paper are as follows. First, we propose a coding-aware routing mechanism, which makes routing decisions based on node-centric metric instead of link-centric metric as used in most existing work, in order to discover potential coding opportunities as much as possible. Second, the mechanism works in a rate adaptative manner and it splits one flow onto multiple paths, which can utilize coding opportunities effectively and balance network load. Third, routing decision at each node is made in a localized manner. Simulation results demonstrate that our mechanism can improve the performance of WMNs wherein multiple concurrent flows exist as compared with existing work.

The rest of this paper is organized as follows. In Section II, we briefly review related work. In Section III, we present the detailed design description of the proposed mechanism. In Section IV, we conduct detailed simulations to evaluate the performance of our mechanism by comparing it with existing work. In Section V, we conclude this paper.

II. RELATED WORK

Several routing protocols [8]-[13] for mobile ad hoc networks have been proposed. Multiple paths routing [11]-[13] establishes multiple paths between a source-destination pair to achieve load balancing and higher throughput as compared with single path routing.

Network coding was first proposed by Ahlswede et al. [14] and it was initially targeted at tackling the multicast issue. Much existing work [15]-[18] has been focused on the construction of efficient network coding codes. Some recent work [19]-[21] has studied the design of efficient network coding codes.
coding mechanisms for wireless multi-hop networks. Recently, some work [22]-[25] has been focused on providing efficient unicast communications by using network coding. However, most of the coding mechanism utilizes coding opportunities passively, which limits its ability in achieving high performance.

In [2], the authors proposed a network coding aware routing mechanism to reduce the expected number of coded transmission. In [3], the authors proposed a coding aware routing mechanism, which considers multiple paths routing. However, both of these two mechanisms adopt centralized linear programming, which is known to have the scalability issue. Wu et al. [4] proposed a markovian metric to be embedded into existing routing protocols to promote the network coding. In [5], Jerry Le et al. proposed a distributed coding aware routing protocol to select the path with the maximum end-to-end throughput. In our previous work [6], we proposed a coding aware opportunistic routing protocol to increase the network throughput. Although these existing mechanisms perform coding aware routing in various ways, none of them consider traffic splitting. The MAC-independent Opportunistic Routing & Encoding (MORE) protocol [7] adopts a different strategy for supporting coding-aware routing such that intra-flow random network coding is embedded into opportunistic routing. However, MORE has not considered the issue of how to increase coding gains when multiple flows are available.

III. PROTOCOL DESCRIPTION

In this section, we present the detailed design of our Rate-adaptive Coding-aware multiple paths Routing (RCR) mechanism. RCR is aimed at increasing the throughput of WMNs wherein multiple concurrent flows exist. The procedure regarding how to encode multiple packets into a single transmission in RCR is the same as that in the opportunistic network coding scheme COPE [1]. In this paper, we assume there is no limitation on power or processing ability at nodes and nodes can overhear all transmissions in their neighborhood provided that no collisions occur.

A. Observation

Before presenting the detailed implementation, let us give some intuitive observations that motivate the design of our mechanism.

1) Coding Aware Routing

Consider the example network in Fig. 2. Suppose there are two flows: One from node $A$ to node $E$ and the other from node $G$ to node $A$. The shortest paths for these two flows are indicated in the figure, respectively. In this case, we can see that there exists no coding opportunity. However, if source $A$ can detour its flow to path $A \rightarrow B \rightarrow D \rightarrow F \rightarrow G \rightarrow E$. This new path creates coding opportunities at nodes $B$, $D$ and $F$. The total number of packet transmissions can then be reduced. This motivates us to study how to increase the coding gain by coding aware routing.

2) Rate-Adaptive Multiple Paths Routing

Next, we show that multiple paths routing with traffic splitting can provide further coding gains when being combined with coding aware routing.

We illustrate the idea of coding aware multiple paths routing using a simple example (see Fig. 3). Suppose there are three flows: $f_1$: node $A$ to node $E$ with a data rate of 2 Mbps, $f_2$: node $E$ to node $A$ with a data rate of 1 Mbps, $f_3$: node $E$ to node $D$ with a data rate of 1 Mbps. With coding aware routing, flow $f_1$ and $f_2$ have coding opportunities at node $B$. However, due to the mismatching of the data rates between these two flows, only half of those packets of flow $f_1$ get coded with the packets of flow $f_2$. However, if node $A$ can split its traffic of $f_1$ and send half of them onto path $A \rightarrow B \rightarrow E$ for a rate matching with $f_3$ and reroute the remaining traffic onto path $A \rightarrow D \rightarrow C \rightarrow E$ for a rate-matching with $f_3$. In this case, further coding opportunities can be created at node $C$. The network throughput is improved as a result. This example motivates us to investigate the traffic splitting multiple paths routing in the design of a coding aware routing mechanism.

B. Routing Metric Selection

The selection of routing metric is crucial to the design of a routing protocol. Most of the traditional routing protocols for WMNs use cost measurements based on link status, such as hop count, Expected Transmission Count (ETX) [26], Weighted Cumulative Expected Transmission Time (WCETT) [27], etc. Although these metrics perform well in traditional networks, they work inefficiently with network coding. The following differences between traditional routing and coding aware routing require the definition of a new metric for the latter.

1. The goal of the coding aware routing is to reduce the total number of transmissions and increase the network throughput by finding more coding opportunities.
2. The gain of network coding depends mostly on the traffic pattern instead of the link status of the network.
3. Coding is executed on a per-node manner instead of on a per-link manner.

From the above differences, we can see that a cost metric for coding aware routing should be node-centric instead of link-centric. This motivates us to define a new metric: Required
Transmission Number.

When a node wishes to forward a packet belonging to a flow, it will try to explore the coding opportunity of this packet by using the opportunistic overhearing mechanism in COPE. If there is a coding opportunity for this packet, which means this packet will be coded and sent together with packets of other existing flows, hence the node does not need an extra transmission for this packet. However, if there is no coding opportunity for this packet, the node has to send this packet independently as a new transmission, which means one required transmission is necessary for this packet.

Definition 1 (Required Transmission Number)
Consider a flow $f$ follows a path $p = s \rightarrow n_0 \rightarrow n_1 \rightarrow \ldots \rightarrow n_k \rightarrow d$. The required transmission number associated with an intermediate node $n_i$, $0 \leq i \leq k$, for the flow $f$ is denoted by $\text{cost}(n_i)$, which is the number of required transmission(s) for $n_i$ to forward a packet $p$ belonging to flow $f$.

- $\text{cost}(n_i) = 0$, if there are coding opportunities at $n_i$ for flow $f$.
- $\text{cost}(n_i) = 1$, if there is no coding opportunity at $n_i$ for flow $f$.

The cost associated with a path $p$ is defined as the sum of required transmission number of its constituent intermediate nodes.

$$\text{cost}(p) = \sum_{0 \leq i \leq k} \text{cost}(n_i) \quad (1)$$

A path with smaller cost has higher priority to be chosen. A tie will be broken by hop distance. In the calculation of path cost, source node is not considered because it has to send all packets originated from itself in any cases.

Now we have defined a node-centric cost metric as motivated by Observation 1. Next, we move to Observation 2, from which we see that rate adaptive traffic splitting multiple paths routing can provide further coding gain. We need to investigate how to effectively split the traffic of a flow to maximize the total coding gain without causing much increase in protocol complexity. This motivates us to define the concept of Matched Coding Rate. This metric tells how much traffic of a flow is needed for this amount of allocated traffic to be coded effectively with other flows at a node.

Definition 2 (Matched Coding Rate)
Consider a flow $f$ follows a path $p = s \rightarrow n_0 \rightarrow n_1 \rightarrow \ldots \rightarrow n_k \rightarrow d$. The matched coding rate at an intermediate node $n_i$, $0 \leq i \leq k$, for flow $f$ is the maximal possible rate (if allocated) of subflow of $f$, which can be encoded with those other flows currently passing through node $n_i$. We denote this rate by $R(n_i)$. If there is no flow can be coded with flow $f$ at node $n_i$, we set $R(n_i) = \infty$, which means such node does not take effect in calculating matched coding rate of a path.

The matched coding rate associated with a path for $f$ is determined by its bottleneck node, which has the minimal matched coding rate $R(n_i)$ in the path. The reason is as follows. If we allocate an amount of traffic higher than $R(n_i)$ of the bottleneck node to the path, there will be a lot of packets cannot be coded at this node.

$$R_f(p) = \min_{0 \leq i \leq k} R(n_i) \quad (2)$$

Network coding exploits the broadcast nature of wireless medium. However, the broadcast medium is more susceptible to congestion. The congestion in at a node can be caused by heavy load at that node, the interference of the node’s neighbors and hiding terminals, etc. Therefore, we need to define a metric to model how much traffic can be allocated to a node without causing congestion. Many congestion detection mechanisms have been proposed. Here, we use the packet buffer queue length to estimate the rate that will cause the congestion at a node for simplicity. Accordingly, we define the following metric.

Definition 3 (Congestion Avoidance Rate)
Consider a flow $f$ follows a path $p = s \rightarrow n_0 \rightarrow n_1 \rightarrow \ldots \rightarrow n_k \rightarrow d$. We denote congestion avoidance rate associated with an intermediate node $n_i$, $0 \leq i \leq k$, by $C_f(n_i)$, which indicates the maximal rate of flow $f$ can be allocated to node $n_i$ without causing congestion. If there is no congestion at all, $C_f(n_i)$ is set to the wireless link capacity of the network.

The congestion avoidance rate associated with a path is defined as the minimal congestion avoidance rate of its constituent intermediate nodes.

$$C_f(p) = \min_{0 \leq i \leq k} C_f(n_i) \quad (3)$$

The design of our RCR mechanism is based on the above defined metrics. By applying these metrics in shortest path routing, RCR is expected to increase the throughput of the network.

C. Protocol Design

RCR is composed of two phases: Route discovery and route maintenance.

1) Route Discovery

This phase allows the source node of a flow to discover paths to the intended destination with the awareness of coding opportunities and coding rate requirements of flows in the network. RCR uses an on-demand routing strategy similar to Dynamic Source Routing (DSR) [8], and it requires each node in the network to maintain a list of its one-hop neighbors. When a new flow is to be routed, the source node floods a Route Request (RREQ) into the network. Several RREQs taking different paths may reach the intended destination. The destination node selects a set of paths according to the cost associated with each of these paths in terms of the metrics defined above and sends Route Reply (RREP) packets back to the source via each of the chosen paths. RCR builds multiple paths using above request/reply cycle.

RREQ Propagation

The main goal of RCR is to build a set of multiple paths with rich coding opportunities via traffic splitting to increase the throughput of the network. To achieve this goal, the destination node must know as more available paths as possible for it to select optimized path set according to the cost and matched coding rate of the paths. Cache replies at intermediate nodes are suggested to be set off for the destination node to learn the updated information associated with each end-to-end paths and to make the best decisions on path set selection.

When a source node $s$ wants to transmit a flow $f$ to an intended destination node $d$, it floods the RREQ packet, which contains the following information:
1. The source node ID and the sequence number, which can uniquely identify the packet.
2. The date rate of flow \( f \), which is denoted by \( R_f \).
3. Each intermediate node \( n_i \) that it encountered.
4. The required transmission number \( \text{cost}(n_i) \), matched coding rate \( R_c(n_i) \), and congestion avoid rate \( C_f(n_i) \) associated with each intermediate node \( n_i \).

Upon the receipt of a RREQ packet, an intermediate node \( n_i \) will first check if it will suffer from congestion. If \( C_f(n_i) \) is zero, \( n_i \) will discard the received RREQ packet. Otherwise, \( n_i \) executes the following procedures and then re-broadcasts the packet:

1. Get the required transmission number \( \text{cost}(n_i) \) and records it in the RREQ. This can be easily done by using the opportunistic overhearing mechanism in COPE.
2. Records the matched coding rate \( R_c(n_i) \) in the RREQ.
3. Records the congestion avoidance rate \( C_f(n_i) \) in the RREQ.

**Route Selection and Traffic Splitting**

The destination node will wait for a certain period of time to receive enough number of RREQs. The destination node \( d \) then uses the information carried by the received RREQs to choose a paths set \( P \) for flow \( f \). Procedures for calculating the path set \( P \) for a particular flow \( f \) are shown in Fig. 4. First, destination \( d \) examines all the received RREQs to extract all paths without loops and calculates the required transmission number, matched coding rate, and congestion avoidance rate of all these paths. Then \( d \) sorts these paths in terms of their required total transmission numbers in an increasing order. Next, node \( d \) allocates traffic to these paths with smallest transmission number first. The traffic allocated to a path must meet both requirements of matched coding rate and congestion avoidance rate associated with the path. Each path with non-zero allocated traffic rate will be enqueued into the path set \( P \).

Once the traffic allocation process is finished, the destination node \( d \) gets the path set \( P \) and it can then send RREPs back to the source node \( s \). Each RREP packet carries the following information: The node list of the entire path and the amount of traffic allocated to this path. Each intermediate node receiving a RREP will forward it further back to source node \( s \). After \( s \) gets all the RREPs, it can start to transmit traffic according to the rate allocated to each path.

2) **Route Maintenance**

Each flow has its lifetime. It is important to adjust the routes to keep the high efficiency of network coding with session in and session out. In RCR, each intermediate node \( n \) checks the presence of coding opportunity when forwarding a packet. If \( n \) finds out that coding opportunity for a flow \( f \) no longer exist due to the termination of other flow(s). Node \( n \) will report this change to the destination node of this flow by setting the required transmission number \( \text{cost}(n_i) \) to one and piggybacking this information with the data packet destined to the destination.

Destination node \( d \) checks and updates the required transmission number \( \text{cost}(p) \) of each path upon receiving the data packet. If node \( d \) finds out the \( \text{cost}(p) \) of path \( p \) becomes lower than an unallocated path in the discovered path set \( PATH \), \( d \) will execute the procedures in Fig. 4 again with the latest information to get a new path set which can maximize the coding opportunities.

**IV. Simulation Results**

In this section, we conducted simulations to evaluate the performance of RCR using ns-2 by comparing it with COPE. In the implementation of both mechanisms, we used 802.11b at the MAC layer. 100 static nodes were placed in an 800×800 m² square field. UDP flows were generated with randomly picked source, destination, lifetime and data rate, which ranges from 200 kbps to 2 Mbps. The radio range is approximately 100 meters and the interference range is 200 meters. In COPE, DSR is used as the underlying routing protocol for packet forwarding. We evaluated the performance of both protocols in terms of network throughput and the number of saved transmissions. Only data packets are considered in the computation of throughput. Each result was averaged over five random network topologies. All random topologies were generated by the setdest tools in ns-2.

In the test, we varied the offered load of the network to test the performance of both mechanisms. The offered load is controlled by the number of flows. Fig. 5 shows that the network throughput due to both mechanisms versus the offered load. From the figure, it is seen that RCR can improve the network throughput substantially as compared with COPE. On average, RCR provides 28 percent throughput gain over COPE, and with a maximum ratio of 51 percent. When the offered load is small, the potential coding opportunities are little. Thus RCR performs similarly to COPE. As the offered load grows, RCR improves the network throughput dramatically as compared with COPE by creating potential coding opportunities and splitting the traffic according to the coding rate requirements. When the offered load continues to grow, the congestion degree in the network increases. However, the performance of RCR falls much slower than COPE although both mechanisms deteriorated as the level of congestion increases. This is because RCR can detect the congestion at nodes and split the network.
traffic of each flow onto multiple paths to balance the load of the network and alleviate the congestion around nodes with heavy load.

Fig. 6 plots the number of transmissions saved by both mechanisms. It is observed although saved transmissions of both mechanisms decrease as network load increases beyond a certain threshold, RCR consistently provides great reduction of packet transmissions across a wide variety of offered load due to the coding opportunities created in the route discovery phase. The results again verify the benefit offered by RCR and hence the effectiveness behind our definition of routing metrics.

V. CONCLUSION

In this paper, we proposed a rate-adaptive coding-aware multiple paths routing mechanism to increase the throughput of wireless mesh networks. In our mechanism, the strategy of traffic splitting is used in the on-demand selection of multiple paths for each flow with the awareness of coding opportunities and required coding rate along each of these paths. Simulation results demonstrate that our mechanism achieves high performance in increasing the network throughput as compared with existing work. The proposed mechanism is simple, efficient, and easy to be implemented in wireless mesh networks with dynamic traffic.

ACKNOWLEDGMENT

This work was supported in part by High-Tech Research and Development Program of China under Grant No. 2006AA01Z207-1, the Knowledge Innovation Program of the Chinese Academy of Sciences under Grant Nos. KGCX2-YW-110-6, KG CX2-YW-120 and NSFC grant (Grant No. 60702074). The conference participation is supported by Nokia Bridging the world Program.

REFERENCES
