Enhance the Performance of Quality of Service in Mobile Adhoc Networks

M.Arumugapriya, Mrs.R.Renugadevi, Dr.K.Ramasamy , Dr.K.Vijayalakshmi

Abstract— Network coding, which exploits the broadcast nature of wireless medium, is an effective way to improve network performance in wireless multi-hop networks, but the first practical wireless network coding system COPE cannot actively detect a route with more coding opportunities and limit the coding structure within two-hop regions. An on-demand coding-aware routing scheme (ODCAR) for wireless Ad hoc networks is proposed to overcome the limitations specified above by actively detecting a route with more coding opportunities along the entire route rather than within two-hop regions. ODCAR achieves a tradeoff by adopting ETT as route metric in route discovery. Simulation results show that, compared with Ad-hoc on-demand distance vector routing (AODV), ODCAR can find more coding opportunities, thus effectively increase network throughput, reduce end to end delay and improve packet delivery ratio.

Index Terms— Ad hoc Networks, coding – aware, COPE, expected transmission time, routing metric , routing protocol, network coding .

◆

1 INTRODUCTION

MOBILE ad hoc networks (MANETs) consist of a collection of mobile nodes which can move freely. These nodes can be dynamically self-organized into arbitrary topology networks without a fixed infrastructure. The field of network coding was essentially born with the foundational paper by Ahlswede et al in 2000. The main idea in network coding is to allow each node of a network to combine together data from its in-edges in order to determine what data to transmit on its out-edges. This concept contrasts with the traditional operations of packet-switched networks, such as the Internet, in which each node must relay data (i.e., using routing) from selected in-edges to select out-edges. Network coding generalizes traditional store and forward routing techniques by allowing intermediate nodes in networks to encode several received packets into single coded packets before forwarding. Generally, routing refers to the flow of data packets from source node(s) to destination node(s) (i.e., unicast or multicast) where intermediate node(s) simply replicate and forward without any processing on received packets. Therefore, each node is able to create multiple copies of received packets and forward it on different lines. On the contrary, NC allows each node to perform an operation, for example linear combinations of received data packets before forwarding on different transmission lines.

Based on the theory of network coding, Katti et.al. Proposed a new architecture for wireless Mesh networks—COPE. In essence, COPE incorporates 'opportunistic listening' and 'encoded broadcast' to reduce the number of necessary transmissions. COPE is the first practical network coding system for multi-hop wireless networks and it largely increases network throughput.

It should be noted that the throughput increase gained by COPE directly depends on the number of coding opportunities, and if there is no coding opportunities, there would be no throughput increase. A good routing metric supports in computation of good quality network links. Designing routing metrics is vital in wireless Adhoc networks. The commonly used routing metrics are: hop count, expected transmission count (ETX), expected transmission time (ETT), weighted cumulative transmission time (WCETT), metric of interference and channel switching (MIC).Most commonly used routing metric is Hop count and agile to topology changes. But it does not consider for link load, data rate, interference experienced by the links. ETX takes the effects of packet loss ratios and path length into considerations but ignoring the data rate and interference etc. ETT is an extension of ETX and capture link capacity. Hop count is commonly used as a routing metric in Ad hoc networks. Using this metric new paths must be rapidly obtained whereas high-quality links may not be found in due time. So this metric is apt for ad hoc networks. This is essential in ad hoc networks because of user mobility. Expected Transmission Count (ETX) is the expected number of transmissions a node needs to successfully transmit a packet to a neighbor. To compute ETX, each node periodically broadcasts probes containing the number of received probes from each neighbor. The number of received probes is calculated at the last T time interval. A node A computes the ETX of the link to a node B by using delivery ratio of probes sent on the forward (df) and reverse (dr) directions. These delivery ratios are, respectively, the fraction of successfully received probes from A announced by B, and the fraction of successfully received probes from B, at the same T interval.

$$ETX = 1/df * dr \quad (1)$$

ETT is the product between ETX and the average time a single data packet needs to be delivered.

$$ETT = ETX * t \quad (2)$$

To calculate this time t,

$$t = S / B \quad (3)$$

Divide a fixed data-packet size (S) by the estimated bandwidth (B) of each link

---

- *M.Arumugapriya is currently pursuing masters degree program in computer and communication engineering in P.S.R.Rengasamy college of Engineering for women, India. E-mail: arumugapriya13@gmail.com*
- *Mrs.R.Renugadevi is currently workingas associative professor in P.S.R.Rengasamy College of Engineering for women, India. E-mail: renu.rajaram@gmail.com*
- *Dr.R.Ramasamy is currently working as principal and professor in P.S.R.RengasamybCollege of Engineering for women,India.E-mail: ramasamy@psrr.edu.in.*

2 RELATED WORK

Katti et al., [3] proposed COPE to improve the throughput of wireless ad hoc networks through the use of Network Coding(NC). COPE was the first NC architecture that performs coding of packets (from different sources) before forwarding them to wireless ad hoc networks. COPE includes three techniques; opportunistic listening, opportunistic coding, and learning neighbor state. COPE uses the ETX (Expected Transmission Count) metric as its routing function but overall routing is independent of coding opportunistic characteristics of wireless networks. Regarding centralized coding-aware routing, Ni et al. [4] developed ROCX (Routing with Opportunistically Coded eXchanges), an approach that reduces the total number of coded transmissions. ROCX is based on the ECX (expected number of coded transmissions) metric that estimates coding gain, which is the successful exchange of coded packets between two nodes through a coding point. For lossy wireless networks, the NCAR [8] (Network Coding Aware Routing) protocol has been proposed by Wei et al. NCAR solves the problem related to finding the best coding solution (at intermediate nodes) when there exists more than one unicast flow and more than one coding scheme. NCAR considers availability of potential coding opportunities in the route discovery process and the link delivery ratio in optimal route selection. An On-demand COPE-aware Routing (OCR) protocol has been added to COPE by Kai et al. By adding OCR to COPE, which is used to actively detect coding opportunities and improve the throughput of adhoc networks. OCR consists of three phases, i.e. code judgment , route discovery, and route maintenance. As a pretreatment to network coding, Kai et al. proposed the use of a triple structure (with three parameters; current node, number of input links to current node, and number of output links to current node) to label nodes. At the beginning, through triples talk, and following a simple rule, OCR eliminates redundant points that are not suitable for coding application. The simple rule is that only nodes with a higher number of input links rather than output links are ap-propriate for coding application. The simple rule is that only nodes with a higher number of input links rather than output links are appropriate for coding. The selection of suitable coding points and to follow COPE's condition of practical coding, Kai et al. use a quaternion (with four parameters) structure that presents the status of a packet in the sending queues of a node. The route discovery phase deals with the selection of the best route from source to destination (through RREQ and RREP packets) in terms of high coding opportunities, lowest consumption, and lowest hop path. The route maintenance phase is based on the AODV's route maintenance method that is a regular broadcast of HELLO packets. Coding Aware Opportunistic Routing (COAR) (Yan et al.,) [12] is a coding-aware opportunistic mechanism that integrates both opportunistic routing and NC for wireless mesh networks. COAR exploits the broadcast nature of wireless media and utilizes local state information of a node to make a decision regarding the forwarding of a packet. From awareness of coding opportunities, COAR has the ability to carry out packet forwarding (without any synchronization among nodes). The COAR algorithm con-sists of the following three major steps; forwarder set selec-tion, best forwarder selection, and priority-based forwarding. At the first step, before sending a data packet, each node selects its forwarder set whose selection is based on following two conditions. The first condition is related to a direct neighboring node of a sender and second condition is node's (distance) closeness towards its destination in terms of ETX. Afterwards, the best forwarder selection step chooses the node with higher coding opportunities as the forwarder for packet transmission. Timer-based priority forwarding as a third step minimizes the synchronization requirements and prevents redundant transmission. Le et al.[11] proposed DCAR (Distributed Coding-Aware Routing) protocol that has the following characteristics: DCAR has the ability to discover available paths and concurrently detect coding opportunities on entire paths. DCAR has the ability to find paths that have more coding opportunities and ultimately higher throughput that leads it to look beyond two-hops. In addition, Le et al. also introduced a new metric known as CRM (Coding-aware Routing Metric) that evaluates the performance of a path and facilitates the comparison of coding impossible and coding possible paths.

3 EXPECTED TRANSMISSION TIME ROUTING METRIC AND NETWORK CODING

In this section we calculate the Expected transmission Time as routing metric to determine the route to destination and perform the Network Coding of the proposed routing protocol.

3.1 Expected Transmission Time

Based on Expected Transmission Time (ETT), we proposed our coding-aware and ETT routing metric R_{ETTCA} . For data flow fn, We denote the path from source to destination as L, l(u,v) is a link with transmitter u and receiver v on path L. For separate data flows f1 ,f2 , f3 ,….,fn intersecting at node u, the ETX of every data flow on link l is denoted as R_{ETX1l} ,R_{ETX2l}, R_{ETX3l} ,….., R_{ETXnl} , then if they can be coded together , after coding at node u, the ETX of link l is maximum of R_{ETX1l} ,R_{ETX2l}, R_{ETX3l} ,….., R_{ETXnl} . Let' take 2 flows f1, f2 for example, in which f1 is an existing flow. If there exists a coding opportunity at node u, then the ETX value for link l is max(R_{ETX1l} ,R_{ETX2l}), the ETX for the new flow f2 is max(R_{ETX2l} - R_{ETX1l} , 0). That means if R_{ETX2l} is less than R_{ETX1l} , then f2 can take free ride on f1 .Otherwise , if R_{ETX2l} is greater than R_{ETX1l} , f2 needs R_{ETX2l} - R_{ETX1l}, extra transmission.Based on the above analysis, the coding aware expected transmission count R_{ETXnl} for data flow fn is defined as follows:

$$R_{ETXnl} = R_{ETXnl} - \alpha \min[R_{ETXnl}, \max(R_{ETX*l})] \tag{4}$$

* represent any flow that can be coded with fn at node u .
α =1; existing flow can be encoded with fn at u.
α=0; existing flow cannot be encodedwith fn at u.

$$R_{ETTCAnl} = R_{ETXCAnl}(S / B) \tag{5}$$

S- Average size of packets and B - current link bandwidth.

3.2 On demand Coding-aware Routing Protocol

Source node broadcast route request (RREQ) to discover route to Destination. Determine whether there is coding opportunity or not, before finding a route to destination and which packets can be coded together. To better illustrate the coding conditions. For conditions 1), packets are buffered by nodes and it sends out for a period of time. For condition 2), nodes will need to buffer packets overheard from neighbors for a period of time in the hope that coding opportunities might arise. Source node broadcast route request (RREQ) message, before broadcasting the RREQ, the following fields need to be initialized: Path record, which records the nodes it has traversed. Overhearing nodes (i.e. one-hop neighbors of the source node) and their corresponding parameters such as packet delivery ratio, received signal strength normalized data rate, interference ratio, etc. Existing flow path record, which records the path of existing flows at this node. Overhearing nodes of existing flows, which records the existing flows at this node.

Intermediate nodes receive RREQ and process, upon receiving RREQ, intermediate node checks whether it has already been included in the RREQ's path record. If so, discard the RREQ, intermediate node performs: Overhearing nodes list is updated by adding one-hop neighbors of intermediate node and its corresponding parameters into the list, thus the list gradually increases as RREQ travels through the network. Updating path record by attaching itself into the list. Updating overhearing nodes list of existing flows by attaching the overhearing nodes of existing flows at node. Updating existing flow path record by attaching the existing flows' path at node. Rebroadcast the updated RREQ. When destination receives RREQs calculates the cost of each RREQ received during a short period of time T (say 1 s) according to the metric the route is chosen which has the minimum ETT. According to coding condition, a node is judge whether there is a coding opportunity. Path record and overhearing nodes information of current flow in RREQ is used to find the coding opportunity. Route reply (RREP) contains path record and overhearing nodes. The entire selected path and overhearing nodes information is filled in the path record and overhearing nodes field of RREP respectively. Route reply is generated by destination using reverse path back to source. Intermediate nodes process and forward RREP. After receiving an RREP, intermediate node performs the following: Storing the entire path and overhearing nodes information for the current flow contained in RREP. Storing the coding record corresponding to this node in order to perform network coding later. Source node receives RREP, After the source node receives RREP, it begins to send data packets according to the selected path.

3.3 Packet Coding

Node takes the packet at the head of its output queue, checks which other packets in the queue may be encoded with this packet, XORs those packets together, and broadcasts the XOR-ed version. If there are no encoding opportunities, our node does not wait for the arrival of a matching codable packet. In COPE the node increase with additional information in each transmission when possible, but does not wait for additional codable packets to arrive. COPE performs XOR-ing packets of similar lengths, because XOR-ing small packets with larger ones produce wastage of bandwidth. Packet-size in the Internet varies from 40 and 1500 bytes. We can XOR packets of different sizes that limit the overhead of searching for packets with the same sizes. In this case, the shorter packets are padded with zeroes. The receiving node can easily remove the padding by checking the packet-size of each native packet. COPE only need to consider packets headed to different nexthops not packet headed to the same nexthops. COPE therefore maintains two virtual queues per neighbor; one for small packets and another for large packets (The default setting uses a threshold of 100 bytes). When a new packet is added to the output queue, an entry is added to the appropriate virtual queue based on the packet's nexthop and size. Searching for appropriate packets to code is efficient due to the maintenance of virtual queues. When making coding decisions, COPE first dequeues the packet at the head of the FIFO output queue, and determines if it is a small or a large packet. Depending on the size, it looks at the appropriate virtual queues. For example, if the packet dequeued is a small packet, COPE first looks at the virtual queues for small packets. COPE looks only at the heads of the virtual queues to limit packet reordering. After exhausting the virtual queues of a particular size, the algorithm then looks at the heads of virtual queues for packets of the other size. Thus for finding appropriate packets to code COPE has to look at 2M packets in the worst case, where M is the number of neighbors of a node.Suppose the node encodes n packets together. Let the probability that a nexthop has heard packet i be Pi Then, the probability, PD, that it can decode its native packet is equal to the probability that it has heard all of the n − 1 native packets XOR-ed with its own, i.e.,

$$PD = P_1 \times P_2 \times P_3 \times \dots P_{n-1} \qquad (6)$$

Consider an intermediate step while searching for coding candidates. We have already decided to XOR n − 1 packets together, and are considering XOR-ing the nth packet with them. The coding algorithm now checks that, for each of the n nexthops, the decoding probability PD, after XOR-ing the nth packet with the rest stays greater than a threshold G (the default value G = 0.8). If the above conditions are met, each nexthop can decode its packet with at least probability G. Finally, we note that for fairness we iterate over the set of neighbors according to a random permutation. Formally, each node maintains the following data structures. Each node has a FIFO queue of packets to be forwarded, which we call the output queue.For each neighbor, the node maintains two per-neighbor virtual queues, one for small packets (e.g., smaller than 100 bytes), and the other for large packets. The virtual queues for a neighbor A contain pointers to the packets in the output queue whose nexthop is A. Additionally the node keeps a hash table, packet info, that is keyed on packet-id. For each packet in the output queue, the table indicates the probability of each neighbor having that packet.

3.4 Packet Encoding

Algorithm

```
Pick packet p at the head of the output queue.
Natives = {p}
Nexthops = {nexthop(p)}
if size(p) > 100 bytes then
    which queue = 1
else
    which queue = 0
end if
for Neighbor i = 1 to M do
    Pick packet pi, the head of virtual queue Q(i, which queue)
```

```
if ∀n ∈ Nexthops ∪{i}, Pr[n can decode p ⊕ pi] ≥ G then
p = p ⊕ pi
Natives = Natives U {pi}
Nexthops = Nexthops ∪{i}
end if
end for
which queue = !which queue
for Neighbor i = 1 to M do
Pick packet pi, the head of virtual queue Q(i, which queue)
if ∀n ∈ Nexthops ∪{i}, Pr[n can decode p ⊕ pi] ≥ G then
p = p ⊕ pi
Natives = Natives ∪{pi}
Nexthops = Nexthops ∪{i}
end if
end for
return p
```

3.5 Packet Decoding

Packet decoding is simple. Each node maintains a Packet Pool, in which it keeps a copy of each native packet it has received or sent out. The packets are stored in a hash table keyed on packet id, and the table is garbage collected every few seconds. When a node receives an encoded packet consisting of n native packets, the node goes through the ids of the native packets one by one, and retrieves the corresponding packet from its packet pool if possible. Ultimately, it XORs the n − 1 packets with the received encoded packet to retrieve the native packet meant for it.

4 PROTOCOL IMPLEMENTATION AND PERFORMANCE EVALUATION

4.1 Protocol Implementation

We modify the source code of AODV in NS-2 (v2.31) to implement our proposed protocol. Two different routing schemes (i.e., AODV, ODCAR) are implemented using NS2 and their performances are compared in terms of network throughput, average end-to-end delay. The network throughput we adopted is ‘end-to-end throughput’, i.e. the sum of the throughput of all flows in the network as seen by their corresponding applications. The propagation model adopted is two-ray ground propagation model. We assume that 50 nodes are distributed randomly and the transmission range is 100×300 m. IEEE 802.11b is used for the MAC layer and the promiscuous mode is enabled to implement the pseudo-broadcast in COPE and ODCAR. Data rate for wireless link is set to 2 Mbit/s. All flows, which are randomly generated, are constant bit rate (CBR) flow with packet size 500 B. Simulation time is 500 s for each run.

4.2 Simulation results and Analysis

Scenario: Throughput per flow and end to end delay against different flow rates five flows are generated randomly at an interval of 2 sec to 10 sec.

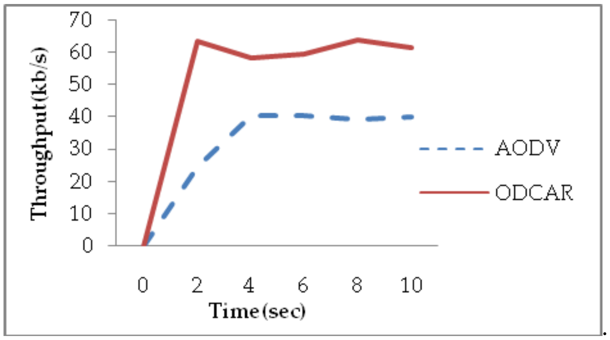

Fig. 1. Throughput comparison graph for AODV and ODCAR

As is shown in Fig. 1 and Fig. 2 that when the load of the network is relatively low, the performance improvement of network coding is not obvious due to lack of sufficient coding opportunities. But as load of the network increases, coding opportunities increases, the performance improvement of the proposed ODCAR becomes obvious compared to the AODV routing schemes.

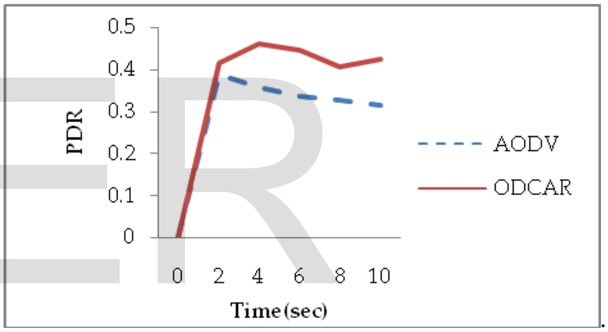

Fig. 2.Packet Delivery Ratio comparison graph for AODV and ODCAR

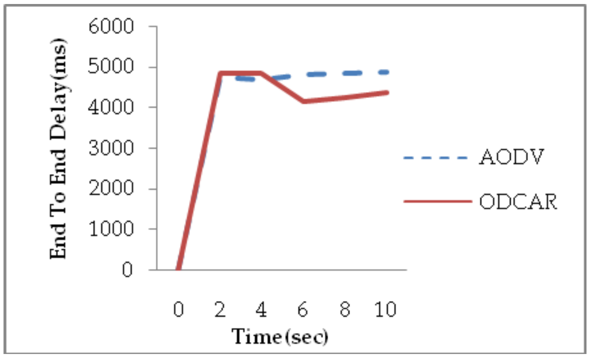

Fig. 3.End to End Delay comparison graph for AODV and ODCAR

Because the proposed ODCAR can actively detect coding opportunities in route discovery, thus OCAR can find more coding opportunities and achieve a higher throughput and lower end to end delay. It should be noticed that when load of the network increases from 4 sec to 10sec, the average throughput of AODV remains constant. On the other hand, when the load of the network from 0 sec to 2 sec, throughput of ODCAR increases. This is because network coding alleviates network congestion and more coding opportunities lead to larger performance improvement. In this experiment, the average throughput improvement of ODCAR over AODV is 22.8%. The average end to end delay improvement over AODV is about 0.7 s for ODCAR.

5 CONCLUSION AND FUTURE WORK

An ODCAR for Mobile Ad hoc network is proposed in this paper. The proposed routing scheme can actively detect coding opportunities on the entire path and discover a path with more coding opportunities to achieve higher throughput. ODCAR adopts R_{ETTCA} as the routing metric in route discovery, which incorporates potential coding opportunities, thus achieves a tradeoff between routing flows close to each other for utilizing coding opportunities. Based on the performance comparison of traditional unicast routing (AODV), coding-aware routing (ODCAR), it is demonstrated that although ODCAR requires more bytes in routing, the gains we achieve are effective network throughput increase, end to end delay reduction and network congestion alleviation. The routing overhead (Extra length of RREQ, RREP and HELLO message) of ODCAR is available. ODCAR piggybacks extra information on these routing control messages. So it has more overheads. Reducing this overhead is the future work.

REFERENCES

[1] C. Perkins, E. Belding-Royer and S. Das. “Ad hoc On-Demand Distance Vector (AODV) Routing”. http://www.faqs.org/rfcs/rfc3561.html.
[2] S. Biswas and R. Morris, “Opportunistic routing in multi-hop wireless networks," SIG-COMM Comput. Commun. Rev., vol. 34, no. 1, pp. 69-74, 2004.
[3] S.Katti, H.Rahul, W.Hu,D.Katabi ,M.Medard and J.Crowcroft.”XOR in the Air:Practical Wireless Network Coding”.Proceedings of ACM SIGCOMM,PP.243-254,2006.
[4] B.Ni, N. Santhapuri, Z.Zhong and S.Nelekuditi. “Routing with Opportunistically Coded Exchanges in Wireless Mesh Networks”. Poster session of SECON 2006.
[5] E.Rozner, J.Seshadri,Y.Mebta and L.Qiu,”Simple opportunistic routing protocol for wireless mesh networks”, in Proc. IEEE Wi-Mesh’06, pp.48-54,2006.
[6] Y. Lin, X. Zhang, and B. Li, “CodeOR: opportunistic routing in wireless mesh networks with segmented network coding," in Proc. IEEE ICNP08, pp. 13-22, 2008.
[7] B.Radunovic,C.Gkantsidis,P.Key, and P.Rodriguez,”An Optimization Framework for Opportunistic Multipath Routing in Wireless Mesh Networks’, in proc.ofIEEE INFOCOM,Minisyposium,2008.
[8] X.Zhang and B.Li, “Optimized Multipath Network Coding in Lossy Wireless Networks”, in Proc.of IEEE ICDCS,2008.
[9] L. Jilin, J.C.S. Lui, and C. Dah-Ming, “DCAR: Distributed Coding-Aware Routing in Wireless Networks," IEEE Trans. Mobile Comput., ol.9, no.4, pp.596-608, April 2010
[10] Y. Yan, B. Zhang, J. Zheng, and J. Ma, “CORE: a coding-aware opportunistic routing mechanism for wireless mesh networks," IEEE Wireless Communications, vol.17, no.3, pp.96-103, June 2010.
[11] S. Fu, K. Lu, T. Zhang, Y. Qian, and H. Chen, “Cooperative wireless networks based on physical layer network coding,” IEEE Wireless Communications, vol. 17, no. 6, pp. 86–95, Dec.2010.
[12] D. Koutsonikolas, W. Chih-Chun, and Y.C. Hu, “Efficient Network-Coding-Based Opportunistic Routing Through Cumulative Coded Acknowledgments," IEEE/ACM Trans. Networking, vol.19, no.5, pp.1368-1381, Oct. 2011
[13] B . Nazer and M. Gastpar, “Reliable physical layer network coding,” Proceedings of the IEEE, vol. 99, no. 3, pp. 438–460, Mar.2011.