AI-Augmented, Ripple-Diamond-Chain Shaped, Rateless Routing in Wireless Mesh Networks with Multi-Beam Directional Antennas

Ke Bao*, Fei Hu*†, and Sunil Kumar, Senior Member, IEEE‡
*Electrical and Computer Engineering Department, The University of Alabama, Tuscaloosa, AL, USA († Corresponding author)
‡Electrical and Computer Engineering Department, San Diego State University, San Diego, CA, USA

***Abstract*—Use of multi-beam directional antennas (MBDAs) allows a node to simultaneously communicate in multiple directions on its different beams, which can significantly improve the data throughput compared with omni-directional or single-beam directional antennas. However, conventional multi-hop wireless routing protocols cannot effectively exploit the benefits of MBDAs. In this paper, we present a novel routing scheme for a wireless mesh network (WMN) equipped with MBDAs, which has the following two features: (1) *Ripple-Diamond-Chain (RDC)* shaped routing: To exploit the multi-direction transmission capability of MBDAs, we use rateless codes to obtain loss-resilient *symbols* for data packets, which can be simultaneously transmitted on multiple beams of a node. Then we use *ripples* to differentiate the nodes in each hop in the tree topology of WMN. While the symbols are dispatched on multiple beams of a main path node in one ripple, they converge into another main path node in the next ripple. The entire route thus looks like a diamond chain, which can exploit the MBDA benefits. (2) *AI-Augmented path selection*: Our routing scheme is augmented by using two artificial intelligence (AI) algorithms. The fuzzy logic (FL) is used to define a weighted link quality, in order to adapt to different QoS requirements. The reinforcement learning (RL) is used to select the best main path based on the cumulative reward in all the links. In the simulations, we use the video as well as other time-sensitive traffic to evaluate efficiency of RDC routing, as well as intelligent path determination in WMN.**

Index Terms - Multi-beam directional antennas, Multi-hop routing, Wireless mesh network, Fuzzy logic, Reinforcement learning (RL), Q learning, Artificial intelligence, Rateless codes.

I. Introduction

Wireless mesh networks (WMNs) have become popular due to flexible network topology, low-cost maintenance, and large-scale coverage [1]. The wireless backbone nodes in WMN are called mesh routers (MRs). Each MR serves as a tree root as it can communicate with nodes in its tree, where each tree node is called a mesh client (MC) in Figure 1.

For a given transmit power, the directional antennas propagate the radio signals over a longer transmission range than the omni-directional antenna. However, a directional antenna can only transmit in one direction at a time, and is therefore blind to other directions. Multi-beam directional antennas (MBDAs) overcome this limitation by allowing simultaneous packet transmissions or receptions in multiple beams (or directions) [2].

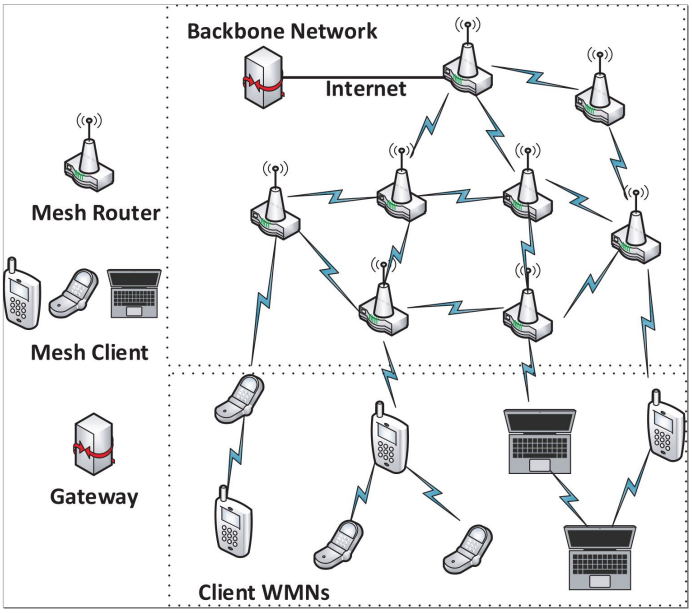


Fig. 1: WMN architecture

In this paper, we study an efficient routing scheme by exploiting the multi-beam transmission and reception capability of MBDAs. Conventional wireless routing schemes simply search the shortest path which can use only one beam of the MBDA, and thus waste the throughput in other beams. Also, the existing multi-path routing schemes do not have a tight control of the convergence property of the paths. For example, if the paths are widely disjoint, only a few nodes would be located in the intersection of multiple paths which can use multiple beams. Most of the nodes would still use single-beam communication since they only communicate with one node in the next hop.

Obviously, a one-shot link selection in each path, without considering a comprehensive, long-term link quality change, could result in sub-optimal routing. A Markov-like state transition model may be used to capture the link state change (in terms of fading, capture, interference, etc.) in different time slots. The path establishment process should select each link based on the link state transition model.

In this paper, we propose a novel WMN routing scheme

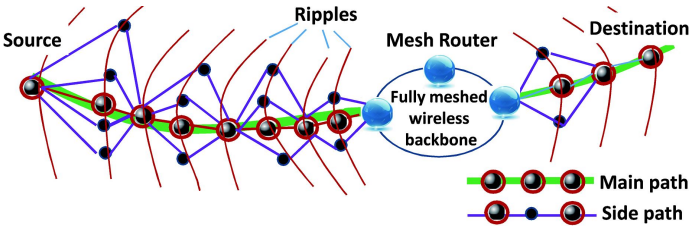


Fig. 2: Concept of ripple-diamond-chain routing.

that can fully utilize the multi-beam transmissions by using the *ripple-diamond-chain* (RDC) transmission, as illustrated in Figure 2. The nodes belonging to the same *ripple* have the same number of hops to the MR (the tree root). Note that the ripple IDs can be easily determined by using any ad hoc network routing protocol (such as the dynamic source routing (DSR) protocols). To form the RDC topology, first, the main path is searched, which consists of a series of links with the best link quality among all the links in the same ripple. To utilize the node's multi-beam communication capability, the side paths are then established by allowing other nodes in the same ripple to help in forward data packets. The main path nodes become the multi-beam divergence and convergence points (for traffic diffusion and aggregation). All the nodes in the main and side paths together form a chain of diamonds. When one ripple is sending data, the next ripple can only receive data. However, the 2-ripple-away nodes can also send data for pipelined transmissions, and so on. Therefore, besides the diamond chain formation, we also propose a ripple-to-ripple transmission schedule control scheme in the RDC-based routing.

However, the use of RDC requires that the following five challenging issues are addressed: (1) *Finding the main path:* The main path does not necessarily have the shortest distance to the MR. Conventional ad hoc routing schemes may not be suitable here since they do not aim to select a path with the best cumulative (long-term) channel quality across all links. (2) *Addressing varying channel conditions in multiple beams:* The rateless coding can be employed on time-varying channels since it does not need to explicitly adjust the symbol sending rates. However, the rateless codes need to be integrated with multi-beam transmission in MBDAs. (3) *Multi-beam transmission protocol in each ripple*: The RDC routing consists of ripple-to-ripple, localized communication, which typically belongs to the MAC layer. Our routing must therefore be integrated with an efficient multi-beam MAC in each ripple. Especially, we need a multi-beam oriented transmission control scheme to synchronize the multi-beam communication, since a MBDA cannot allow some of its beams to be in transmission and the rest in the reception mode due to self-interference. Then the problem is: How do we design an integrated RDC routing (across different ripples) and multi-beam MAC scheme (in each ripple)? (4) *The ripple-to-ripple schedule control* for alternative transmission (Tx) and receiving (Rx) operations across all the ripples in the main and side paths. (5) *QoS-oriented multi-beam transmission* to select appropriate side paths for different priorities of data.

Our solution to the above issues consists of a series of tightly coupled design modules, as illustrated in Figure 3. Three most important modules (or tasks) are: (1) Main path search (shown by the solid green line in Figure 2), (2) the diamond chain formation consisting of the main and side paths, and (3) the ripple schedule control (for ripple-to-ripple, pipelined transmissions).

For Task 1, we will first define a weighted link quality metric, by considering the following factors: (i) the decoding cumulative distribution function (CDF), which reflects the statistical distribution of the ACK feedback delays during the transmission of rateless coded packets; (ii) capture effect, which reflects the possibility of a node's directional antenna being captured by unproductive traffic; (iii) diamond transmission potential, which measures the probability of a link becoming part of the main path in an RDC route. Based on different QoS requirements, we adjust the weights of the above factors, and use the fuzzy logic (FL) to obtain the dynamic link quality metric. This link quality metric is used in the reinforcement learning (RL) based path search.

For Task 2, we add side paths to the main path to form a diamond-chain routing architecture. We then use rateless codes to encode the packets into different pieces (i.e., symbols), and dispatch them on multiple beams and paths. In each ripple (i.e., one-hop neighborhood), we propose a special multi-beam MAC protocol to achieve a collision-free neighborhood communication.

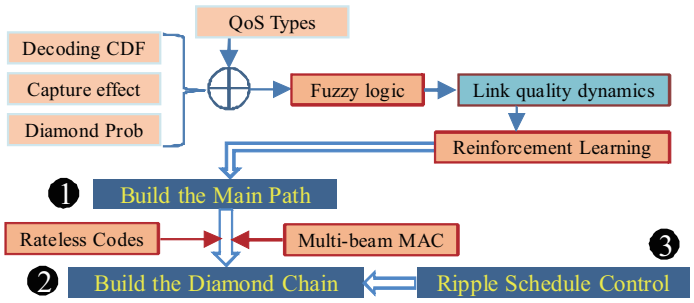


Fig. 3: Proposed solution: Function modules.

Task 3 is solved through a pipelined ripple-to-ripple schedule control scheme.

In summary, the main contributions of this paper are: (1) a comprehensive link quality measurement model based on the weighted integration of packet delay, antenna capture effect, and diamond transmission probability; (2) a RDC-shaped multi-beam routing protocol with rateless codes transmission as well as the ripple-to-ripple schedule control; (3) an AI (artificial intelligence) augmented main routing path establishment based on FL and RL algorithms.

This paper is a significantly extended version of [20]. We have added the rateless coding based multi-beam data transmission scheme as well as the learning based link quality control, including new simulation results.

The rest of the paper is organized as follows: The related work is briefly discussed in Section II, followed by the link quality modeling in Section III. Section IV describes the RL-based main path establishment, followed by the diamond

routing process in Section V. It also describes the ripple schedule control principle. The simulation results are discussed in Section VI, followed by the conclusion in Section VII.

II. Related Work

Multi-Beam Network Protocols: An MBDA (see Figure 4) has three main features: (1) it can easily detect the incoming signals in any beam by using the direction of arrival (DoA) estimation. (2) If it switches from Tx to Rx mode (or from Rx to Tx mode), all beams must be switched together into the same mode. (3) When the multi-beam antenna switches to Tx mode, the destination nodes in all beams should get ready in Rx mode at the same time. Otherwise, the beam bandwidth is wasted.

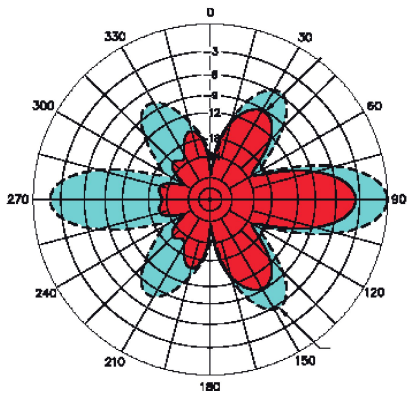


Fig. 4: Multi-beam directional antenna.

There is very little research on the network protocols specifically designed for MBDAs. Most of the existing schemes consider the single-beam directional antennas. There are a only few schemes on how to optimize the MAC protocols to exploit MBDA benefits. For example, an enhanced IEEE 802.11 distributed coordination function (DCF) framework was proposed in [3], in order to achieve concurrent multi-beam transmissions. The point coordination function (PCF) enhancement was described in [4], in order to adapt to multi-beam QoS requirements. A well-controlled multi-beam scheduling protocol was discussed in [5].

Multi-Path Routing: Multi-path routing can reduce the impact of channel fading by distributing the packets on multiple paths [6]. It can also schedule more traffic load in higher quality paths. Some schemes try to find node disjoint paths, in order to avoid a single-point failure [7]. However, some other schemes allow the existence of some intersection nodes among multiple paths [8]. In our proposed scheme, we want to fully utilize a node's capability of multi-beam transmissions, and assume that each node on the main path has enough buffer size to hold the aggregated traffic from other nodes' beams. Therefore, the RDC architecture is more suitable for our scheme as it tightly controls the traffic divergence and convergence schedules in each ripple.

AI-Enhanced Routing: In our previous work [9], we successfully used RL to solve the cognitive radio spectrum handoff issues. We also used Bayesian learning to detect idle spectrum in cognitive radios [10][11]. However, we are not aware of any work on AI-based multi-beam routing in WMNs. In this paper, we will use both FL and RL to search for the main path.

III. Link Quality Modeling

In this section, we model the link quality by defining a metric that indicates how good a link is, in terms of becoming a part of the main path. This metric integrates the following four factors: the CDF of rateless codes, capture effect, link bandwidth and diamond transmission potential. Besides these factors, this section will also explain how FL can be used to perform weighted integration of these factors. Our FL scheme can automatically assign different weights to each factor based on the QoS requirements of different traffic classes.

A. CDF of Rateless Codes

The rateless codes can adapt to channel conditions without the need for accurate sending rate adjustment. While conventional wireless networks need to select a proper sending rate from a few pre-determined options, the rateless codes send out the encoded packet pieces (called symbols) continuously, until the ACK message is received which indicates that the packet(s) has been successfully reconstructed at the receiver. After this, the sender starts to send the symbols of the next packet(s). The rateless codes typically introduce only a small amount of redundancy and perfectly fit the multi-beam transmission architecture, as the symbols can be simultaneously transmitted on separate links based on their respective link quality. Also, the priority-aware Fountain codes [12] can encode the packets based on their priority levels, such that the higher priority packets have higher redundancy (and can therefore be more easily recovered).

With the rateless codes, the ACK feedback can help us to indirectly deduce the link quality for the purpose of *main path* establishment. In this paper, we use the CDF concept to measure the link quality, where the CDF is defined as the probability with which the encoded packet can be recovered successfully without errors after a certain number of symbols (n) have been received [13]. Obviously, such a probability distribution (i.e., CDF curve) increases monotonically with the number of symbols received. The CDF curve (i.e., a probability distribution density (PDF)) is sensitive to the encoding parameters, channel conditions, and code block length. We need to collect only a small number of records, i.e., the data pairs with n (number of symbols sent between two consecutive pauses) and D (ACK feedback delay), in order to obtain the CDF curve.

The use of CDF allows a sender to know the proper pause timestamps, based on these probability curves. If CDF is not used, the sender must pause for ACK packet every time after sending the minimum number of symbols for each packet. If the receiver still cannot reassemble packet #1, no ACK is sent back (see Figure 5 (left)). Note that the use of frequent pauses wastes the channel resources.

With CDF (Figure 5 (right)), since the sender already knows the statistical distribution about how many symbols it should send before each pause, it can choose to pause at the right

time intervals after sending the required number of symbols for a packet. Because the ACK takes some time to get back to the sender side (ACK occupies over 18% of communication overhead [13]), the sender can send some packet #2 symbols before its first pause. As shown in Figure 5 (right), while the ACK for packet #1 is under the way, the sender has sent out some of the symbols for packet #2, before it pauses at the right time to wait for ACK of packet #1.

Note that the CDF can be easily obtained through an online learning algorithm, such as the Gaussian approximation or maximum-likelihood (ML) estimation. In this paper, we use the average ACK waiting delay within a unit time (such as one second) as the link quality metric [13]: $\overline{\Delta} = [\sum_{i=1}^{M-1}(t_{i+1} - t_i)]/M$, here t_i and t_{i+1} represent two consecutive pause time instants, and M is total number of pause times in a unit time interval.

B. Capture Effect

Since the multi-beam antenna typically keeps all its beams in the 'ON' state so that it can listen to incoming data from any direction, the capture problem is unavoidable. For example, in CSMA/CA based MAC protocols, a node can get stuck in carrier listening mode (i.e., captured by another neighboring node's traffic in a certain beam) because it believes that the beam has useful data for it [14]. Thus, this node cannot send data to other node(s) since the multi-beam antenna requires that all its beams are either in sending or receiving mode. One option for the node is to turn off its corresponding beam completely if it does not expect data packets for itself in the next communication phase. To build a high-quality main path in RDC-based routing, the links with high probability of being captured by other nodes should be avoided.

To define the capture level of a node i, we first define an indicator, I_{ij}. If node i is currently participating in active communication in beam j, $I_i = 1$; otherwise, it is 0. Even when the node i is not actively communicating with any other node (i.e., $I_i = 0$), it can be captured by other neighboring nodes. We define U_{ij} as the probability of node i being captured by its neighboring node(s) in its beam j. The value of U_{ij} can be obtained through the statistics of past capture events in different beams of node i. Assuming the node i has M beams, its total capture level, C_i, can be defined as: $C_i = \sum_{j=0}^{M}\{W_1 \times I_{ij} + W_2 \times U_{ij}\}$, where W_1 and W_2 depend on the importance of productive and unproductive traffic, respectively, in node i, and $W_1 + W_2 = 1$. Since unproductive traffic contributes most to the capture effect, we set up $W1 = 0.3$, and $W_2 = 0.7$.

C. Link Bandwidth

This metric represents the transmission capacity of a link in a given beam. Generally, the bandwidth measurement can be obtained by the effective operation time in a certain period, i.e., CITR (cumulative idle time ratio), which is defined as:

$$CITR = \frac{idle\,time\,period}{monitoring\,time\,period} \tag{1}$$

Then the CITR can be updated depending on how we evaluate the importance of the long-term average and a recent CITR value (this is similar to Internet RTT estimation):

$$CITR \leftarrow (1-a) \times CITR_{i-1} + a \times CITR_i \tag{2}$$

Where $CITR_{i-1}$ and $CITR_i$ denote the previous and current $CITR$ value, respectively, and the coefficient a is set to 0.7. The bandwidth of a link between the beam of the sender (node c) and the receiver (node x) in each ripple can be represented as:

$$BWF = min(CITR(c), CITR(x)) \tag{3}$$

.

D. Diamond Transmission Potential

Here, we introduce a metric to measure how well a path can achieve diamond transmissions, and find out how many links in this path can effectively participate in RDC transmissions. Recall that RDC transmission distributes the data into different beams in a ripple, and all traffic converges at the next ripple. Therefore, one cycle of a diamond transmission consists of two ripples (hops). In practice, some nodes in a chosen route may not be able to utilize diamond transmission due to the lack of links available for building side paths. In this case, only a single link can be used to forward data to the next ripple. We call such a link as the bottleneck link, as it can drag down the performance of diamond transmission. As one of the link quality metrics of diamond transmission, the number of bottleneck links, B_i belonging to node i, is maintained in each node.

In order to determine if a specific node is capable of launching a diamond transmission in the next two ripples, two factors should be considered. First, a node should be able to perceive all available routes to the destination node. This routing knowledge can be obtained via route discovery protocols such as AODV. Second, every node should possess a neighbor table that contains the information about all the accessible two-ripple (two-hop) neighbor nodes and the beam IDs through which it can reach the two-hop neighbors. Moreover, the number of available branches from a node is also a critical parameter since more branches mean more side paths.

We introduce an indicator variable, D_{ix}, to measure the potential of serving diamond transmissions for each node. Suppose node i and node x are a pair of two-ripple neighbors, and node i can forward the data to node x via $(n+1)$ different beams. Then $D_{ix} = n$. If two nodes have a single path, $D_{ix} = 0$, which means that the diamond transmission is not available for this two-ripple diamond unit. This value can be obtained from the two-ripple neighbor table in each node.

E. FL-based Link Quality Metric Integration

We use the simple additive weighting (SAW) method [15], which is a widely used FL method, to integrate the above four factors. To exemplify the concept of SAW, a decision matrix is given below, where the first column represents the link bandwidth (BW). Its range is $[0, 1]$, and the value approaching

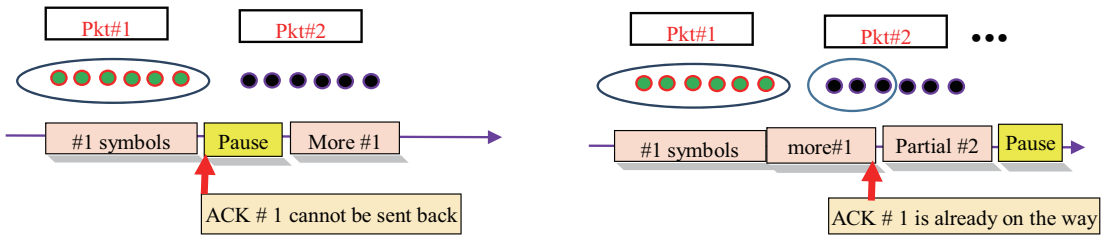


Fig. 5: Transmission of rateless coded symbols without using CDF (Left), and with CDF (Right).

1 denotes a larger link bandwidth. The second column denotes the CDF value (CDF), which represents the average number of symbols sent for successfully recovering the original packet. A smaller value of this variable represents a better link quality. The third column denotes antenna capture level (CP). We set its range as [0, 4]. The fourth column represents the diamond transmission potential (DT). Higher value in this column indicates more extra links available to serve as side paths.

$$D = \begin{array}{c} \\ B_1 \\ B_2 \\ B_3 \\ B_4 \end{array} \begin{array}{c} \begin{array}{cccc} BW & CDF & CP & DT \end{array} \\ \begin{pmatrix} 0.8 & 237 & 2.3 & 2 \\ 0.5 & 1104 & 1.4 & 0 \\ 0.3 & 300 & 0.2 & 1 \\ 0.7 & 400 & 3.8 & 3 \end{pmatrix} \end{array} \tag{4}$$

We use the following two formulae to normalize the above matrix. In both (5) and (6), each x_{ij} is the entry of B_i with respect to a specific metric. If the case is "the larger, the better", (5) is applied for normalization; otherwise, (6) should be used.

$$r_{ij} = \frac{x_{ij}}{x_j^{max}} \qquad i = 1, ..., 4 \qquad j = 1, ..., 4 \tag{5}$$

$$r_{ij} = \frac{x_j^{min}}{x_{ij}} \qquad i = 1, ..., 4 \qquad j = 1, ..., 4 \tag{6}$$

After applying (5) or (6), we can obtain the normalized matrix as:

$$D' = \begin{array}{c} \\ B_1 \\ B_2 \\ B_3 \\ B_4 \end{array} \begin{array}{c} \begin{array}{cccc} BW & CDF & CP & DT \end{array} \\ \begin{pmatrix} 0.8 & 0.84 & 0.1 & 0.5 \\ 0.5 & 0.18 & 0.14 & 0 \\ 0.3 & 0.67 & 1 & 0.25 \\ 0.7 & 0.5 & 0.05 & 0.75 \end{pmatrix} \end{array} \tag{7}$$

In the following, we discuss the settings of SAW weights for the five types of traffic with different QoS requirements:

- Speech and audio streaming (highest priority)- 100 kbps, up to 300 ms delay;
- Interactive video (e.g., video conferencing with 2nd highest priority) - 480p resolution, 10 frames per second, 512 kbps, up to 300ms delay;
- Live video streaming (with 3rd highest priority) - 720p resolution, 30 frames per second, 2Mbps, up to 2 s delay;
- Video on demand (such as Youtube video with 4th highest priority) - 720p resolution, 2Mbps, up to 5 s delay;
- File downloads (lowest priority)- 50MB file in 5 minutes (requires retransmission of lost/dropped packets).

These data flows have distinct preferences with respect to the setup of the metrics. For instance, the HD video requires a large bandwidth and high diamond transmission potential, whereas the audio and file downloads do not have high requirements on these two metrics. However, they may prefer a better channel quality and a low capture level to reduce the packet loss. Therefore, we should assign different SAW weights to the above four factors for each of the five types of data flows.

Since the audio traffic does not require high data transmission rate and therefore does not need diamond transmission, we set up the weight vector as:

$$W_{audio} = [0.3, 0.3, 0.2, 0.2].$$

The interactive video needs more bandwidth than the audio data, and more side paths are preferred. Thus, we set up the weights as:

$$W_{video} = [0.7, 0.8, 0.3, 0.5].$$

Both the HD live video and HD video on demand streaming require high and more side paths. We set up the weights as:

$$W_{HDvideo} = [0.9, 0.9, 0.4, 0.8]$$

For the file downloads that has the lowest priority, we set up the weights as:

$$W_{data} = [0.2, 0.4, 0.7, 0.1]$$

By multiplying the above weight vectors by the normalized matrix, we can obtain the Q-value for data flow, which is then used in the Q-learning based main path establishment (discussed in the next section).

IV. RL-Based Main Path Establishment

In this section, we explain the process of using the above link quality metric to seek the best main path based on RL scheme. This main path will become the "central pipe" of our proposed RDC routing scheme. Before we describe the RL-based main path establishment algorithm, we describe how the DSR can be extended to the multi-path DSR scheme. These available multiple paths will then be used to search the main path via the RL algorithm.

A. Multi-Path WMN Routing

The DSR [16] is a typical multi-hop route discovery protocol that aims to find the shortest path. As mentioned before, the shortest path does not guarantee that all of its links have the best quality. Therefore, we extend the DSR protocol to the multi-path DSR, in order to obtain all the candidate paths, which will be used to search for the main path.

If we ask a MR in the WMN backbone to issue the route request (RREQ) message to all its MCs, the DSR protocol can

easily find the hop ID for each node. We call these different hop IDs as ripple IDs since all nodes in the same ripple have the same hop distance to the MR (Figure 1). Each MC belongs to a certain ripple. In DSR, all intermediate nodes discard the RREQ messages they have received before. Through the following minor modifications of route discovery process in DSR, we can find all the available multiple paths between the source and destination.

- Every RREQ message caches the node IDs information along the path that it has traversed.
- Every node broadcasts the received RREQ messages, unless a RREQ message has been relayed by this node before. This rule prevents the count-to-infinity problem.
- The destination node receives multiple RREQ messages with the information of multiple available paths from a source node. The destination node sends RREP messages to source, corresponding to each RREQ.
- Thus, the source node obtains multiple available paths to a destination node from these RREP messages.

To possess sufficient information for route selection, after the above multi-path DSR, each node maintains a one-hop as well as two-hop neighbor table. Note that the tables should also have the information on which beam the neighbor is located in. The above discussed FL-based link quality metric is also put in the table for each beam (link).

In a nut shell, after a minor revision of the DSR protocol, we will get to know all the available candidate paths from a source to the destination. These paths and their links have different link quality levels. We need to use a cumulative method to search for the best path (i.e., with the best overall quality for all the links of the path). Obviously, such a path does not necessarily have the shortest hop count. This path becomes the "main path" in our RDC routing scheme (to be discussed in Section V). Next, we describe the RL (Q-learning) based algorithm to find the main path. RL is a typical cumulative optimization algorithm that considers the overall reward performance after performing link-to-link state and action update.

B. RL-Based Main Path Establishment

For a real-time main path search, we cannot use a model-based RL scheme due to its complex state space search. A model-free RL approach, called Q-learning [17], which uses a simple Q-table to represent the state space, can greatly speed up the path search. The Q-learning process can be realized by modeling the route selection as a Markov decision process (MDP), in which we find the best route to the destination by maximizing the expected rewards of all links. Figure 9 shows the principle of MDP-based main path search.

Briefly, a MDP consists of 4-tuple (S, A,T, R), in which S represents a finite set of states; A denotes a set of actions; $T = P_{s,s'}(a)$ represents the transition probability from state s to state s' when taking action a in state s; and $R_a(s, s')$ is the reward received when switching to state s' after applying action a. Normally, a MDP problem can be solved through five steps of iteration: (1) an agent (node) learns the entire MDP environment (i.e., WMN network conditions) and which state it is in; (2) Based on the current state, the agent adopts an available action a; (3) In the next phase, the agent transfers to the next state s' and obtains a reward value from the system; (4) The agent updates the path search policy based on the reward it just received; (5) Repeat the above 4 steps at the current state.

To adapt to the dynamic channel conditions as well as the decentralized WMN architecture, we adopt the Q-learning to efficiently find the optimal route (i.e., main path). Q-learning has the following five elements:

- States: Here the state of a node includes the parameters in each beam (link) such as its throughput, packet drop rate (PDR), bit error rate (BER), etc.
- Actions: The set of candidate actions $a_t \in A_t$ at each state S_t denote the available transmissions to the one-hop neighbors of the node. Such an action should point out which beam to use, and which neighbor to communicate with next.
- Rewards: Here the reward is the FL-based integration of the link quality metrics discussed in Section III.
- State transition: The state transition matrix $T = P_{s,s'}(a)$ is a fixed matrix, and each state transition probability can be determined beforehand based on the empirical data obtained before.
- Online learning: We will adopt Bellman principle (discussion next) to perform on-line policy search based on the cumulative rewards.

The Bellman optimality equation [9] is used to optimize the Markov decisions based on the long-term reward calculation. It is used as a utility function:

$$V^*(s) = \underset{a \in A}{max} E_\pi * \left\{ \sum_{k=0}^{\infty} \gamma^k r_{t+k+1} | s_t = s, a_t = a \right\} \quad (8)$$

Here $0 < \gamma < 1$ is a discount factor that confines the impact from the long-term decisions. When $\gamma = 0$, the node makes the decision in a myopic manner, i.e., without considering the long-term optimization. On the other hand, when γ approaches 1, the agent would emphasize the future reward, in which the RL system is more farsighted. In the above equation, r_t, s_t and a_t denote reward, state and action, respectively. $V^*(s)$ is the utility value for taking action $a = a_t$ at state $s = s_t$, and then executing the optimal policy π^*.

The Bellman optimality equation [9] can be denoted as the action-value equation $Q^*(s, a)$:

$$\begin{aligned} Q^*(s_t, a_t) &= E\left\{ r_{t+1} + \gamma V^*(s') | s_t = s, a_t = a \right\} \\ &= E\left\{ r_{t+1} + \gamma \underset{a' \in A}{max} Q^*(s', a') | s_t = s, a_t = a \right\} \\ &= E\left(r_{t+1} \right) + \gamma \sum_{s'} P_{s,s'}(a) \underset{a' \epsilon a}{max} Q^*(s', a') \end{aligned}$$

We can utilize the model-free Q-learning to optimize the main path selection by iteratively updating the Q-values for given links between nodes:

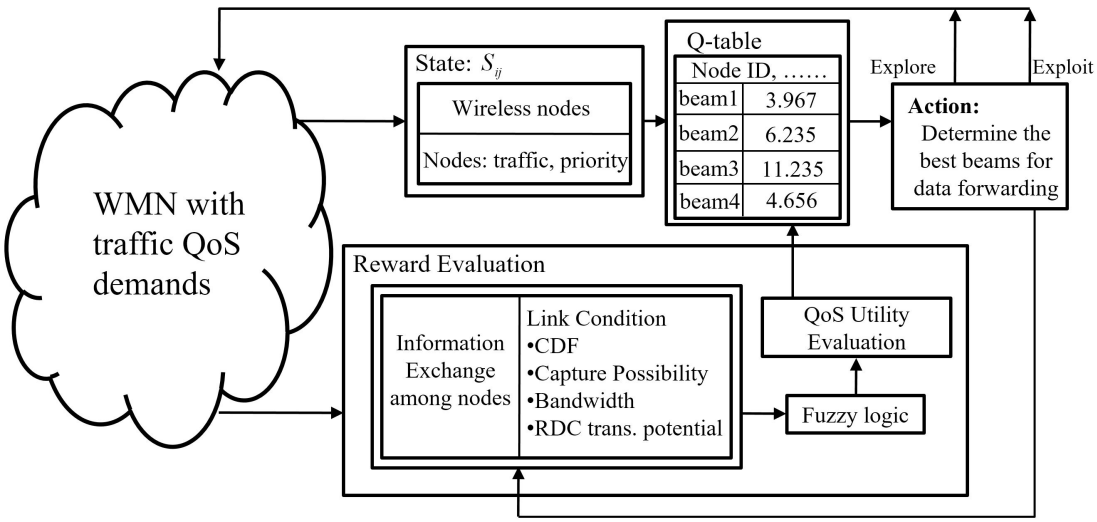


Fig. 6: Reinforcement learning for main path search.

$$Q(s,a) = (1-a)Q(s,a) + \alpha\left\{r_{t+1} + \gamma \max_{a' \in A} Q(s',a')\right\} \quad (9)$$

Besides states, action and discount factor explained previously, the other two parameters used in Q-learning are: (1) α: The learning rate is set between 0 and 1. Setting it to 0 means the Q-values are never updated, and therefore nothing is learned. Setting to a high value (such as 0.9) allows the learning to occur quickly. (2) max_{α}: The maximum reward that is attainable in the state following the current one, i.e., the reward for taking the optimal action thereafter.

In the beginning, every node of a WMN maintains a Q-table. Once a node is about to send out a packet, it may have multiple candidate one-hop neighbors according to the available routes found during the DSR-based multi-path route discovery process (Section IV.A). The node sends the packet to one of these candidates as an action. After the packet is sent, the reward r is collected and piggybacked by the ACK message, and the node updates the Q-value in the Q-table with the reward r. Then, the next node holding the packet becomes the current node in Q-learning. This process is repeated until the packet arrives at the final destination.

V. Proposed RDC-Shaped Routing

After the main path is established via the above RL scheme, we add side paths to it, in order to form a RDC topology that can fully utilize the multi-beam capability. In this section, we explain the formation of a RDC routing architecture as well as the ripple schedule control scheme.

A. Diamond Chain Formation

As described in Section IV.A, all the candidate paths between a source and destination are formed through the use of multi-path DSR. In DSR protocol, each node can maintain its 1-hop and 2-hop neighbor tables for side path establishment purpose. As shown in Figure 7, after the main path (red arrows) is formed, we can easily add side paths (red lines) around the main path by selecting the neighbors that belong to the same ripple (i.e., the same hop ID) as the next-hop relay nodes.

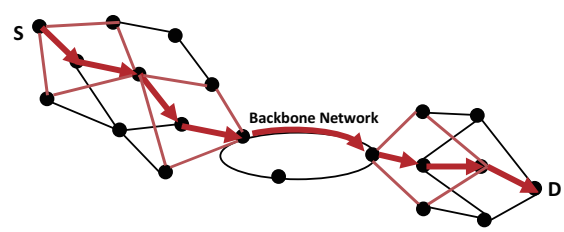
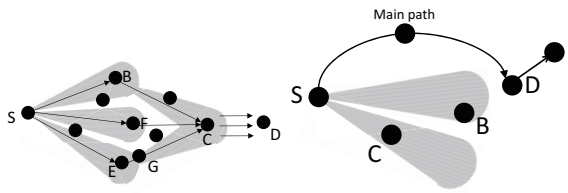


Fig. 7: Diamond chain: Add side paths to the main path.

Figure 8 (a) illustrates the basic idea of side path generation. After determining the main path (where S, F, and C are the main path nodes), through RREQ propagation, we know that nodes B, E, and G are in the same ripple as node F. Here S could maintain these nodes' profiles (such as their ripple IDs, beam directions, link quality, etc.) in its 1-hop and 2-hop neighbor tables. If node S chooses nodes B and E as the side path nodes, it distributes the data (i.e., rateless coded symbols) to nodes B, F and E in three different beams in the same superframe time duration (to be discussed in the next section). During the next superframe time duration, nodes B, F and E will forward their respective symbols to node C. This process is repeated every two ripples (hops). They thus form a diamond chain routing architecture. Each node in the main path serves as either the source of the divergence links (such as node S) or the destination of the convergence links (such as node C).

Since we use the rateless codes in diamond chain, the packet symbols go through the main path and side paths. Based on the 1-hop neighbor table, we know the link quality in each beam. We allocate the number of symbols to the main and side paths, proportional to their respective link quality. Recall that the link quality here is an integrated FL-based metric, as discussed in Section III.

Path collision issue: Some important issues should be



(a) Side path generation (b) Path collision problem

Fig. 8: The cases of path generation and collision in RDC routing [20].

considered during the diamond chain formation. One of them is the path collision problem as discussed below. In Figure 8 (b), suppose the main path is known, where S is the divergence node and D is the convergence node. Node S needs to find side paths among all of its 1-hop neighbors. Suppose node S establishes two side paths to nodes B and C, and sends data to both of them. In the next phase (superframe time), both nodes B and C would send data to node D.

If nodes B and C are in the same beam of the convergence node D, collision occurs because node D cannot simultaneously communicate with both of them [20]. To avoid this issue, we require any two main path nodes that are two ripples away (e.g., nodes S and D) exchange their 1-hop neighbor tables. Here the main path node D can use a multi-beam antenna DOA (direction of arrival) detection scheme [3][5] to find out that nodes B and C are located in its same beam ID. By exchanging the neighbor table, S can avoid the selection of both B and C as side path nodes (i.e., it selects only one of them) [20].

Handling the bottleneck links: Each diamond 'chain' of RDC route consists of nodes in a 2-hop range, including the first hop for data divergence and the second hop for convergence. However, a bottleneck link can exist when the number of hops from the sender to receiver is an odd number, as shown in Figure 9 (Left). Since node R receives more data than what it can send out at any time, R→D is a bottleneck link.

If there are multiple links available to the destination node, a simple solution is to use node disjoint paths in the last three hops, as shown in Figure 9 (Right). Another solution is to change the allocation of ripple-to-ripple air time (i.e., packet transmission duration). As shown in Figure 9 (Left), while A and B send the data to R during one time interval, R can be allocated a longer time interval after it gets the "token" to forward the data to the next ripple (Section V.C will discuss ripple-to-ripple pipelined transmission control).

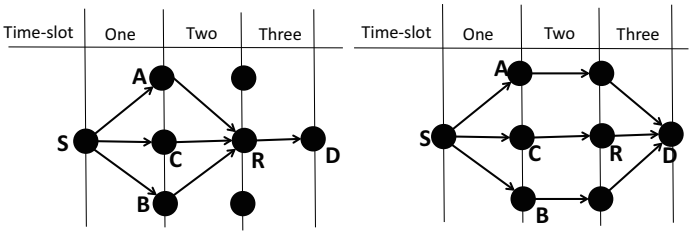


Fig. 9: RDC routing in the presence of a bottleneck link.

Bidirectional transmissions: Use of MBDAs and RDC routing can also enable bidirectional streaming. Due to the synchronized multi-beam transmission in MBDAs, the packet delivery in every time slot can be bidirectional. As shown in Figure 10 (Left), packet #1 from Level 1 is forwarded to the main path node in Level 2, while packet #2 from Level 3 can be simultaneously delivered to the same main path node. Figure 10 (Right) shows simultaneous multi-beam data transmission. Thus the data can be transmitted or received in both directions at the same time. This enables some applications, such as the interactive video calls.

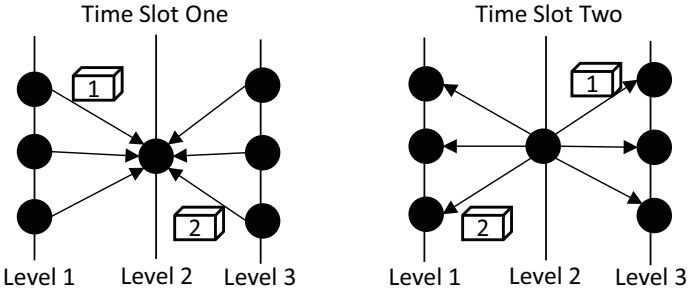


Fig. 10: Bidirectional streaming in RDC routing.

Algorithm 1 show the pseudo code of our RDC-based routing scheme.

Algorithm 1 RDC Processing

```
Input: Basic information of the data flow from application layer
if RDC is required then
    Find all available paths to Destination;
    Choose the best path according to the Q-value;
    Determine the length of the time-slot and other information;
    Send out the tokens to decided beams and broadcast SCH frames to the other neighbors;
    According to these control frames, the corresponding nodes start to forward data;
else
    Proceed to the regular two-layer MAC protocol.
end if
```

B. Multi-Beam MAC in Each Ripple

We need to deal with the localized, one-hop neighborhood communications in each ripple. A suitable transmission control strategy is designed since conventional IEEE 802.11 protocols cannot be directly used in MBDA equipped nodes. For completeness, we briefly describe our MBDA-oriented intra-ripple MAC scheme here. Note that all nodes in each ripple can either be in Tx or Rx status due to the properties of half-duplex MBDA and RDC routing scheme. We thus define a single round of Tx or Rx time duration in a particular ripple as a superframe. As shown in Figure 11, to adapt to the multi-beam QoS-aware communication requirements, we use the enhanced point coordination function (PCF) and enhanced distributed coordination function (DCF) in each superframe. General PCF and DCF have been defined in IEEE 802.11 standards.

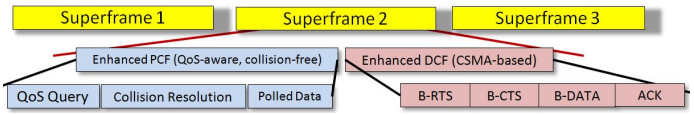


Fig. 11: Multi-beam MAC frame architecture.

Enhanced PCF operations: The node in the main path can serve as a point coordinator (PC) in PCF mode. Each PCF operation includes three phases (see Figure 11): The PC uses *QoS query* phase to ask each node (in side paths) in the next ripple to feedback its flow QoS parameters. The collision resolution phase is then used to solve the collisions during the *QoS response* phase since each beam of the node may have multiple nodes sending back QoS responses. The polled data phase is then used for official data transmission from the PC to each node of every beam.

Enhanced DCF operations: Figure 12 shows an example of multi-beam DCF operations. Note that nodes C and A are in the same beam of the target node (in the main path), which can talk with only one of them based on the traffic priority. Here the node uses (RTS-CTS-DATA-ACK) messages to communicate with node A first. Node C waits for DNAV (directional network allocation vector) based on the time specified in node A's CTS. An enhancement we made to the conventional DCF is that the backoff timer is adjusted to guarantee concurrent multi-beam transmissions for MBDAs [3]. Unlike IEEE 802.11 DCF, the contention window (CW) based backoff after DIFS is removed, in order to achieve the beam synchronized communication. But we require each node to wait for the CW-based random backoff before (not after) DIFS duration. And all beams should wait for the same time durations if not receiving CTS.

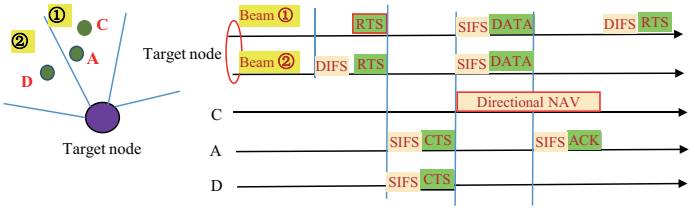


Fig. 12: The enhanced DCF phase [3].

Every main path node maintains a table, called DCF Beam Table (see Table I), and updates it after sending RTS in any beam. The main path node needs to know which nodes are actively communicating with it in each beam.

C. Ripple Schedule Control

From the above discussion, we can see that each ripple is either in traffic divergence (Tx) or convergence (Rx) state. Obviously, there is a need to control the ripple-to-ripple data propagation since two neighboring ripples cannot both be in the same state. Our ripple transmission schedule control scheme is inspired by the water ripple propagation pattern (with alternate crests and troughs). Since we are looking into the local communication in each ripple, we again discuss the enhancement to IEEE 802.11 MAC scheme.

Besides the traditional IEEE 802.11 control messages (DATA, NULL, RTS, CTS and ACK), we use two other messages, called RTS with Intelligent Feedback (RIF) and CTS with Intelligent Feedback (CIF) [3]. They are employed as "tokens" in the ripple protocol. A node cannot send out data without holding a token, as in the ripple-based MAC scheme [19]. Moreover, another control frame, called the SCH frame, is used to arrange the beam transmissions other than the ones negotiated via RIF and CIF.

Figure 13 exemplifies the schedule control of ripple-to-ripple transmissions. Suppose node S intends to forward an urgent data flow from left to right in Step 1, and the next node in the main path is R1. Before data transmission, S forwards the token frame - RIF to R1 (see the red arrow in Step 1). Due to the spatial reuse in MBDAs, other beams of S can also forward the tokens to the corresponding neighbors at the same time.

Figure 14 shows the time line of the above procedure. Before the data transmissions, MAC layer decides the length of the superframe via the data amount information contained in SCH and RIF frames. In other words, the time length of the superframe is determined by the node that holds the token in the data flow. For example, if node S intends to forward n bytes to R1 in this superframe, it will calculate the airtime for forwarding these n bytes. Then it sends this information to all its neighbors through the SCH and CIF frames. These frames can also carry the QoS information.

VI. Performance Evaluation

The performance of our RDC-based multi-beam routing protocol is discussed in this section, including the multi-beam ripple-to-ripple MAC protocol. In the simulations, we assume that every node in the WMN is equipped with MBDAs with six beams (beamwidth of 60 degrees), where each beam can support 2.5 Mbps of data rate, with a total of 15 Mbps data rate per node. In the following experiments, the time slot duration of 10ms is used, as recommended by IEEE 802.22 standard [18].

A. The Ripple-to-Ripple Transmission Performance

As discussed in Section V, our RDC routing consists of a series of localized diamond chain transmissions. Especially, we run an intra-ripple multi-beam MAC protocol with enhanced PCF and DCF modes in each ripple. This localized transmission control is important since the whole route consists of the pipelined diamond chain transmissions. Therefore, we first evaluate the communication performance in each ripple, in terms of throughput and delay.

We have implemented three types of localized transmission protocols in each ripple: (1) MB-PCF+DCF: Our multi-beam MAC with enhanced PCF and DCF, (2) MB-DCF: Multi-beam MAC with enhanced DCF (without enhanced PCF), and (3) OM-802.11: Conventional IEEE 802.11 protocol (uses omni-directional antennas, without multi-beam protocol enhancement) in each ripple. In Figure 15a, the x-axis is the average data generation rate in each node, and the y-axis represents the average throughput of each node. As we can see, our

TABLE I: DCF Beam Table

Beam ID	Active Nodes	Traffic Type	Airtime	Bidirectional Data	Antenna Capture Probaboloty	Rateless CDF Factor
1	D	CBR	2ms	ACKs	0.7	$101\mu s$
2	A	VBR	2.3ms	Reverse video flow	0.2	$60\mu s$
2	C	ABR	1ms	No reverse data	0.8	$91\mu s$
...	...	...	...	...	...	...

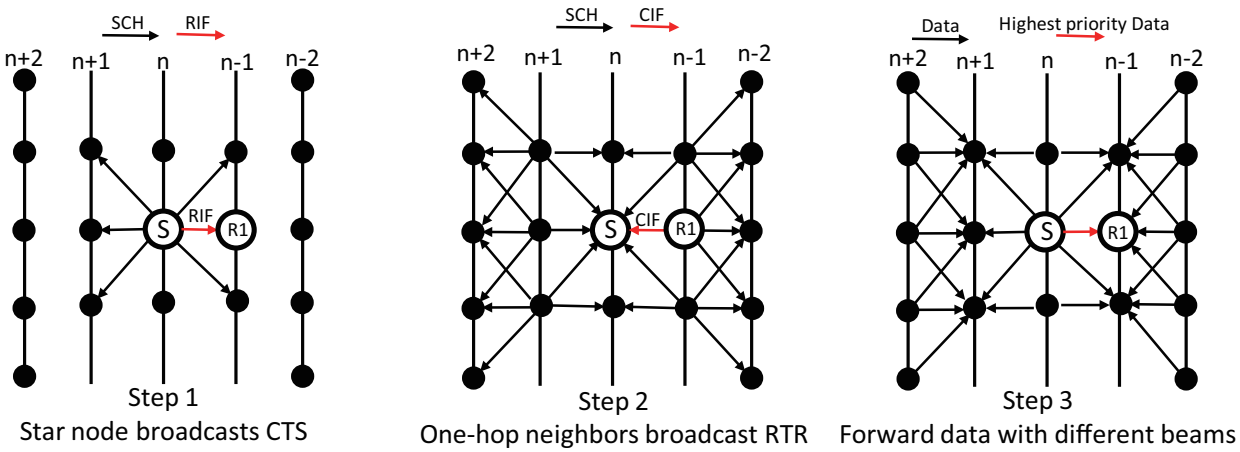


Fig. 13: Ripple schedule control: Three basic steps.

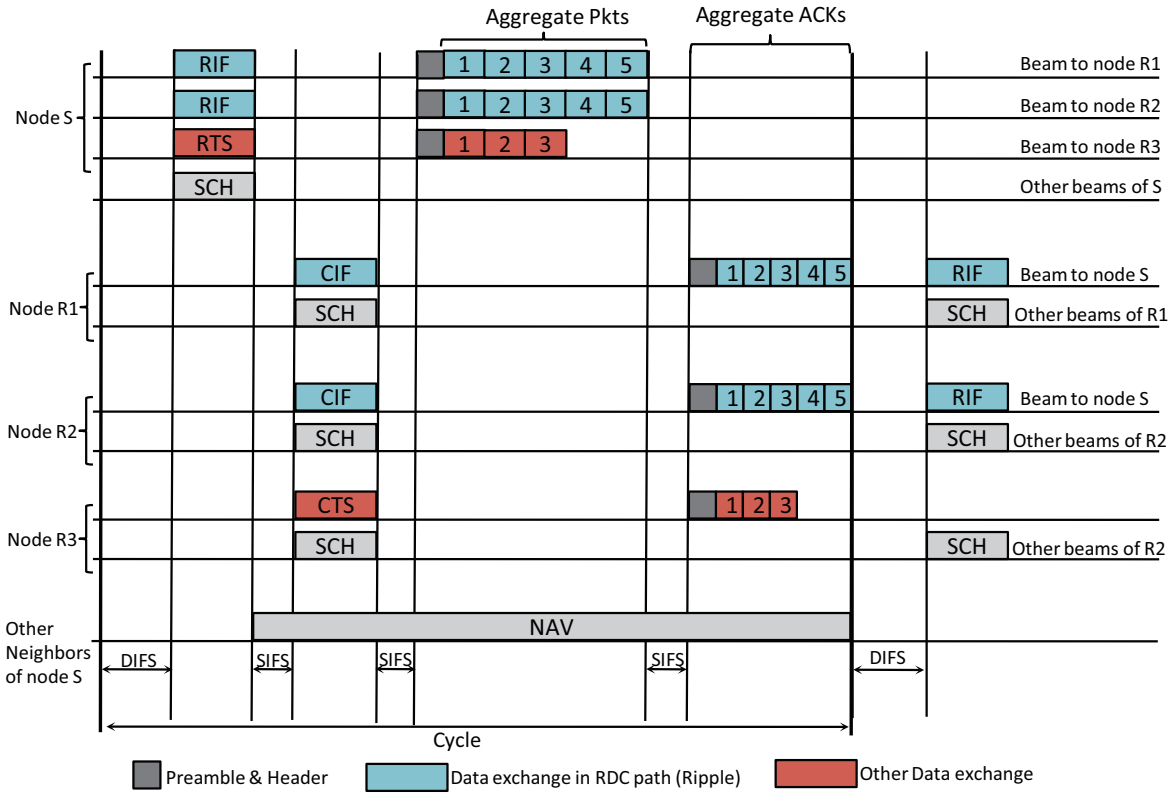
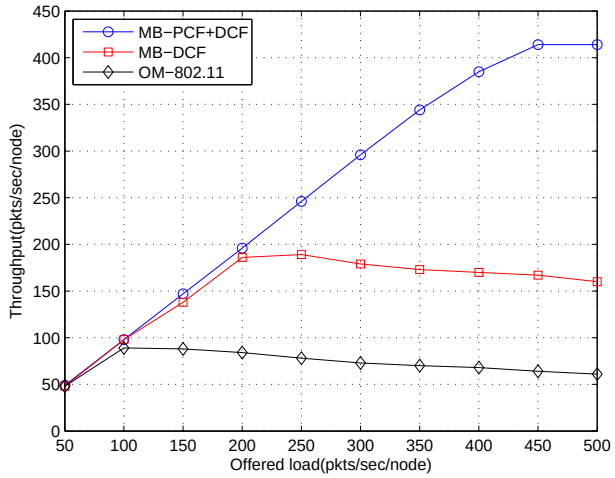


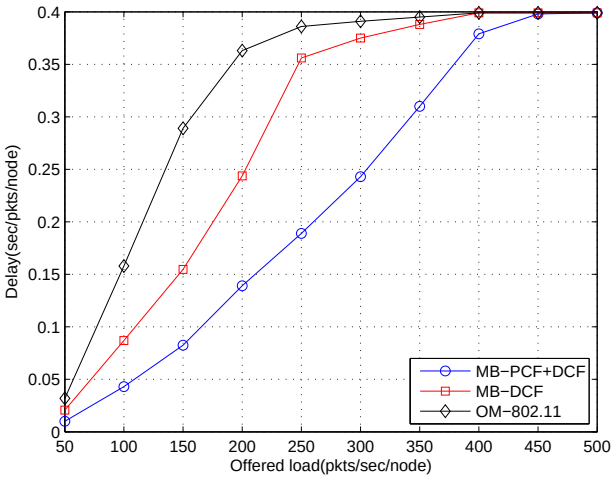
Fig. 14: Timeline of ripple schedule control scheme.

multi-beam MAC has a much higher throughput than the other two schemes. The conventional IEEE 802.11 protocol has the lowest throughput, because it uses the omni-directional transmission. Figure 15b shows the delay performance. Again, our scheme has the lowest transmission delay due to the multi-beam concurrent communication capability.

QoS Performance: In this set of experiments, we evaluate the localized ripple communication performance for data flows with different QoS priorities. Here we use two video flows with different priority and latency constraints. Priority #1 flow can tolerate only up to 200ms of end-to-end delay, whereas priority #2 flow can tolerate up to 300ms of end-to-end delay.

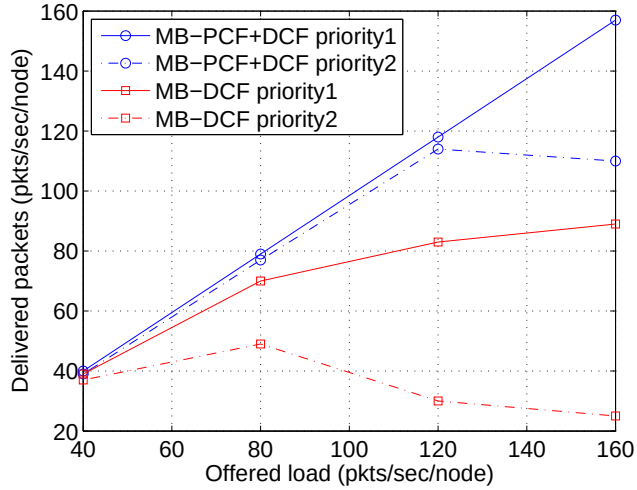


(a) Node Throughput

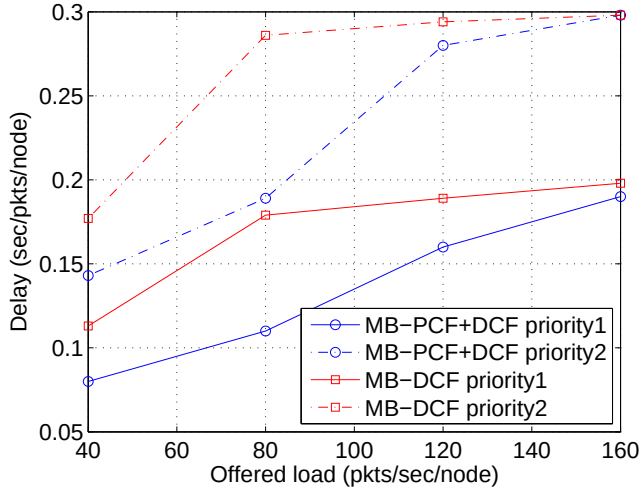


(b) Average Delay

Fig. 15: Performance of our multi-beam enhanced PCF+DCF, MB-DCF and conventional IEEE 802.11 MAC schemes.



(a) Node throughput



(b) Average delay

Fig. 16: QoS performance in ripple communication.

Figure 16 shows the throughput and delay performance of two multi-beam based protocols for both flows. Our MB-PCF+DCF protocol provides higher throughput with lower delay than the MB-DCF protocol, when the offered load increases. Furthermore, both protocols successfully allocate more resources to the higher priority flow, despite its lower latency.

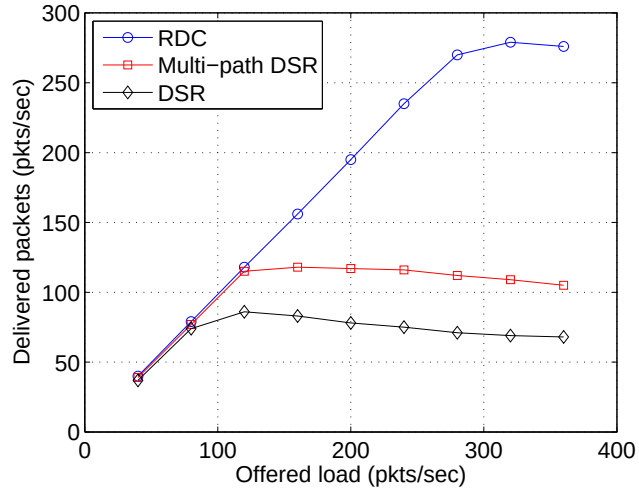
B. RDC Routing Performance

We compare the performance of our proposed RDC-based routing scheme (with multi-beam MAC in each ripple) against (i) DSR-based routing (with IEEE 802.11 MAC in each ripple), and (ii) multi-path DSR-based routing with our multi-beam MAC in each ripple. Note that the multi-path DSR scheme does not form the diamond chain. Instead, it randomly selects multiple, disjoint paths to deliver the data. Thus it has less nodes that are using multi-beam capabilities.

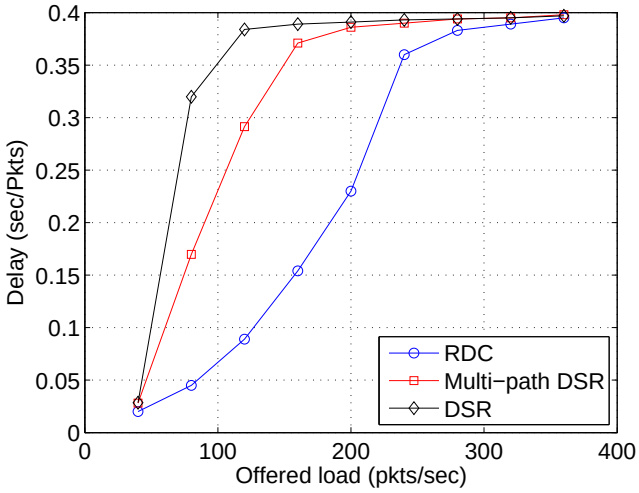
Figure 17(a) shows that our RDC-based routing scheme has much higher throughput than the remaining two routing schemes, when the offered load is higher than 100 packets per second. Figure 17(b) shows that our RDC routing scheme also has lower end-to-end delay than the remaining two routing schemes. These results confirm that the use of more beams and links in each node is especially helpful for heavy load.

C. Learning-Enhanced Routing Performance

Next, we study the performance enhancement achieved by using the RL-based main path establishment scheme. In this experiment, the discount rate (γ) of our Q-learning system is set to be 0.5. In Figure 18(a), the Q-learning based RDC routing scheme has slightly higher packet delivery rate than non-learning case. Figure 18(b) shows that the learning-based RDC routing has shorter end-to-end delay. This is because the use of Q-learning can find the best main path.

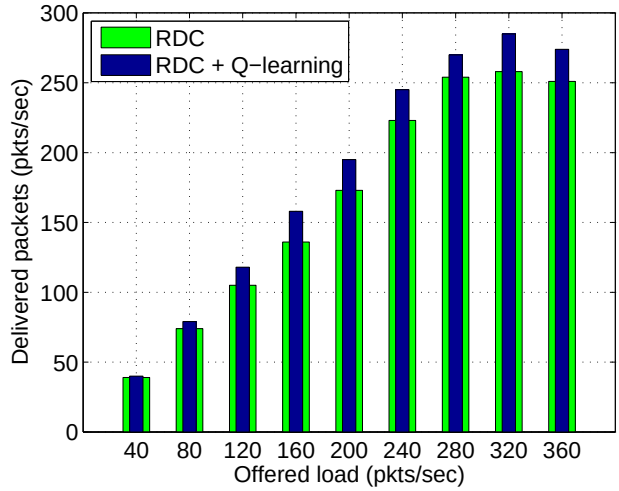


(a) Delivered packets per second

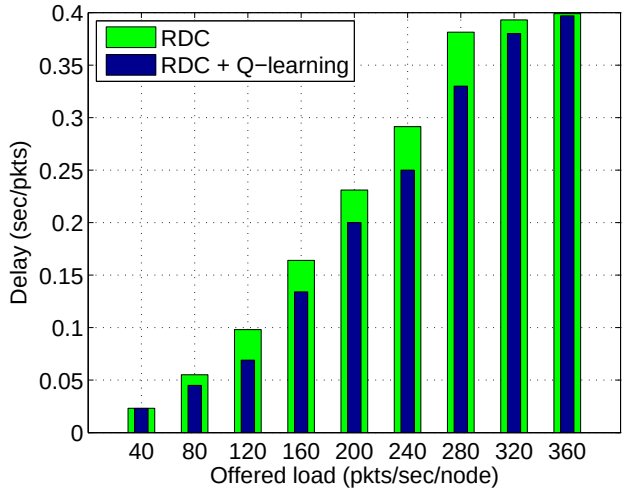


(b) Average delay

Fig. 17: Performance of our RDC routing, multi-path DSR, and single-path DSR schemes.



(a) Delivered packets per second



(b) Average delay

Fig. 18: Performance of RDC routing scheme with and without Q-learning.

D. Video Transmission Performance

We used the H.264/AVC encoded video transmission to test the efficiency of our RDC-based routing scheme. The video frame resolution is 800x600 pixels, encoded at a bit rate of 850 kbps and 30 frames per second. Figure 19 shows one video frame reconstructed based on the following three multi-hop transmission schemes: (a) the DSR-based routing, without using the multi-beam MAC in each ripple (instead, IEEE 802.11 is used), (b) DSR-based routing with our multi-beam MAC in each ripple, and (c) our proposed RDC routing with multi-beam MAC in each ripple. Figure 19a has the worst quality due to its ignorance of multi-beam antennas in both routing and MAC layers. Figure 19b has better frame quality because it uses the DSR scheme, along with our multi-beam MAC in each ripple. Figure 19c has the best frame quality because our proposed routing scheme forms the multipath diamond shaped routes to maximize the throughput. In this scheme, each beam helps to forward the rateless coded video packet symbols. Also, our multi-beam MAC protocol uses the enhanced PCF and DCF to exploit the multi-beam delivery capability in each ripple.

In order to show the advantage of our learning-based main path selection scheme, a HD video with 1280x720 resolution is transmitted at the bit rate of 1.6Mbps, with 30 frames per second. Figure 20 shows one reconstructed video frame based on the following two multi-hop transmission schemes: (a) RDC based routing without the use of learning algorithm, and (b) RDC based routing with Q-learning based algorithm to search the main path. Figure 20(a) has very good frame quality, which clearly shows the advantage of using learning-based scheme to build the main path. On the other hand, Figure 20(b) has a lower frame quality because it does not use learning-based RDC routing.

VII. Conclusions

In this paper, we presented a novel multi-beam routing protocol based on RDC formation, which can fully exploit

(a) DSR without multi-beam MAC (b) DSR with multi-beam MAC (c) RDC routing with multi-beam MAC

Fig. 19: Comparison of three routing schemes for streaming video quality.

(a) RDC routing with Q-learning (b) RDC routing without learning

Fig. 20: Effect of using Q-learning algorithm in RDC routing for video streaming.

the high data transmission capability of multi-beam antennas. The RDC-based routing scheme consists of a main path and multiple side paths. We also applied rateless codes which are suitable for multi-beam transmission, and can effectively work with lossy channels. We further used the AI algorithms (FL and RL) to build the main path that consists of the best quality links from a statistical distribution perspective. The ripple-to-ripple schedule control was designed to achieve pipelined transmissions in different ripples. Our simulation results with time-sensitive data and video transmission validated our RDC routing efficiency. In future, we will further improve our RDC routing scheme to better support the QoS metrics, including load balancing among all paths.

VIII. Acknowledgment of Support

Authors acknowledge Government's support in the publication of this paper. This material is based upon work funded by AFRL, under Contract No. FA8750-14-1-0075. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of AFRL and DOD.

References

[1] Ian F. Akyildiz, X. Wang, and W. Wang, "Wireless mesh networks: A survey" *Comput. Netw.*, ISDN Syst. 47, 4 (March 2005), 445-487.

[2] J. C. Porcello, "Designing and implementing Multibeam Smart Antennas for high bandwidth UAV communications using FPGAs," *IEEE Aerospace Conference*, pp. 1-12, 2-9 March 2013.

[3] V. Jain, A. Gupta, D. P. Agrawal, "On-Demand Medium Access in Multihop Wireless Networks with Multiple Beam Smart Antennas," *IEEE Trans. Parallel and Distributed Systems*, vol. 19, no. 4, pp. 489-502, April 2008.

[4] Z.-T. Chou, C.-Qi Huang, J.M. Chang, "QoS Provisioning for Wireless LANs With Multi-Beam Access Point," *IEEE Trans. Mobile Computing*, vol. 13, no. 9, pp. 2113-2127, Sept. 2014.

[5] L. Bao and J.J. Garcia-Luna-Aceves, "Transmission scheduling in ad hoc networks with directional antennas," *In Proc. 8th ACM Annual International Conf. Mobile Computing and Networking (MobiCom'02)*, New York, NY, USA, pp. 48-58.

[6] K. V. Muni, K. C. Sekaran, and A. Kandasamy, " Node - link disjoint multipath routing protocols for wireless sensor networks — A survey and conceptual modeling," *In Proc. International Conf. Advanced Computing, Networking and Security (ADCONS'11)*, pp. 405-414, 2011.

[7] W. Almobaideen, R. Al-Soub and A. Sleit, "Packet and circuit network convergence with OpenFlow," in *MSDM: Maximally Spatial Disjoint Multipath Routing Protocol for MANET,* Communications and Network, Vol. 5 No. 4, 2013, pp. 316-322.

[8] L. Wang, J. Griffioen, and K. Calvert, "ISE03-6: An Intersection-Based Multipath Routing Scheme," *IEEE GLOBECOM*, pp. 1-6, Nov. 27-Dec. 1, 2006.

[9] Y. Wu, F. Hu, S. Kumar, Y. Zhu, A. Talari, N. Rahnavard, and J. D. Matyjas, "A Learning-Based QoE-Driven Spectrum Handoff Scheme for Multimedia Transmissions over Cognitive Radio Networks," *IEEE J. Selected Areas in Communications*, vol. 32, no. 11, pp. 2134-2148, November 2014.

[10] X.-L. Huang, F. Hu, J. Wu, H.-H. Chen, G. Wang, and T. Jiang, "Intelligent Cooperative Spectrum Sensing via Hierarchical Dirichlet Process in Cognitive Radio Networks," *IEEE J. Selected Areas in Communications*, Vol. 33, No. 5, May 2015, pp. 771 - 787.

[11] X.-L Huang, G. Wang, and F. Hu, "Multi-Task Spectrum Sensing in Cognitive Radio Networks via Spatio-Temporal Data Mining", *IEEE Trans. Vehicular Technology*, Vol. 62, No. 2, pp. 809-823, Feb. 2013.

[12] Y. Wu, S. Kumar, F. Hu, J. D. Matyjas, "Cross-layer Forward Error Correction Scheme using Raptor and RCPC Codes for Prioritized Video Transmission over Wireless Channels," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 24, no. 6, pp. 1047-1060, June 2014.

[13] P. Iannucci, J. Perry, H. Balakrishnan, and D. Shah, "No Symbol Left Behind: A Link-Layer Protocol for Rateless Codes," in *MobiCom,* August 22-26, 2012, Istanbul, Turkey.

[14] R. R. Choudhury and N. H. Vaidya, *Capture-Aware Protocols for Wireless Multihop Networks Using Multi-Beam Directional Antennas*, Technical Report, UIUC. 2005.

[15] F. Farid, S. Shahrestani, and R. Chun, "A fuzzy logic approach for Quality of Service quantification in wireless and mobile networks," in *IEEE 39th Conf. Local Computer Networks Workshops*, pp. 629-636, 8-11 Sept. 2014.

[16] D. B. Johnson and D. A. Maltz, "Dynamic Source Routing in Ad-Hoc Wireless Networks," *Mobile Computing,* T. Imielinski and H. Korth, Eds., Kluwer, pp. 153-181, 1996.

[17] G. Santhi, A. Nachiappan, M. Z. Ibrahime, R. Raghunadhane, and M. K. Favas, "Q-learning based adaptive QoS routing protocol for MANETs," *International Conf. Recent Trends in Information Technology (ICRTIT)*, pp. 1233-1238, 3-5 June 2011.

[18] IEEE 802.11 Standards, see http://standards.ieee.org/about/get/802/802.11.html.

[19] R.-G. Cheng, C.-Y. Wang, L.-H. Liao, and J.-S. Yang, "Ripple: A Wireless Token-Passing Protocol for Multi-hop Wireless Mesh Networks," *IEEE Comm. Letters*, vol. 10, no. 2, Feb. 2006.

[20] K. Bao, F. Hu, E. Bentley, and S. Kumar, Diamond-Shaped Mesh Network Routing with Cross-Layer Design to Explore the Benefits of Multi-Beam Smart Antennas, *International Conf. Computer Communications and Networks (ICCCN), WiLAN Workshop*, Waikoloa, Hawaii, USA, August 1-4, 2016.