BREW: A Bandwidth Reservation Protocol for Multirate Anypath Routing in Wireless Mesh Networks

Xun Gao, Fan Wu, Xiaofeng Gao, and Guihai Chen
Shanghai Key Laboratory of Scalable Computing and Systems
Department of Computer Science and Engineering
Shanghai Jiao Tong University, China
gaoxunttkl@gmail.com; {fwu, gao-xf, gchen}@cs.sjtu.edu.cn

***Abstract*—In wireless mesh networks (WMNs), more and more applications, especially Video-on-Demand (VoD) services, appear to have strict requirements on guaranteed bandwidth. Most of the existing bandwidth reservation protocols for WMNs follow the approaches from wired networks, and cannot well adapt to the characteristics of wireless communication media, such as lossy and dynamic quality of wireless links. In this paper, we present the first Bandwidth REservation protocol for multirate anypath routing in WMNs, namely BREW. BREW aggregates bandwidths from multiple anypath routes to improve the flow acceptance ratio and the system-wide throughput, and can be integrated with most of the existing opportunistic routing (OR) protocols, especially the most efficient multirate anypath routing protocols. Furthermore, we extensively evaluate the performance of BREW. Our numerical results show that BREW outperforms the closely related work BOR/AC, in terms of flow acceptance ratio and system-wide throughput.**

I. INTRODUCTION

In recent years, multimedia streaming has increased dramatically in wireless networks. More and more applications, such as Video-on-Demand (VoD) and distributed file backup, appear in wireless mesh networks (WMNs). Some of these applications have strict requirements for quality of service (QoS) with bandwidth guarantees [1]. However, the characteristics of WMNs, such as the lossy and dynamic quality of links, make it challenging to provide guarantees on bandwidth. What's more, nodes share the communication media in WMNs using IEEE 802.11 family of protocols, and thus interfering neighbor nodes cannot transmit simultaneously. Thus a bandwidth reservation protocol is not easy to be implemented in WMNs. To address these challenges, several QoS schemes based on traditional routing paradigm have been proposed, such as INSIGNIA [2], SWAN [3] and CEDAR [4]. Unfortunately, none of them can work with the newly emerged opportunistic routing (OR) protocols.

To achieve high system-wide throughputs in lossy wireless environments, opportunistic routing protocols have been proposed. OR protocols elegantly aggregate the power of multiple paths to achieve significant throughput improvement, by letting any node that overhears a packet to participate in packet forwarding. Several representative works on OR have been proposed, such as ExOR [5], MORE [6], CCACK [7], CORMAN [8], and O3 [9]. These protocols calculate routes in WMNs that use a single transmission rate, but do not provide any bandwidth guarantees to data flows. Recently, Zhao et al. [10] proposed BOR/AC to provide bandwidth guarantees to flows with single rate OR protocols. Unfortunately, single rate OR protocols cannot fully exploit the capacity of the WMNs, and may cause network disconnection at high transmission rates. Laufer et al. [11] proposed the first multirate anypath routing protocol for achieving optimal end-to-end throughput. However, there is no bandwidth reservation protocol that can be applied to multirate anypath routing in WMNs.

In this paper, we propose the first Bandwidth REservation protocol for multirate anypath routing in WMNs, namely BREW. BREW makes admission control and bandwidth reservation for each data flow. It aggregates bandwidths from multiple anypath routes to improve acceptance ratio of flows and the system-wide throughput.

The contributions of this work are listed as follows:

- As far as we know, BREW is the first bandwidth reservation protocol with admission control for multirate anypath routing in WMNs.
- The BREW has three connected components: (A) network pruning and multirate anypath routing module prunes away bandwidth exhausted nodes from the network and computes an optimal anypath route from the rest of the network; (B) anypath route capacity estimation module calculates the capacity of a given anypath route; and (C) admission control and bandwidth reservation module interacts with the previous two modules to decide whether a coming data flow can be accepted or not. If yes, the module reserves the bandwidth for the coming data flow. The advantage of BREW is that it enables dividing a large data flow into several sub data flows whose

---

This work was supported in part by the State Key Development Program for Basic Research of China (Grant No. 2012CB316201), in part by China NSF grant 61170236, 61272443, 61133006, 61073152, 61202024 and in part by Shanghai Science and Technology fund 12PJ1404900, 12ZR1414900 and 12ZR1445000. The opinions, findings, conclusions, and recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the funding agencies or the government.

F. Wu is the corresponding author.

978-1-4673-5939-9/13/$31.00 ©2013 IEEE

bandwidths are reversed on different anypath routes.

- We extensively evaluate the performance of BREW, and compare it with BOR/AC [10]. Our numerical results show that BREW outperforms BOR/AC, in terms of flow acceptance ratio and system-wide throughput. With transmission rate 1Mbps, 2Mbps, 5.5Mbps, and 11Mbps, the flow acceptance ratio with BREW is up to 5.6, 3.6, 1.34, and 2.45 times that of BOR/AC and the system-wide throughput with BREW is up to 3.8, 2.29, 1.10, and 3.10 times that of BOR/AC.

The rest of the paper is organized as follows. In Section II, we present technical preliminaries. In Section III, we present the design of BREW. In Section IV, we show numerical evaluation results. In Section V, we discuss related works. In Section VI, we conclude the paper.

II. TECHNICAL PRELIMINARIES

In this section, we present the system model, and review the basic idea of multirate anypath routing protocol [11] and the method of calculating forwarders' workload.

A. System model

We consider a static wireless mesh network, denoted by a graph $G(V, E)$, where V is the set of nodes and E is the set of virtual links in the wireless mesh network. Each node can work at transmission rates from a set T. All the nodes use a CSMA/CA based protocol without RTS/CTS as the MAC layer protocol. The delivery probability from node $i \in V$ to node $j \in V$ under a certain rate $r \in T$ is denoted by p_{ij}^r, meaning that if a packet is transmitted from node i to node j at transmission rate r, then with probability p_{ij}^r the packet can be decoded. We note that with different transmission rates, a node can reach different sets of neighboring nodes.

B. Multirate Anypath Routing Protocol

In order to describe the basic idea of anypath routing, [11] defines hyperlink, denoted by an ordered pair (i, J), where $J \subseteq V$ is a nonempty subset of V. For each hyperlink (i, J), we have a delivery probability $p_{i,J}^r$ and a hyperlink distance $d_{i,J}^r$ under transmission rate r. The hyperlink delivery probability $p_{i,J}^r$ is defined as the probability that a packet transmitted from i is successfully received by at least one of the nodes in J with transmission rate r. Here, we assume that wireless receptions at different nodes are independent [12] [13]. Then:

$$p_{i,J}^r = 1 - \prod_{j \in J} \left(1 - p_{ij}^r\right). \tag{1}$$

In order to support multirate, the expected anypath transmission time (EATT) is used as a distance metric [11]. The hyperlink distance $d_{i,J}^r$ for each rate $r \in T$ is defined as:

$$d_{i,J}^r = \frac{1}{p_{i,J}^r} \times \frac{m}{r}, \tag{2}$$

where m is the maximum packet size. Here, $d_{i,J}^r$ is the time the node i takes to transmit a packet of size m at rate r over a lossy hyperlink with delivery probability $p_{i,J}^r$ to one of the nodes in J. Then, we calculate the distance D_i^r from a node i to the destination node d via a forwarding list J at rate r:

$$D_i^r = d_{i,J}^r + D_J^r, \tag{3}$$

where D_J^r is the hyper distance from set J to the destination. D_J^r is a weighted average on the distances of the nodes in the forwarding list J:

$$D_J^r = \sum_{j \in J} w_j^r D_j, \quad with \sum_{j \in J} w_j^r = 1, \tag{4}$$

where weight w_j^r is the probability of node j being the relaying node at transmission rate r, D_j means the minimum distance from a node j to the destination node d among all rates.

Based on knowing how to calculate the hyperlink distance $d_{i,J}^r$ and D_J^r, we can use the Shortest Multirate Anypath First algorithm (SMAF) [11] to calculate the shortest multirate anypath route from the source node s to the destination node d. The SMAF algorithm takes the delivery probabilities at each of the transmission rates between nodes as input, and outputs the distance EATT D_i, forwarding list $F_i \subseteq V$, and the best transmission rate $T_i \in T$ for each node $i \in V$. Formally,

$$\begin{aligned} D_i &= min\{D_i^r | r \in T\}, && (5) \\ T_i &= \underset{r \in T}{argmin}(D_i^r), && (6) \\ F_i &= \underset{J \subseteq V}{argmin}(d_{i,J}^{T_i} + D_J^{T_i}). && (7) \end{aligned}$$

C. Forwarder Workload Calculation

We now review the method from [6] to calculate the expected number of packets that a forwarder should transmit.

For any two nodes, i and j, let $i < j$ denote that $D_i < D_j$. Let z_i^r be the excepted number of transmissions that forwarder i must make to deliver one packet from the source node s to the destination node d with transmission rate r. For each forwarder j, whose EATT is higher than i, the expected number of packets forwarder i receives from forwarder j is $z_j^{T_j} p_{ji}^{T_j}$. Forwarder i should forward it only if no node with lower EATT hears the packet, this happens with probability $\prod_{k<i}(1 - p_{jk}^{T_j})$. Thus, for all forwarders having higher EATTs than forwarder i, the expected number of packets that forwarder i should forward, denoted by R_i, is:

$$R_i = \sum_{j>i} \left(z_j^{T_j} p_{ji}^{T_j} \prod_{k<i} \left(1 - p_{jk}^{T_j}\right) \right). \tag{8}$$

Here $R_s = 1$.

We note that the forwarding list F_i calculated by the SMAF algorithm is:

$$F_i = \{j \mid D_j < D_i \bigwedge p_{ij}^{T_i} > 0, j \in V\} \tag{9}$$

Therefore, it's not necessary to consider F_i during the calculation of R_i. The probability that at least one downstream node receives the packet from i is $1 - \prod_{k<i}\left(1 - p_{ik}^{T_i}\right)$. So, the expected number of transmissions that forwarder i must make is:

$$z_i^{T_i} = \frac{R_i}{1 - \prod_{k<i}\left(1 - p_{ik}^{T_i}\right)}. \tag{10}$$

D. Objective

The objective of this work is to design a bandwidth reservation protocol for multirate anypath routing in WMNs to improve the acceptance ratio of the flows, as well as the system-wide throughput.

Here, the acceptance ratio of the flows, denoted by p_f, is:

$$p_f = \frac{n_{ac}}{n_t}, \tag{11}$$

where n_{ac} is the number of accepted data flows, and n_t is the total number of data flows.

The system-wide throughput is the sum of the end-to-end throughputs of accepted data flows.

III. DESIGN OF BREW

In this section, we present the detailed design of BREW. BREW is composed of three major components: (A) network pruning and multirate anypath routing; (B) anypath route capacity estimation; and (C) admission control and bandwidth reservation.

A. Network Pruning and Multirate Anypath Routing

Given a WMN $G = (V, E)$, the SMAF algorithm is used to find a route $(D_i, F_i, T_i)_{i \in V}$ with the shortest distance from the source s to the destination d, without considering the currently available idle time on each of the nodes[1]. Therefore, we need to prune the network graph before inputting it to the SMAF algorithm, to prevent the SMAF algorithm from choosing nodes, whose bandwidths have been fully utilized.

Therefore, we set an idle time threshold θ, and remove the node, whose currently idle time, denoted by I_i, is less than θ, from the network.

After network pruning, the SMAF algorithm can be applied to calculate an optimal anypath route from the source node s to the destination node d in the rest of the WMN. Algorithm 1 shows the pseudo code of our network pruning and multirate anypath routing algorithm.

Algorithm 1 Network Pruning and Multirate Anypath Routing Algorithm (NP-MAR)

```
Input: Network graph $G = (V, E)$, source $s$, destination $d$,
    profile of link loss probabilities $(p_{ij}^r)_{i,j \in V, r \in T}$.
Output: Anypath route $(D_i, F_i, T_i)_{i \in V}$
1: for all $i \in V$ do
2:     if $I_i \leq \theta$ then
3:         $p_{ij}^r \leftarrow 0, \forall j \in V, \forall r \in T$.
4:         $p_{ji}^r \leftarrow 0, \forall j \in V, \forall r \in T$.
5:     end if
6: end for
7: $(D_i, F_i, T_i)_{i \in V} \leftarrow SMAF(G, s, d, (p_{ij}^r)_{i,j \in V, r \in T})$
8: return $(D_i, F_i, T_i)_{i \in V}$.
```

In Algorithm 1, if a node does not have sufficient idle time left (*i.e.*, $I_i \leq \theta$), we remove it from the network by disconnecting it from the other nodes in the network (*i.e.*, setting the delivery probabilities to and from the node to be 0) (Lines 1-6). Then, the pruned network is inputted to the SMAF algorithm to compute the shortest multirate anypath route (Line 7). The time complexity of Algorithm 1 is the same as that of SMAF algorithm, *i.e.*, $O(|V|log|V| + (|E| + |V|)|T|)$.

B. Anypath Route Capacity Estimation

Given an anypath route $(D_i, F_i, T_i)_{i \in V}$ computed by Algorithm 1, we need a method to calculate the capacity of the route, in order to determine how much bandwidth we can reserve on it. We present our method for anypath route capacity estimation in this subsection.

To deliver one packet from the source node s to the destination node d, the expected time that node $i \in V$ needs to take for transmitting packets, denoted by $t_i^{T_i}$, is:

$$t_i^{T_i} = \frac{m \cdot z_i^{T_i}}{T_i}, \tag{12}$$

where $z_i^{T_i}$ can be calculated by formula (10).

Since the communication media is shared by the nodes in the WMNs, a node should keep silent when its neighboring nodes are involved in transmissions. Consequently, to deliver one packet from the source node s to the destination node d, a node $i \in V$ needs to stay silent for time of length:

$$t_i^{\nabla} = \sum_{j : j \in V \wedge (p_{ij} > 0 \bigvee p_{ji} > 0)} t_j^{T_j}. \tag{13}$$

Then, the total time $t_i^{\star}$ that needs to be reserved on node i for delivering one packet is the sum of $t_i^{T_i}$ and t_i^{∇}:

$$t_i^{\star} = t_i^{T_i} + t_i^{\nabla}. \tag{14}$$

Considering the currently remaining idle time I_i on node i, we can determine the maximum packet delivery rate that can be supported at node i, denoted by M_i:

$$M_i = \frac{I_i}{t_i^{\star}}. \tag{15}$$

Since the capacity C of an anypath route is always limited by the bottleneck node, the capacity of the route can be calculated as follows:

$$C = m \cdot \min\{M_i | i \in V\}. \tag{16}$$

C. Admission Control and Bandwidth Reservation

The admission control and bandwidth reservation module interacts with the previous two modules to decide whether a given data flow should be accepted or not. If the data flow is accepted, the module determines the bandwidth reservation that can satisfy the need of the data flow.

We recursively call Algorithm 1 to compute the current shortest distance anypath route $(D_i, F_i, T_i)_{i \in V}$ from the source node s to the destination node d, and calculate the capacity C of the route. Then, we try to satisfy the (remaining)

---

[1] In this paper, we keep track of the idle time for each of nodes in the WMN, since it is independent of the transmission rate. For consistence with existing works on bandwidth reservation, we transform the time reserved on a node to bandwidth by multiplying its transmission rate.

bandwidth request Q of the data flow f with the current route. Specifically, we distinguish two cases:

- The capacity C of the current anypath route is less than the (remaining) bandwidth request Q. We create a sub-flow with throughput C, and tentatively reserve the maximum bandwidth on the route for the sub-flow. If this sub-flow's bandwidth reservation is sealed later, then the sub-flow will run independently to deliver a throughput of C for the data flow f. For each node i on the current anypath route, the (tentative) bandwidth , denoted by W_i, reserved in this round is:

  $$W_i = T_i \cdot t_i^\star \cdot \min\{M_i | i \in V\}. \qquad (17)$$

  Then, the remaining bandwidth request becomes

  $$Q = Q - C. \qquad (18)$$

- The capacity of current anypath route C is larger or equal to the (remaining) bandwidth request Q. This means that the data flow j can be accepted. Again, we create a sub-flow with bandwidth Q, and reserve the bandwidth on the current route. For each node i on the current anypath route, the bandwidth reserved in this round is:

  $$W_i = T_i \cdot t_i^\star \cdot \frac{Q}{m}. \qquad (19)$$

  Besides, the tentative bandwidth reservations made in previous rounds are all sealed.

Algorithm 2 shows the pseudo codes of our algorithm for admission control and bandwidth reservation.

In Algorithm 2, $TEMP$ stores the idle time of each node $i \in V$ before new flow f arrives. If no anypath can be found by Algorithm 1 (NP-MAR), meaning that there is no more anypath route to provide bandwidth for data flow f, each node $i \in V$ that participates in bandwidth reservation for data flow f should release the tentatively reserved bandwidths, and data flow f is rejected (lines 5-7). If the capacity of current anypath route C is less than the (remaining) bandwidth request Q, then we tentatively reserved the maximum bandwidth on the route for the sub-flow, and a next round bandwidth searching is needed (lines 10-15). If the capacity of current anypath route C is larger or equal to the (remaining) bandwidth request Q, this means that flow f can be accepted. We then accept data flow f and confirm its reservation (lines 17-21).

Algorithm 2 Admission Control and Bandwidth Reservation Algorithm (ACBR)

Input: Network graph $G = (V, E)$, source s, destination d, link loss probabilities $(p_{ij}^r)_{i,j \in V, r \in T}$, bandwidth requirement Q of the new flow.

Output: Bandwidth reservation $(W_i^k)_{i \in V, k \in U}$, where U is the set of sub-flows.

```
 1: $TEMP \leftarrow I$; $k \leftarrow 0$.
 2: while $TRUE$ do
 3:     $k \leftarrow k + 1$; $U \leftarrow U \cup \{k\}$.
 4:     $(D_i, F_i, T_i)_{i \in V} \leftarrow NP\text{-}MAR(G, s, d, (p_{ij}^r)_{i,j \in V, r \in T})$
 5:     if $D_s = \infty$ then
 6:         $I \leftarrow TEMP$.
 7:         return Flow rejected.
 8:     else
 9:         Calculate route capacity $C$.
10:         if $C < Q$ then
11:             $Q \leftarrow Q - C$.
12:             for all $i \in V$ do
13:                 $W_i^k \leftarrow T_i \cdot t_i^\star \cdot \min\{M_i | i \in V\}$.
14:                 $I_i \leftarrow I_i - \frac{W_i^k}{T_i}$.
15:             end for
16:         else
17:             for all $i \in V$ do
18:                 $W_i^k \leftarrow T_i \cdot t_i^\star \cdot \frac{Q}{m}$.
19:                 $I_i \leftarrow I_i - \frac{W_i^k}{T_i}$.
20:             end for
21:             return Flow accepted and $(W_i^k)_{i \in V, k \in U}$.
22:         end if
23:     end if
24: end while
```

D. Illustrative Example

We use an example to illustrate the process of using BREW in Figure 1. Figure 1(a) shows the initial status of a WMN. Here, we just consider two transmission rates: 5.5Mbps and 11Mbps. The values in the circles are the current idle time of the nodes. The pairs of delivery probability and transmission rate are shown beside virtual links. Figure 1(b) shows that when a data flow f_1 with 1Mbps bandwidth requirement comes, BREW first finds the optimal anypath route from S to D using NP-MAR (*i.e.*, $\{S \rightarrow (A, B) \rightarrow D\}$). The capacity of the anypath route is 2Mbps, which is larger than the bandwidth requirement of f_1. So BREW accepts the flow, and makes bandwidth reservation for it by updating nodes' idle time. Figure 1(c) shows that when a data flow f_2 with 2Mbps bandwidth requirement comes, BREW again finds the optimal anypath route $\{S \rightarrow (A, B) \rightarrow D\}$, and calculates the capacity of route. Unfortunately, the capacity of the anypath route is 1Mbps, which is less than the bandwidth requirement of f_2. So BREW tentatively creates a sub-flow for f_2, and reserves the whole bandwidth on the route for the sub-flow. Thus, the remaining bandwidth requirement of f_2 is 1Mbps. Then BREW tries to find a new anypath route to provide bandwidth for the remaining bandwidth requirement, but no anypath can be found. So BREW releases the tentatively reserved bandwidths, and rejects f_2. Figure 1(d) shows that when another data flow f_3 with 1Mbps bandwidth requirement comes, BREW finds the optimal anypath route $\{S \rightarrow (A, B) \rightarrow D\}$ whose capacity is 1Mbps. So f_3 is accepted. After making reservation for f_3, the idle time of node A becomes 0, and A cannot participate in packet forwarding any more.

IV. NUMERICAL RESULTS

We use NS-2 to implement our bandwidth reservation protocol for multirate anypath in WMNs — BREW, and compared its performance with BOR/AC. In this section, we present our

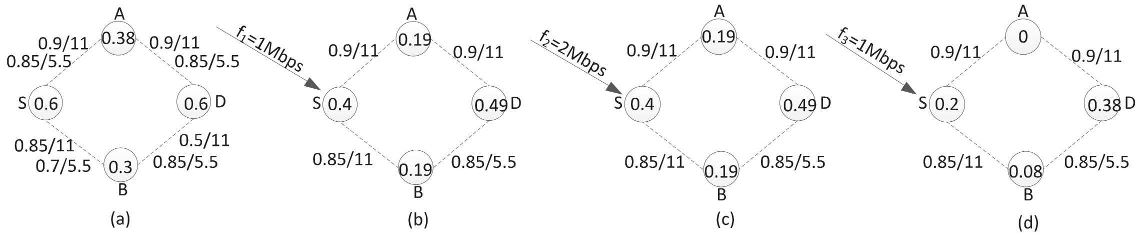

Fig. 1. An illustration for using BREW. The values in the circles are the current idle time of the nodes. The pairs of delivery probability and transmission rate are shown beside virtual links. Here, we assume that there are two transmission rates: 5.5Mbps and 11Mbps. (a) shows initial status. (b) shows that a data flow f_1 with 1Mbps bandwidth requirement is accepted and nodes' idle time is updated. For ease of illustration, we let the bandwidth be satisfied by a single (sub-)flow, and use a delivery probability / transmission rate pair to indicate the routing decision on each virtual link. (c) shows that a data flow f_2 with 2Mbps bandwidth requirement is rejected because of not sufficient bandwidth. (d) shows that a data flow f_3 with 1Mbps bandwidth requirement is accepted.

numerical results. Our comparison results show that BREW outperforms BOR/AC in terms of flow acceptance ratio and system-wide throughput.

A. Methodology

We simulate BREW and BOR/AC with the following setup:

- Hardware: Simulations run on a computer with a 3.1GHz CPU and 4GB memory.
- Transmission Rates: Each node can work at the following four transmission rates: 1Mbps, 2Mbps, 5.5Mbps, and 11Mbps.
- Carrier Sensing Range: Each node uses CSMA/CA without RTS/CTS as the MAC layer protocol and has different carrier sensing ranges with different rates.
- Terrain and Nodes: 20-60 nodes are randomly placed over a square of $1000m \times 1000m$. 22 data flows are generated with randomly selecting source and destination, and the required bandwidth of data flow is randomly selected from 0 to 0.5Mbps. The packet size is 1500 bytes.

B. Results

Figure 2 shows the average flow acceptance ratios of BREW and BOR/AC. Since BOR/AC only works with single transmission rate, we separate the cases for BOR/AC with different transmission rates, including 1Mbps, 2Mbps, 5.5Mbps, and 11Mbps with standard deviations. We can see that the average flow acceptance ratios of BREW are always higher than those of BOR/AC. The results show that the acceptance ratio of BREW is up to 5.6, 3.6, 1.34, and 2.45 times that of BOR/AC with transmission rate 1Mbps, 2Mbps, 5.5Mbps, and 11Mbps, respectively.

Figure 3 shows the system-wide throughput of BREW and BOR/AC with different transmission rates. The results show that the system-wide throughputs of BREW are always higher than those of BOR/AC. The system-wide throughput of BREW is up to 3.8, 2.29, 1.1, and 3.1 times that of BOR/AC with transmission rate 1Mbps, 2Mbps, 5.5Mbps, and 11Mbps, respectively.

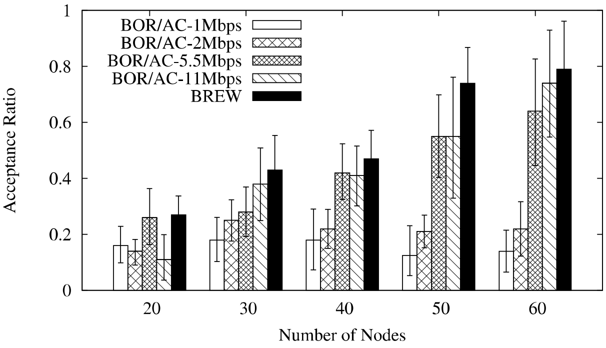

Fig. 2. Average flow acceptance ratios with standard deviations for different number of nodes.

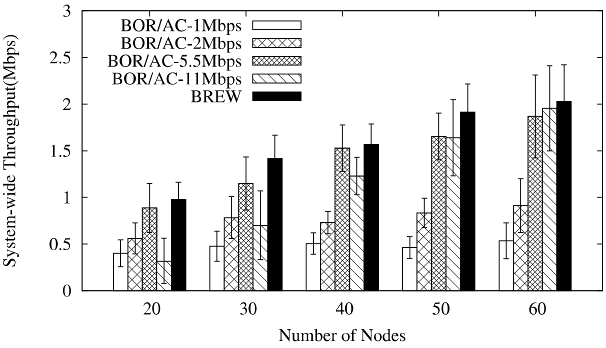

Fig. 3. The system-wide throughput with standard deviations for different number of nodes.

From Figure 2 and Figure 3, we can see that while the performance of BOR/AC is greatly affected by the density of the nodes and the selected transmission rate, BREW achieves high performance in different environments. Specifically, in the case 20 nodes, the average flow acceptance ratio and the system-wide throughput of BOR/AC with 11Mbps are lower than those of BOR/AC with 1Mbps, 2Mbps, and 5.5Mbps. This is because many nodes cannot reach each other at 11Mbps due to the poor link quality, and thus the network is disconnected. When there are sufficient number of nodes, i.e., 60 nodes, BOR/AC performs best with a transmission rate of 11Mbps. In contrast, BREW always achieves the best performance in different scenarios.

We note that in the case of 30 nodes, the average flow acceptance ratio of BOR/AC with 11Mbps is higher than BOR/AC with 5.5Mbps, while the system-wide throughput is not. The reason is that the bandwidth requirement of flow

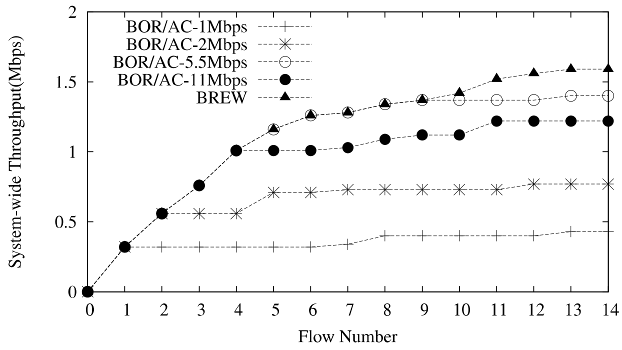

Fig. 4. The system-wide throughput with a sequence of random data flows in the scenario with 40 nodes.

is randomly selected, BOR/AC with 5.5Mbps accepts more larger flows than BOR/AC with 11Mbps, and this impacts the average flow acceptance ratio of BOR/AC with 5.5Mbps.

Figure 4 illustrates the system-wide throughput of BREW and BOR/AC with 22 flows coming in a fixed order in the scenario with 40 nodes. We do not show the results for flow 15-22, because these flows are rejected by all the five evaluated bandwidth reservation protocols. From the results, we can see that BREW always performs at least as good as BOR/AC. Compared with BOR/AC with different transmission rates, BREW achieves a significant increment for system-wide throughput.

V. RELATED WORKS

Traditional QoS protocols just use single path, but opportunistic routing presented these years can use set of next hops to transmit packets. Biswas and Morris [5] designed and implemented ExOR, which is an opportunistic routing protocol relaying on MAC layer. Chachulski et al. presented MORE [6], which is a routing protocol using both opportunistic routing and network coding to increase the end-to-end throughput. Koutsonikolas et al. presented CCACK [7], which exploits a novel acknowledgment scheme to allow nodes to acknowledge to their upstream nodes in a simple way. Han et al. presented O3 [9], which is a novel OR protocol that combines with inter-flow coding and rate limiting to improve the end-to-end throughput. Wang et al. presented CORMAN [8], which is an OR protocol extended to ExOR and works in mobile ad hoc networks. These protocols just use single rate. Laufer et al. presented multirate anypath routing [11], and in this protocol, each node uses both a set of next hops and selected transmission rate to reach a destination. Unlike opportinistic routing, Jiming et al. presented a fully asynchronous distributed algorithm based on dual decomposition to achieve the maximum utility [14].

Several QoS protocols have been proposed for wireless mesh networks. Some of them use TDMA to allocate the slots of time for the transmission of nodes [15]. Some of them just use CSMA-based model and are independent of the MAC layer. For example, INSIGNIA [2] is an in-band signaling protocol designed for mobile ad hoc wireless networks. It uses control signals and soft-state resource management approach to support QoS. Among these works, Guimaraes et al. [16] and Zhao et al. [10] are close to our work. [16] presents a QoS reservation mechanism for multirate ad hoc wireless networks that allows bandwidth allocation on a per flow basis, but it doesn't work with anypath routing protocols. [10] provides bandwidth guarantees with anypath routing protocols but it uses single rate which limits the throughput improvement of BOR/AC.

VI. CONCLUSION

In this paper, we have presented BREW, which is a novel bandwidth reservation protocol for multirate anypath routing in WMNs. BREW exploits the broadcast nature of the wireless communication media, and integrates bandwidth from multiple anypath routes to satisfy the bandwidth requirements from data flows. Through extensive evaluations, we have shown that BREW can achieve higher system-wide throughputs and flow acceptance ratios than those of the closely related work BOR/AC.

REFERENCES

[1] P. Bahl, "Opportunities and challenges of community mesh networking," in *Keynote Presentation @ MICS Workshop in ETH Zurich*, 2004.

[2] S. Lee, G. Ahn, and A. T. Campbell, "Improving UDP and TCP Performance in Mobile Ad Hoc Networks with INSIGNIA," *IEEE Communications Magazine*, vol. 39, pp. 156–165, 2001.

[3] G. Ahn, A. T. Campbell, A. Veres, and L. Sun, "SWAN: Service Differentiation in Stateless Wireless Ad Hoc Networks," in *INFOCOM*, 2002.

[4] R. Sivakumar, P. Sinha, and V. Bharghavan, "CEDAR: a Core-Extraction Distributed Ad Hoc Routing Algorithm," *IEEE Journal on Selected Areas in Communications*, vol. 17, pp. 1454–1465, 1999.

[5] S. Biswas and R. Morris, "ExOR: Opportunistic Multi-Hop Routing for Wireless Networks," in *SIGCOMM*, 2005.

[6] S. Chachulski, M. Jennings, S. Katti, and D. Katabi, "Trading Structure for Randomness in Wireless Opportunistic Routing," in *SIGCOMM*, 2007.

[7] D. Koutsonikolas, C. Wang, and Y. C. Hu, "CCACK: Efficient Network Coding Based Opportunistic Routing Through Cumulative Coded Acknowledgments," in *INFOCOM*, 2010.

[8] Z. Wang, Y. Chen, and C. Li, "CORMAN: A Novel Cooperative Opportunistic Routing Scheme in Mobile Ad Hoc Networks," *IEEE Journal on Selected Areas in Communications*, vol. 30, pp. 289–296, 2012.

[9] M. K. Han, A. Bhartia, L. Qiu, and E. Rozner, "O3: Optimized Overlay-based Opportunistic Routing," in *MobiHoc*, 2011.

[10] P. Zhao, X. Yang, J. Wang, B. Liu, and J. Wang, "BOR/AC: Bandwidth-aware Opportunistic Routing with Admission Control in Wireless Mesh Networks," in *INFOCOM*, 2012.

[11] R. Laufer, H. Dubois-Ferriere, and L. Kleinrock, "Multirate Anypath Routing in Wireless Mesh Networks," in *INFOCOM*, 2009.

[12] A. Miu, H. Balakrishnan, and C. E. Koksal, "Improving Loss Resilience with Multi-Radio Diversity in Wireless Networks," in *MobiCom*, 2005.

[13] C. Reis, R. Mahajan, M. Rodrig, D. Wetherall, and J. Zahorjan, "Measurement-Based Models of Delivery and Interference in Static Wireless Networks," in *SIGCOMM*, 2006.

[14] J. Chen, W. Xu, S. He, Y. Sun, T. P, and X. Shen, "Utility-based asynchronous flow control algorithm for wireless sensor networks," *IEEE Journal on Selected Areas in Communications*, vol. 28, pp. 1116–1126, 2010.

[15] W. Liao, Y. Tseng, and K. Shih, "A TDMA-based Bandwidth Reservation Protocol for QoS Routing in a Wireless Mobile Ad Hoc Network," in *ICC*, 2002.

[16] R. Guimaraes, L. Cerda, J. M. Barcelo, J. Garcia, M. Voorhaen, and C. Blondia, "Quality of service through bandwidth reservation on multirate ad hoc wireless networks," *the Journal of Ad Hoc Networks*, vol. 7, pp. 388–400, 2009.