A Practical Approach for Providing QoS in Multichannel Ad-Hoc Networks using Spectrum Width Adaptation

Vijay Raman
Dept. of Electrical and Computer Engineering
University of Illinois at Urbana-Champaign
Urbana, IL 61801
Email: vraman3@illinois.edu

Matthew C. Caesar
Dept. of Computer Science
University of Illinois at Urbana-Champaign
Urbana, IL 61801
Email: caesar@cs.illinois.edu

Abstract—**Multichannel wireless networks provide the flexibility to utilize the available spectrum efficiently for achieving improved system performance in terms of throughput and spectral efficiency. However, there has been no practical means for provisioning quality of service (QoS) in multichannel wireless networks. While previous proposals providing signaling and adaptation mechanisms for QoS, they support only fixed-width channels, restricting system performance in networks supporting variable-width channels. In this paper, we propose a distributed mechanism for provisioning QoS by adapting the channel widths in a multichannel, ad-hoc network. Our algorithm builds upon the well-known ETT routing metric to incorporate bandwidth adaptability. We also propose mechanisms for performing admission control and congestion control jointly in a multihop setting. We demonstrate the performance of our algorithm using a modified AODV routing protocol through extensive simulations. Our simulations results show that our proposed approach can achieve up to twice the spectral efficiency and data rates when compared to the greedy approach. Furthermore, our results show that our proposed approach scales well as the network density increases.**

I. INTRODUCTION

Multichannel wireless networks provide more efficient use of spectrum by allowing multiple concurrent transmissions between nodes that share contention regions. Multichannel wireless nodes may also receive and transmit simultaneously when multiple radios are used, improving throughput and reducing transmission delay. The capacity of a multichannel wireless network with n randomly distributed nodes scales linearly with the number of channels when the ratio of the number of channels to the number of interfaces is of the order of $O(\log n)$ [1]. Hence for practical networks, throughput is maximized by using as many channels as possible, limited in practice by the wireless technology used.

At the same time, demands for bandwidth and high service quality have continued to increase for next-generation applications of wireless networks. Emergency and disaster response applications require the ability to seamlessly migrate yet maintain guaranteed communications with coordination points [2]. Media applicatoins such as VoIP, video, and live streaming traffic require larger amounts of bandwidth, potentially with varying bit rates. To reduce costs, best-effort flows that are bursty in nature with minimal bandwidth requirements (such as web browsing and file transfer applications) are routed over the same infrastructure. Due to the different bandwidth and delay requirements across different kinds of traffic, it is necessary to provide *quality of service* (QoS) to the applications for improving system performance and minimizing wastage of resources.

Several approaches have been proposed in the literature for provisioning QoS in multichannel wireless networks [3], [4], [5] (which we briefly overview in Section II). Most of these approaches require that system resources, such as bandwidth and data rates, be adaptively varied depending on the application requirements. However, this is practically difficult with the current generation of wireless networks. This is because, most existing multichannel wireless network implementations [6], [7], [8] and standards [9] propose to use fixed width physical channels. For instance, IEEE 802.11a devices use a fixed channel width of 20 MHz. This therefore restricts the extent to which the physical resources can be adapted.

More recently, wireless networks that support channel width adaptation are beginning to be built. These networks are composed of radios that allow wireless nodes to control the size of the frequency band used for communication. The added flexibility provided by such networks gives an opportunity to more efficiently utilize wireless spectrum, but also complicates the spectrum allocation problem, by introducing a wider range of possible solutions. Recent work has leveraged channel width adaptation for the purposes of load balancing [10], [11], interference minimization [12], and efficient spectrum utilization [13]. In this paper, we extend the idea of channel width adaptation for proposing a practical means for provisioning end-to-end QoS in multichannel wireless networks. In our approach, the physical channel widths are chosen dynamically on a per-flow basis at every hop. For this purpose, we introduce a new routing metric that is based on the well known expected transmission time (ETT) metric [14]. Additionally, we propose techniques for performing admission and congestion control in the network. We evaluate the performance of our algorithm using a modified AODV routing protocol [15] through extensive simulations.

The remainder of the paper is organized as follows. In Section II we discuss some relevant related work. We discuss the system model and problem statement in Section III.

Our proposed algorithm is discussed in Section IV and the simulation results along with the assumptions are discussed in Section V. Finally, we conclude the paper in Section VI.

II. RELATED WORK

There have been several works in the literature on dynamic spectrum allocation in wireless networks. However, most of this work focuses on dividing a wide band of spectrum amongst wireless devices depending on certain traffic-dependent parameters. For instance, in [16], the authors discuss a spectrum adaptation mechanism for a composite wireless network that comprises both 802.11b and cellular networks based on the temporal and spatial usage patterns of the devices in the network. Furthermore, most of the earlier works on cellular communication systems focus on bandwidth or time-slot adaptability. Most currently deployed cellular networks are capable of providing packetized voice, but are not capable of providing QoS to the applications. In our work, we propose a scheme for adapting the channel widths of an already channelized system (such as 802.11a/b/g networks) with the goal of providing QoS.

The notion of bandwidth adaptability in wireless networks has been recently researched in [10] and [11]. In this paper, the authors have demonstrated bandwidth adaptability in 802.11a/b wireless networks using emulations on a FPGA-based wireless emulator and few initial experiments. The authors show that narrow bandwidth transmissions can have a greater communication range and experience reduced interference when compared to wide-channel transmissions. On the other hand, they also show that wide-channel communication can achieve higher data rates and increase the overall spectrum utilization in the network. We propose to use the inherent tradeoff involved between narrow and wide channel communication for differentiating the traffic flows with the goal to provide QoS depending on the application needs.

To the best of our knowledge, most of the previous work on QoS provisioning for wireless networks has focused on fixed channel width. In [3], Tang et al. have proposed separate optimization problems for channel allocation and QoS routing for multi-channel, fixed-width wireless networks. They assume all wireless nodes to be stationary. In [4], Xu et al. have presented a QoS framework over landmark routing (LANMAR) with a single fixed width channel per node. They propose a probing-free call admission control (CAC) mechanism and thus claim lower admission delays. However, the cost of maintaining large routing state on every landmark node increases overhead. In particular, if a node has to send data to a node in another landmark area, and the requested data-rate for the flow is between the highest and lowest available data-rate within that landmark, then QoS-LANMAR resorts to probing. If such a situation occurs frequently, QoS-LANMAR resorts to a probing-based mechanism, which can harm performance. In addition, the performance of their proposed approach also largely depends on the availability of mobile backbone networks (MBNs). In [17], the authors discuss a link-state approach coupled with a core node set extraction. The authors propose to use localized link-state exchanges. Our approach, however, does not require any link state maintenance.

In [18], Perkins et al. have presented QoS extension for the Ad Hoc On demand Distance Vector (AODV) routing protocol [15]. *While this work focuses primarily on signalling and path setup, we extend this work by developing algorithms to allocate spectrum widths to achieve QoS goals.* The choice of a reactive routing protocol helps in avoiding the need for a separate CAC mechanism. Furthermore, we implement our algorithm along with the route discovery part of the AODV routing protocol, which provides the benefit of requiring less routing state. In [5], Liao et al. have attempted to provide a probe-ticket based approach for provisioning QoS. The unique aspect of their route discovery mechanism is what they call a ticket-splitting approach. The authors maintain a notion of a ticket for every node, which is divided into multiple sub-tickets. QoS provisioning is achieved by allocating bandwidth to every sub-ticket on every intermediate node during route-discovery. This approach may present an interesting solution in the absence of a contiguous spectrum at certain nodes.

III. SYSTEM MODEL AND PROBLEM FORMULATION

In this section, we provide a brief overview of the network model and the multichannel protocol assumed by our algorithm. We then formally state the problem that is addressed in this paper.

A. Network Model

The network consists of a set of n static wireless nodes that are distributed in an ad-hoc fashion. Each of the wireless nodes are equipped with m radios (or wireless interfaces), of which m_r radios are used for receiving data from other nodes and m_t radios are used for transmitting data. The receive radios are allocated channels based on a minimum interference channel allocation mechanism, which is modified form of the local balancing channel allocation algorithm proposed in [19]. An overview of this algorithm along with the modifications is described in Section IV. The channel allocated to the receive radios is fixed for durations that are larger than a packet transmission time. For this reason, we refer to these radios as "fixed" radios and the channel allocated to these radios as the "receive" channel. The transmit radios, on the other hand, are not allocated any channel to start with. Instead, the channel on which they transmit is decided dynamically based on a multichannel protocol, described in the next subsection. We therefore refer to these radios as 'switchable' radios. We make sure that no two radios (both transmit and receive) within a node operate simultaneously on the same channel.

1) Multichannel protocol: The multichannel protocol used in our model was originally proposed in [6] and [20], and we provide a brief overview here for clarity. Whenever a packet has to be transmitted on a multihop path, one of the radios in a node switches its channel to the receive channel of an intended neighboring node (depending on the routing table entry) and transmits the packet on that channel. For instance, consider the example shown in Figure 1. In this example, we show

a multihop communication from node A to node C through node B. Here, the switchable radio in node A switches to the receive channel of node B and transmits the data. Later, node B switches its switchable radio to the receive channel of node C and forwards the data.
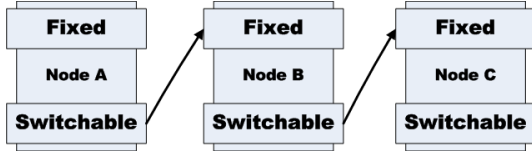

Fig. 1.   Example multi-channel protocol operation

The multichannel protocol operation described above requires that every node be aware of the receive channels of their neighboring nodes. This is made possible by the exchange of a broadcast `hello` message that contains the receive channel information. Every node periodically sends out a `hello` message on all the channels so that all its neighbors that may be listening on any of the channels may receive the `hello` message. For the purposes of efficient channel allocation, the `hello` messages are propagated over two hops. This allows every node to be aware of the channel information of all the neighbors that are up to two hops away. The term "hop" is defined as follows: If two nodes can have a direct communication link between them, then they are said to be within one hop from each other. If a transmission from one node to another requires h one hop transmissions, then the nodes are said to be h hops away from each other. More details on the multichannel protocol can be found in [6] and [20].

B. Problem Statement

The problem formulation in this section is based on the one proposed in [15]. However, we make some valid assumptions with respect to the network topology, which is not considered in [15]. We model the network as a graph $G(V, E)$, where V is the set of n vertices and E is the set of edges between the vertices. An edge exists between two vertices u and v, if the two nodes are within one hop from each other. Furthermore, we assume that there can be only one edge between any two vertices. In other words, we assume that at most one switchable interface in a node can switch to at most one of the receive channels of the other node. This is different from the multi-graph assumption in [15]. Let $\mathcal{C} = \{c_1, c_2, \ldots c_k\}$ be the set of k channels available for allocation to the m_f fixed interfaces in each of the n nodes in the network. The number of channels $l > m_f$ and we assume that the channel allocation precludes any interference between the neighboring links in the network. Each of the channels are capable of operating on one of the l bandwidths chosen from the set $\mathcal{W} = \{w_1, w_2, \ldots w_l\}$. Due to the hardware restrictions, we assume that the bandwidth values are chosen from a discrete set. Additionally, we assume that the bandwidth values are in increasing order. In other words $w_1 < w_2 < \ldots w_l$. If $\mathcal{A}$ is a

channel allocation that assigns m_f different channels to fixed interfaces in the nodes, then $\mathcal{A} : \mathcal{C} \to \mathcal{V}$.

Let $L(e)$ be the current load on the edge $e \in E$, shared among some q ongoing flows. Let the q flows be a combination of q_{ef} elastic flows and q_{sf} streaming flows. The remaining bandwidth in the link e, $R(e)$ is given by $w_l - L(e)$. Let f be a new incoming flow that has to be routed from a node s to t, $(s, t \in V)$ with some rate requirement. Let the relevant minimum bandwidth required to service this flow be $b_f \in \mathcal{W}$. Our problem is to find a path from s to t that can satisfy the rate requirements for the flow f and maximize the end-to-end rate. In other words, if $\mathcal{P}$ is the set of all paths from s to t and $D(e)$ is the data rate achieved in link e, then our objective is to find a route r such that, $R(e') \geq b_f, \ \forall \{e' \in r : r \in \mathcal{P}\}$ and $\max_{r \in \mathcal{P}} \left\{ \min_{e' \in r} D(e') \right\}$. While the condition on $R(e')$ defines the admission control criteria, the route selection is based on choosing a path that maximizes the minimum rate of the links in the route. This in turn maximizes the end-to-end rate of the route selected.

In the following section, we first provide a brief overview of the channel allocation algorithm used. Later, we discuss our proposed algorithm for choosing the channel widths and the routes for satisfying the QoS requirements of the flows in the network. We then propose a modification to the AODV routing protocol based on the proposed routing metric, which we use in our simulations.

IV. PROPOSED APPROACH

In this section we described our proposed approach. First, we allocate center frequencies across the network by leveraging an existing channel allocation algorithm (Section IV-A). Next, we use admission control to determine which flows to admit into the system based on the availability of channel resources (Section IV-B). We then define a routing metric which takes into account the delay of switching channels and channel width, the set of elastic flows that must be dropped to admit the new flow, and a demand factor used to balance congestion across the network (Section IV-C). Finally, we describe an AODV-based protocol used to set up paths and perform routing (Section IV-D).

A. Channel Allocation Algorithm

Before proceeding to a discussion on the channel allocation algorithm, we first describe the current approach to channelization in existing multichannel technologies, such as IEEE 802.11a and 802.11b. The channels in these technologies have fixed widths and the neighboring channels overlap with each other (the amount of overlap, however, depends on the actual technology). In our problem, if we adapt the channel widths, the amount of overlap may increase or decrease. A decrease in bandwidth may in fact be beneficial (since a corresponding decrease in interference is simultaneously obtained [12]). Similarly, an increase in channel overlap may increase the interference in the network. For simplicity, we only use a subset $C_s \subset C$ of non-neighboring channels for allocating

spectrum to nodes, and assume initial channel widths that are specified in existing technologies (e.g., 20 MHz in 802.11a). When increasing the channel width, we combine the channel that was initially chosen with a neighboring channel that was not chosen initially. For instance, in 802.11a there are 12 channels (namely 36, 40, 44, 48, 52, 56, 60, 64, 149, 153, 157, and 161). Let us assume that we choose the channels 36, 48, 60, 149, and 161 for the initial allocation. All of these channels are allocated with the standard width of 20 MHz. If the width of channel 48 has to be increased to 40 MHz, the we combine the channel 48 with (upper) 5 MHz of channel 44 and (lower) 5 MHz of channel 52, thereby totaling 5 MHz. In case of edge channels, such as 36, 149, and 161, we simple combine them with their neighboring channel (36↔48, 149↔153, and 157↔161) for increasing their widths to 40 MHz. This way, we use the entire spectrum of channels.

We use the algorithm described in [19] to allocate center frequencies, and we provide a brief overview here. According to this algorithm, every fixed interface in a node initially starts up on one of the subset of channels chosen for allocation. Later, they exchange their chosen channels with their one and two hop neighbors using broadcast `hello` messages. Each node then counts the number of one and two hop neighbors that are assigned a particular channel and calculates the average and the minimum utilization (number of nodes) of each of the channels. A node then probabilistically decides to switch its channel if the load on the current channel is above the average and the minimum loads over all the channels. The current channel is then switched to a channel that has the minimum load. The probability with which a node switches its channel is chosen appropriately to ensure convergence of the algorithm.

B. Admission Control Mechanism

The proposed admission control mechanism is executed at a node to determine whether or not the bandwidth requirement of an incoming flow can be satisfied on any of the links. Accordingly, a node first obtains the information on the bandwidth currently used in a link by all the existing flows along with the number of elastic and streaming flows currently serviced on the link. It then computes the bandwidth available for a the new incoming flow. If the available bandwidth can satisfy the requirement of the incoming flow, then the new flow is admitted. If not, the node drops elastic flows successively until the available bandwidth becomes sufficient for the new flow. If there is no sufficient bandwidth even if all the elastic flows were dropped, the incoming flow is rejected. Note that a node may not perform admission control on all of its links. Instead, the set of links on which a node performs admission control depends on the actual routing protocol used. Furthermore, as mentioned in Section III, the bandwidth requirements are chosen from a discrete set of values. The pseudo-code for this mechanism is as follows:

At a node with an incoming streaming flow,
1. *Check available bandwidth*

2. *if (available bandwidth ≤ required bandwidth AND {available bandwidth + elastic bandwidth} ≥ required bandwidth)*
    Repeat until all elastic flows are dropped
      Drop an elastic flow
      If available bandwidth ≥ required bandwidth
        Admit the streaming flow
        Exit
    end repeat
3. *Reject the streaming flow*

Admitting a flow at a node by itself may not guarantee that the flow will be eventually sent through this node. The decision on whether or not a node is chosen for forwarding a flow depends on the routing protocol used. We clarify more on this in Section IV-D.

C. Routing Metric

The routing metric used by out approach is a variant of the multichannel routing metric (MCR) [19] and the expected transmission time (ETT) metric [14]. The ETT is specified per link and is given by, $ETT = ETX * \frac{S}{D}$, where ETX [21] is the expected number of transmission attempts (including retransmissions) required for transmitting a packet, S is the average packet size, and D is the data rate of the link. Several mechanisms for measuring the ETX has been proposed in the literature [21], [19] and all the mechanisms involve computing the loss probability associated with the forward and the reverse direction of a link. The data rate D of the link depends on the channel width chosen on that link. The proposed routing metric combines the ETT metric with the hardware delay involved in switching the channel (to one of the neighboring channels for transmission), namely C_{sw} and the delay involved in adapting the channel width, namely C_{bw}. Additionally, we associate a penalty for the number of elastic flows that need to be dropped for admitting a flow, namely C_{drop} and another penalty metric that captures the number of ongoing streaming flows in the network, namely *demand factor* C_{demand}. The demand factor is used for the purpose of congestion control. We postpone the discussion on this until we discuss the routing protocol. If c_i is the channel used in the i-th hop of route and w_i is the corresponding channel width used, the end-to-end routing metric for a path involving h hops, namely QOSAR (QoS-based ad-hoc routing metric) is given by,

$$QOSAR = \sum_{i=1}^{h} [ETT(i) + C_{sw}(c_i) + C_{bw}(w_i)$$
$$+ \{C_{drop}(i) + C_{demand}(i)\}]$$

In their definition of the MCR metric, the authors of [19] introduce a factor for the interference in the network. We ignore that factor in or definition for simplicity. However, if desired, the interference factor can be easily integrated to our metric without affecting the performance of the metric.

1) Demand factor and Congestion Control Mechanism:
We will now explain how the demand factor in the $QOSAR$ metric is used for congestion control. Consider that a certain node, say node A, receives a RREP with a bandwidth requirement b_{f1} for a flow f_1. Let the available bandwidth at node A be b_a and let $b_{f1} < b_a$. Node A will therefore broadcast the RREP message along with bandwidth b_a and other costs. Before this response propagates to the source of this flow, let us assume that another request arrives at A from a different source requiring a bandwidth of $b_{f2} < b_a$ for a flow f_2. In this case, node A can either choose not to rebroadcast the RREP as it has already responded with its available bandwidth for the flow f_1 or it can respond with a bandwidth that is smaller than b_a. However, if the first flow, f_1 did not choose the route via A, then both of the above actions may result in a route denial for the flow f_2. We therefore, propose to attach a demand factor, C_{demand} to the routing responses, which is simply the number of routing requests at a node. Therefore, in the example above node A responds with the same bandwidth b_a for the flow f_2, but includes a C_{demand} value that is incremented by one to account for the flow f_1. Because a node having a high demand factor can be a potential bottleneck congestion node, including the demand factor information proactively in the routing metric can help reduce, to some extent, the chance of congestion in the network.

D. Modified AODV Routing Protocol

We assume that every incoming flow has a certain rate guarantee that is known to the routing protocol. Furthermore, the protocol is aware of the relevant bandwidth (or channel width) necessary for satisfying the rate requirement. In this section, we only present the modifications to the AODV routing protocol. The actual details of the AODV routing protocol can be found at [15]. Whenever a a new streaming flow has to routed through the network, the following steps are performed,

1) The source node sends a route request (RREQ) message including the required bandwidth for servicing the flow.
2) Intermediate nodes broadcast a RREQ if no route to destination is known.
3) If an intermediate node has a route to the destination or it is the destination node itself (if the RREQ reaches the destination node) reply with a route reply (RREP), including the required bandwidth extracted from the RREQ.
4) Upon receiving a RREP packet, an intermediate node performs admission control as discussed in Section IV-B.
5) If the flow is rejected during admission control, then the node does not broadcast the RREP any more.
6) If the flow is not rejected, then the following are performed:
   - The node includes the available bandwidth for this flow in the RREP message.
   - It also includes the cost for switching a channel, C_{sw} and the width, C_{bw} (if any), and the penalty

values C_{drop} and C_{demand} in the RREP message and broadcasts the RREP message again.
   - A reverse pointer is set to the node from which it received the RREP, similar to th original AODV routing protocol.
7) Upon receiving one or more RREP messages, the source node decides on the best possible route based on the route metric described earlier.

V. SIMULATION RESULTS

In this section we start by giving an overview of our simulation methodology in Section V-A, and then describe performance results in Section V-B.

A. Simulation Model

In our event-driven simulations, we place 2500 nodes distributed randomly on a 50×50 area. Every node is equipped with two radios of which one is used for transmitting data and the other is used for receiving data. For modeling the link layer we have used the actual traces from [22], which is based on real world experiments. This model defines the network topology by associating with every link a probability value, which decides whether a transmission on that particular link will be successful or not. The probability values are defined as a function of distance and transmission power. In Figure 2, we have replicated the probability of successful packet reception for two of the power values that are defined in [22], namely medium and low power (corresponding to a potentiometer setting of 66 and 69, respectively). The plot shows that the probability of successful packet reception decreases as the distance from the packet source increases. Furthermore, the plot shows that the probability of successful reception of a packet improves as we increase the transmit power.

The flow arrivals to the network are assumed to be Poisson with a rate λ arrivals per second. The value of λ is varied from 0.02 to 0.1 in steps of 0.2, and further from 0.2 to 1.0 in steps of 0.2. This ensures that our simulations are compared across a wide variety of arrival intensities (offered load). Furthermore, we assume that all the new arrivals are streaming flows requiring a certain QoS provisioning. Each of these flows are assumed to have a provided bandwidth requirement, which is chosen uniformly at random from a set of five bandwidths, namely $\{5MHz, 10MHz, 20MHz, 40MHz\}$. We assume an IEEE 802.11a network and therefore there are twelve channels to choose from, each of which can be tuned on any of the five bandwidths listed previously. The nodes are allocated channels before the start of the simulation. We assume that the simulated nodes already have a set of elastic and streaming flows distributed uniformly at random, and therefore the nodes are termed backlogged even before the new arrivals. The flows that already exist in the nodes are assumed to be a bandwidth chosen uniformly at random from the set of five bandwidths listed earlier. The available bandwidths at each node are adjusted based on the bandwidths used by the flows that are already in service.
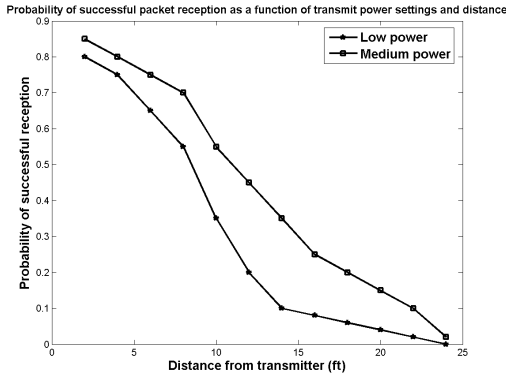
Fig. 2. Probability of successful packet reception as a function of distance and power
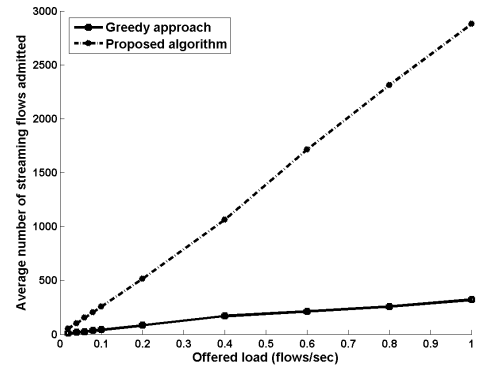

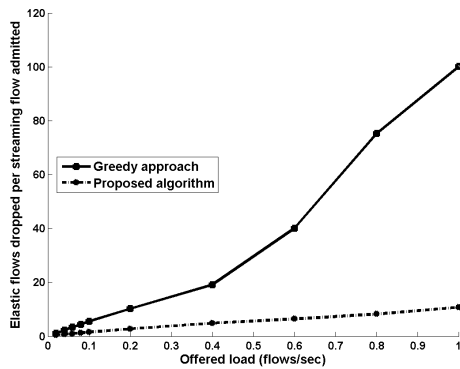Fig. 3. Overall network utility in terms of number of admitted flows

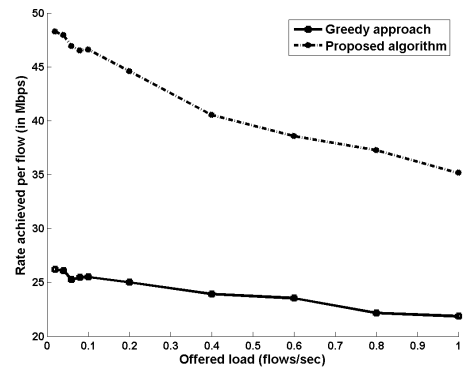Fig. 4. Total number of dropped flows in the network

Fig. 5. Average rate achieved per flow

We assume that the network, traffic, and propagation characteristics are static for the period of the simulation. For each of the offered loads, we run 100 randomly generated networks with the given parameters. The performance results presented are averaged across all the flows and all the 100 network realizations. For each of the runs, we compare the performance of our proposed approach based on the modified AODV protocol using the $QOSAR$ metric and a greedy approach, which is described as follows: The greedy algorithm is a naive approach, in which every hop chooses a neighbor with the minimum ETT and switching cost as the next hop. In other words, instead of choosing the route based on the summation of the ETTs and other costs (described in Section IV) obtained from all the nodes between a source and a destination, the next hop nodes are naively chosen based on just the ETT values and switching cost known locally.

B. Performance Results

We now discuss the various performance results obtained through simulations. We first plot the overall network utilization in terms of number of admitted flows in the network. We measure this by measuring the total number of dropped flows (both elastic and streaming flows) that were rejected due to insufficient bandwidth at a node, and deducting it from
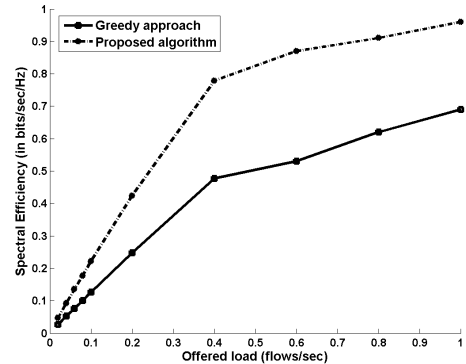

Fig. 6. Per-flow spectral efficiency achieved

the total number of incoming flows. Figure 3 compares the network utilization for the greedy and the proposed algorithms. We observe that the difference in the network utilization between our proposed approach and the greedy approach increases drastically as the offered load increases. This shows that fewer flows are admitted in the network in the greedy approach. Furthermore, we can also observe that by adapting the bandwidth based on the flow requirements, we can sustain

more flows in the network irrespective of the offered load. We next plot the average number of elastic flows dropped for every streaming flow admitted in to the network in Figure 4. We can readily observe that the average number of dropped flows in the case of greedy algorithm is higher than that in our proposed algorithm. Furthermore, we observe that the average number of dropped flows increases almost exponentially in the case of the greedy approach as the offered load increases.

Next, we compare the average transmission rate achieved per flow in the network. Figure 5 shows the plots for the rates achieved per flow for the two algorithms. We can observe from that plot that our proposed algorithm can achieve almost 50% higher rate than the greedy algorithm approach. In particular, the rate achieved per flow at an offered load of 0.1 is around 25.2 Mbps in the case of greedy approach, while it is 46.2 Mbps for our proposed approach. Even at the highest offered load of 1.0 that is compared in our simulation, we observe that our proposed approach can achieve up to 35.3 Mbps, while the greedy approach can achieve just 28 Mbps. Next, we compare the spectral efficiency, which is the ratio of rates achieved to the available bandwidth in the network, for the two algorithms. Figure 6 shows the corresponding plots. We first observe that the spectral efficiency increases as the offered load increases for both the algorithms, due to the increased network utilization. Additionally, we observe that the difference in spectral efficiency between our proposed approach and the greedy approach increases as the offered load increases. Our proposed approach achieves a spectral efficiency of up to 0.95 (at an offered load of 0.1) while the greedy approach can achieve an efficiency of 0.65 at the same offered load.

Next, we wish to compare our approach with the greedy approach as the network density varies. For this purpose, we fix the network size as a 50×50 square as before and the offered load to be 0.1. However, we vary the number of nodes in the network from 1000 to 5000. First, we observe from Figure 7 that the network utilization increases with the number of nodes for both the approaches. Additionally, as before, the network utilization in the case of our proposed approach increases drastically with the number of nodes when compared to the greedy approach. Figure 8 shows the plots for the average number of elastic flows dropped per streaming flow admitted in the network. We observe that the number of flows dropped in the case of our proposed approach is much less than that in the case of the greedy approach. The number of flows dropped, however, does not increase drastically as the network density increases. Next, we plot the rates achieved by the two approaches in Figure 9 for the various network densities. As before, our approach achieves almost twice the rate achieved by the greedy approach. Furthermore, the rate achieved remains almost constant with the network density implying that our proposed approach scales well for dense networks. Finally, the spectral efficiency plots in Figure 10 shows that our approach is much more efficient than the greedy approach and the efficiency increases with the network density.

VI. CONCLUSION

In this paper, we discussed a method for provisioning QoS in a multichannel wireless network by adapting spectrum widths of wireless channels. We proposed a joint algorithm that performs routing, admission control, and congestion control with the goal to maximize the overall network utilization and the spectral efficiency. Our algorithm uses a modified AODV routing protocol to coordinate allocation decisions across wireless nodes. Using simulations, we evaluated the performance of our algorithm by comparing it with a greedy algorithm where nodes route traffic based on only the local information. Our results show significantly improved network utilization and spectral efficiency when compared to the greedy algorithm.

REFERENCES

[1] P. Kyasanur and N. Vaidya, "Capacity of multichannel wireless networks: Impact of number of channels and interfaces," *Mobicom*, August 2005.

[2] B. Braunstein, T. Trimble, R. Mishra, B. Manoj, R. Rao, and L. Lenert, "Feasibility of using distributed wireless mesh networks for medical emergency response," *AMIA Symposium*, 2006.

[3] J. Tang, G. Xue, and W. Zhang, "Interference-aware topology control and qos routing in multi-channel wireless mesh networks," *MobiHoc*, May 2005.

[4] K. Xu, K. Tang, R. Bagrodia, M. Gerla, and M. Bereschinsky, "Adaptive bandwidth management and qos provisioning in large scale ad hoc networks," *MILCOM*, 2003.

[5] W. Liao, Y. Tseng, S. Wang, and J. Sheu, "A multi-path qos routing protocol in a wireless mobile ad hoc network," *ICN 2001*, 2001.

[6] C. Chereddi, P. Kyasanur, and N. Vaidya, "Design and implementation of a multi-channel multi-interface network," in *ACM REALMAN Workshop*, vol. 5, May 2006, pp. 23–30.

[7] K. Ramachandran, K. Almeroth, and E.M.B.-Royer, "A novel framework for the management of large-scale wireless network testbeds," in *WinMee Workshop*, April 2005.

[8] V. Navda, A. Kashyap, and S. Das, "Design and evaluation of imesh: an infrastructure-mode wireless mesh network," in *IEEE WoWMoM Symposium*, June 2005.

[9] *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications*, IEEE 802.11 Working Group Std., September 1999 - 2005.

[10] T. Moscibroda, R. Chandra, Y. Wu, S. Sengupta, P. Bahl, and Y. Yuan, "Load-aware spectrum distribution in wireless lans," *IEEE ICNP*, 2008.

[11] R. Chandra, R. Mahajan, T. Moscibroda, R. Raghavendra, and P. Bahl, "A case for adapting channel width in wireless networks," *ACM Sigcomm*, August 2008.

[12] R. Gummadi, R. Patra, H. Balakrishnan, and E. Brewer, "Interference avoidance and control," *HotNets VII*, 2008.

[13] Y. Yuan, P. Bahl, R. Chandra, T. Moscibroda, and Y. Wu, "Allocating dynamic time-spectrum blocks in cognitive radio networks," *MobiHoc*, 2007.

[14] R. Draves, J. Padhye, and B. Zill, "Routing in multiradio, multihop, wireless mesh networks," *ACM Mobicom*, 2004.

[15] C. Perkins and E. Royer, "Ad-hoc on-demand distance vector routing," *Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications*, February 1999.

[16] P. Leaves, K. Moessner, R. Tafazolli, D. Grandblaise, D. Bourse, R. Tonjens, and M. Breveglieri, "Dynamic spectrum allocation in composite reconfigurable wireless networks," *IEEE Communications Magazine*, May 2004.

[17] R. Sivakumar, P. Sinha, and V. Bharghavan, "Cedar: A core-extraction distributed ad hoc routing algorithm," *IEEE Journal on Selected Areas in Communication Vol. 17, No. 8*, August 1999.

[18] C. Perkins, "Quality of service for ad-hoc on-demand distance vector routing," *Mobile Ad Hoc Networking Working Group - Internet Draft*, July 2000.
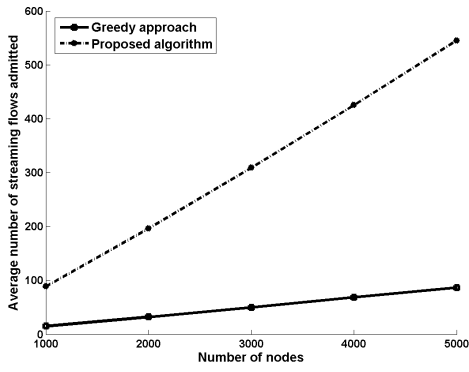
Fig. 7. Overall network utility as a function of network density
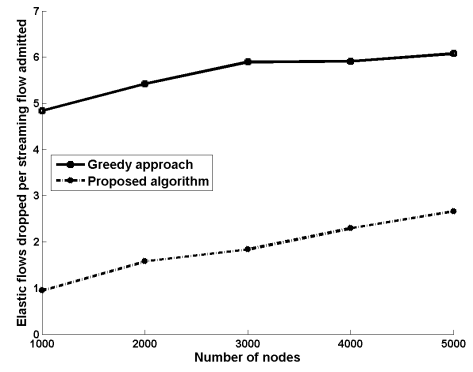

Fig. 8. Total number of dropped flows in the network as a function of network density
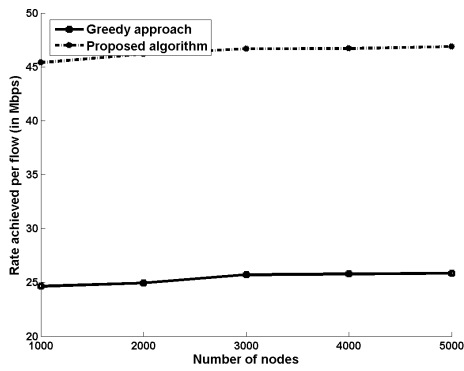

Fig. 9. Average rate achieved per flow as a function of network density
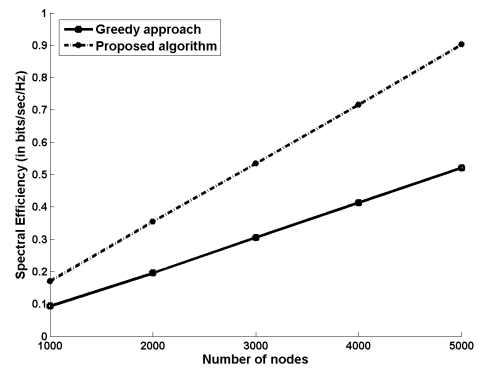

Fig. 10. Per-flow spectral efficiency achieved as a function of network density

[19] P. Kyasanur and N. Vaidya, "Routing and link-layer protocols for multi-channel multi-interface ad hoc wireless networks," *ACM SIGMOBILE MC2R*, vol. 10, pp. 31–43, January 2006.

[20] P. Kyasanur, C. Chereddi, and N. Vaidya. (2006, August) Net-x: System extensions for supporting multiple channels, multiple interfaces, and other interface capabilities. [Online]. Available: http://www.crhc.uiuc.edu/wireless/groupPubs.html.

[21] R. Draves, J. Padhye, and B. Zill, "Routing inmulti-radio, multi-hopwireless mesh networks," *MobiCom*, 2004.

[22] D. Ganesan, B. Krishnamachari, A. Woo, D. Culler, D. Estrin, and S. Wicker, "Complex behavior at scale: An experimental study of low-power wireless sensor networks," *UCLA CS Technical Report*, 2002.