Centralized Feedback-driven Rate Allocation Mechanism for CSMA/CA-based Wireless Mesh Networks

Kamran Jamshaid and Paul A.S. Ward
Shoshin Distributed Systems Group
Department of Electrical and Computer Engineering
University of Waterloo, Waterloo, Ontario N2L 3G1
{kmjamshai, pasward}@shoshin.uwaterloo.ca

ABSTRACT

Wireless Mesh Networks (WMNs) based on commodity 802.11 radios exhibit extreme unfairness, including starvation, in the presence of backlogged traffic sources. While the underlying reasons for this performance degradation are well understood, yet only a few of the proposed solutions are compatible with the current generation of closed 802.11 firmware/hardware. Most existing proposals limit sources to their fair share through a distributed rate control mechanism, in the process incurring a significant overhead to enable such local computation of fair rates.

In prior work we validated the feasibility of centralized rate control in WMNs, assuming knowledge of network topology and link interference. In this work we remove this requirement and propose a zero-overhead centralized rate controller. This controller can be implemented at traffic aggregation points such as gateway routers. Its response is driven by a feedback-based mechanism, allowing it to adapt to changes in network and traffic conditions. Simulations show that our controller can improve the fairness metrics by a factor of 3 over networks without any rate control, and is within 2% of the fairness metrics achieved by a centralized controller with omniscient knowledge of network topology and link interference.

Categories and Subject Descriptors

C.2.1 [Computer-Communication Networks]: Network Architecture and Design—Wireless communication

General Terms

Design, Performance

Keywords

CSMA/CA, 802.11 DCF, Fairness, Congestion control

1. INTRODUCTION

Wireless Mesh Networks (WMNs) based on commodity 802.11 radio chipsets are a low-cost alternative for last mile access. Such networks consist of utility-powered, minimal-mobility mesh routers, together with their clients. The mesh routers communicate with each other over multihop wireless links. The clients connect to their preferred router either via wire or over a (possibly orthogonal) wireless channel. Communication is typically to/from clients through the mesh routers over multiple wireless hops, to a gateway mesh router that has a wired connection to the wider world, typically the public Internet. There are a number of commercial products (e.g., [1]) as well as research testbeds (e.g., [3]) based on a similar architecture.

The performance limitations of 802.11 radios in multihop networks are well-known [12, 19, 23]. In particular, nodes multiple hops from the gateway may experience throughput unfairness, including flow starvation, under heavy traffic loads [7, 18]. This response is rooted in the behavior of the CSMA/CA MAC protocol, as well as its operational specifications in the IEEE 802.11 Distributed Coordination Function (DCF) access mechanism:

1. DCF provides all nodes in a single contention area with equal transmission opportunities. This MAC-level fairness does not translate to end-to-end fairness in multihop networks.

2. CSMA/CA radios may produce misaligned transmissions when two transmitters cannot directly carrier sense each other. This may result in excessive collisions at some receivers, or may even deprive other nodes of transmission opportunities.

Without an explicit rate feedback, these MAC characteristics produce long-term throughput unfairness for backlogged traffic sources. Existing congestion control protocols like TCP fail to provide such a rate feedback in CSMA/CA-based systems [7]. We cover the origins of this behavior in greater detail in Sect. 3.

A number of research publications (e.g., [7, 10, 18]) have proposed distributed algorithms that allow traffic sources to compute and enforce flow rate limits based on current contention levels in the network. These algorithms rely on periodic network-wide distribution of time-varying state information, and thus incur a significant overhead.

We have previously proposed the use of traffic-aggregation points like gateway nodes to enforce rate control policy objectives [11]. In access networks like WMNs, traffic is mostly directed either towards or away from such points. This provides the gateway node with a unified view of the network, creating an opportunity to enable centralized rate control based on this information. Our prior work tested the feasibility of centralized rate control in WMNs, assuming knowledge of network topology and link interference.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

Copyright 2008 ACM 978-1-60558-235-1/08/10 ...$5.00.
bility of such centralized control. We assumed that both network topology and link interference information were available at the gateway router, and used them to compute and enforce rate control. In this paper we remove these requirements and propose a feedback-driven centralized rate controller for adaptive transport protocols. Our feedback rate controller (FBRC) incurs zero-overhead; it uses existing traffic to probe for network capacity, and uses the network response as a feedback control to drive the rate allocation mechanism. By delaying or dropping excess traffic, it forces adaptive traffic sources to conform to the desired rate allocation. FBRC does not require modifications to the mesh routers or the client devices. It provides fair bandwidth allocation in multihop networks where competing flows traverse a varying number of hops and transmissions along individual links are scheduled based only on the localized view of the CSMA/CA-based transmitters.

Our contributions in this paper are as follows: First, we show that the multitude of existing work to improve TCP performance either by limiting congestion window size (e.g., [6]) or rate-limiting based on 4-hop propagation delays (e.g., [4]) fails to improve fairness. Second, we propose FBRC, a zero-overhead centralized rate controller that determines and enforces the correct rate limits for contending flows to fairly share the medium. Third, we propose a binary search heuristic that allows FBRC-like centralized controllers to quickly converge to fair rate allocations. Our heuristic uses feasible and efficient theoretical aggregate capacity bounds. We provide an outline of how such bounds can be computed for WMNs with many-to-one traffic flows.

This paper is organized as follows. We discuss related work in Sect. 2. In Sect. 3, we explain why DCF leads to unfairness in multihop networks with backlogged traffic, and how flow rate control mitigates this behavior. In Sect. 4, we present an architectural design of a centralized rate controller and discuss various design considerations. We evaluate the performance of FBRC in Sect. 5. We conclude by observing what issues remain open.

2. RELATED WORK

The challenges in using CSMA/CA-based MAC protocols for multihop networks have been discussed previously [13,18,19,23]. In general, a flow not only experiences inter-flow contention with other flows sharing the spectrum, but may even interfere with its own transmissions along the path to the destination (i.e., intra-flow contention) [24]. The degree of contention increases with increasing traffic loads. Related work in the literature addresses it from different perspectives: MAC-layer enhancements, transport-layer enhancements, and higher-layer rate control algorithms.

By far the largest body of literature specifically devoted to wireless network fairness is that of the MAC-layer solutions (see [21,24,14,18,20], among others). Such approaches tend to assume that contending flows span a single hop and fairness may be achieved by converging the MAC contention windows to a common value. However, optimal end-to-end fair allocation for multihop flows cannot be achieved by MAC scheduling techniques based only on the local information.

A number of studies have associated the inter-flow contention experienced by a TCP flow to its TCP congestion window exceeding an optimum size. For a chain topology, the optimum window size that maximizes spatial reuse is $1/4^{th}$ the number of hops between a source and destination [6]. Note that this does not resolve any inter-flow contention, and subsequent unfairness and starvation may still ensue. Modifications or alternatives to TCP for multihop networks have also been proposed (e.g., [4]), but these require modifying the transport stack on all mesh routers. This presents integration challenges when these mesh routers communicate with a wired host running the standard network stack. A goodput control scheme with gateway pacing for downstream flows is also proposed in [4]; this is, however, limited to flows with equal wireless path lengths. In some topologies downstream flows may already have better fairness characteristics than upstream flows (e.g., see Fig. 2).

Our solution allows fairness to be enforced for both upstream and downstream flows, independent of their path lengths, and without any modifications to individual mesh routers.

Rate control algorithms operating outside the transport layer have also been shown to improve fairness. Given network topology and traffic demands, conflict graph models [8] such as the clique model [16], its time-fairness extension [7], as well as Jun and Sichitiu’s nominal capacity model [12], may be used to compute optimal bounds on network capacity. Jain et al. [10] and Raniwala et al. [18] propose distributed algorithms based on this conflict graph approach. Raniwala et al. [17] have proposed using an AIMD-based rate control alongside a congestion sharing mechanism. Their approach is designed for many-to-one communication paradigm of sensor networks, but fails in one-to-many (typical downloads in WMNs) scenarios. In that latter case, the congestion information cannot be correctly propagated to all potential interferers.

3. DCF IN MULTIHOP NETWORKS

A core function of any MAC protocol is to provide fair and efficient access to the transmission medium. Here we describe the performance limitations of DCF in multihop networks when the contending nodes are (i) within, and (ii) outside, mutual carrier sense range.

3.1 Nodes within carrier sense range

On average, DCF provides equal transmission opportunities to nodes within carrier sense range. This per-station fairness does not translate to flow-level or end-to-end fairness in WMNs where nodes higher in the routing tree relay an increasing amount of aggregate traffic. Consequently, these nodes experience high queue drops; this results in capacity loss when the dropped packets originated from other nodes and have already consumed a portion of the shared spectrum. For example, in a 2-hop chain (Fig. 1) with upstream flows from nodes 1 and 2 to the gateway node 0, the max-min fair share of nodes 1 and 2 is $\frac{W}{3}$ each, where $W$ is
the nominal link capacity. This adds up to an aggregate fair network capacity of $2\frac{W}{W}$. However, with 802.11 MAC and backlogged senders, the aggregate network capacity reduces to $\frac{W}{W}$ with node 2 starving [12].

Note that this capacity loss from queue overflows affects upstream flows only. For downstream flows, 802.11 MAC fairness limits the packets a gateway can inject in the wireless medium, while per-station fairness ensures that relay nodes always get sufficient transmission opportunities to deliver in-flight packets. Queue drops from excess traffic occur at the gateway router before entering the wireless medium, thus avoiding wireless capacity loss. We validate this behavior for a 3-hop chain using ns-2 [21] simulations. Fig. 2 shows the throughput response for both upstream and downstream flows against the offered load. Assuming 1 Mb/s wireless links, the max-min fair share per stream is about 150 kbps. All flows fairly share the medium till the offered load hits this fair share point. Beyond this point, the upstream flows (Fig. 2(a)) exhibit increasing unfairness with rising load, first starving node 3 and then node 2. Node 1’s throughput saturates around 400 kbps with other nodes starving; this capacity loss is due to queue overflows at intermediate routers. The downstream flows in Fig. 2(b) do not exhibit this unfairness and capacity loss.

### 3.2 Nodes outside carrier sense range

When two transmitters are outside carrier sense range, the interference range of the two transmitters. This scenario is valid even when $R_1$ is outside the transmission range but still within the interference range of $S_2$.

### 3.3 DCF in rate-limited multihop networks

Link-layer retransmissions in DCF provide recovery against transient wireless losses. Our results in Fig. 2 suggest that these recovery mechanisms provide reliable delivery guarantees as long as the offered load does not exceed the fair-share network capacity. Beyond that, queue drops dominate and this invokes higher layer congestion control mechanisms, when available. We have corroborated this observation with a large number of experiments on other topologies, and this is consistent with the results in [7, 11, 18]. The remaining challenge is to determine the correct fair-share capacity and rate-limiting the nodes accordingly.

Rate limits have also been proposed to improve the performance of individual TCP streams in multihop networks. Work in this area either either limit the size of TCP congestion window (e.g., [5]), or limit a flow to a rate based on 4-hop propagation delays in the network (e.g., [4]). These

![Figure 3: Illustrative topologies showing DCF performance limitations in multihop networks.](image)
rate limits are designed to reduce intra-flow contention along the path to the destination, but are not sufficient to remove inter-flow contention. We have validated this on a 4-hop chain (Fig. 1) with uploads from nodes 1, 2, 3 and 4 to the gateway node 0. The TCP congestion window limit was set to a maximum of 2. Fig. 4 shows that limiting congestion window size shows only a minor improvement over the base case with no such arbitrary limits.

4. FEEDBACK RATE CONTROLLER

FBRC is a system module on the gateway mesh router. It sits between the MAC layer and the network layer, operating transparently between them. Its three main components perform the following functions (Fig. 5):

1. Flow classification
2. Rate evaluation and allocation
3. Flow rate enforcement

Figure 4: Limiting TCP congestion window size does not eliminate inter-flow contention, and has little impact on flow rate fairness.

Figure 5: The three main architectural components of FBRC: flow classification (Sect. 4.1), rate evaluation and allocation (Sect. 4.2), and rate enforcement token buckets (Sect. 4.3).

4.1 Flow classification

In this first step, FBRC performs flow classification for all data traffic (ingress and egress) through the gateway. In this work we classify flows based on the source or destination node. Thus a flow \( f \) represents the aggregate of all microflows originating from, or destined to, node \( i \) in the network. In this context, we use nodes and flows interchangeably in our discussion. Our classification methodology requires a simple lookup of the packet header given a known offset, and can be performed efficiently. We note that such a classification is consistent with the common practices employed by ISPs on wired access networks, where capacity is managed on a per-subscriber basis.

4.2 Rate evaluation and allocation

The rate evaluation component uses the measured flow rates to adjust the behavior of the rate allocation component. Together, the two constitute a closed-loop feedback controller shown in Fig. 6.

Rate evaluation: This is based on a simple principle: if all flows obtain their allocated rate, then the network capacity is possibly underutilized (i.e., the network is operating in a regime to the left of the fair share line as depicted by the behavior in Fig. 2(a)); the resulting control action is to increase flow rates. However, if a flow obtains less than its allocated rate, then we are likely driving the network beyond its fair capacity (similar to the representative behavior beyond the fair rate in Fig. 2(a)). Consequently, the flow rates need to be decreased.

Figure 6: Rate evaluation and allocation work in a closed-loop feedback control.

FBRC considers time in slots (or epochs) of duration \( \delta \). The rate obtained by flows in epoch \( \tau_i \) determines the control (corrective) action that is applied by the controller for the duration of the epoch \( \tau_{i+1} \). The actual time duration of the epoch (i.e., \( \delta \)) is configurable, though for stability it should operate at different timescales than the control action of TCP senders. For instance, \( \delta \) can be set to multiples of round-trip time so that TCP sources can react to changes in rate allocation and stabilize around their new values.

At the end of every time epoch \( \tau_i \), the mechanism measures the rate \( r_i \) obtained by a flow \( f_i \) and compares it to the rate \( b_i \) allocated for that flow during the epoch \( \tau_i \). We associate a weight \( w_i \) with a flow \( f_i \), such that \( b_i \) equals \( \frac{\sum_{i=1}^{N} w_i}{\sum_{i=1}^{N} w_i} \) fraction of the available capacity for the \( N \) active flows. Two possibilities exist:

1. If \( r_i \geq b_i \) for all \( i \in N \), the mechanism assumes the network capacity may be underutilized, i.e., the current estimate of network capacity \( C_{\text{meas}} = \sum_{i=1}^{N} r_i \) is low. It signals the rate allocation component to increase flow rates.
2. If \( r_i < b_i \) for any \( i \in N \), the mechanism determines that the flow \( f_i \) is experiencing unfairness. It assumes that its current estimate of the capacity \( C_{\text{meas}} = \sum_{i=1}^{N} r_i \) is too high. It signals the rate allocation component to decrease flow rates to lower the capacity utilization.

The pseudo-code for such a rate evaluation component is shown in Algo. 1.

Rate Allocation: The rate allocation component determines new flow rates based on feedback from the rate evaluation component. It first determines a network capacity
Input: Epoch duration $\delta$, unfairness threshold $\gamma$ 
\hspace{1cm} (0 < \gamma \leq 1), allocated rate vector $[b_1, b_2, ..., b_N]$, 
\hspace{1cm} and the measured rate vector $[r_1, r_2, ..., r_N]$

Output: Rate increase or decrease decision

1 while once every $\delta$ time units do
2 \hspace{1cm} if $\frac{C_{est}}{b_i} < \gamma$ for any active flow $f_i$ then
3 \hspace{1cm} \hspace{1cm} decreaseRates;
4 \hspace{1cm} else
5 \hspace{1cm} \hspace{1cm} increaseRates;
6 end
7 end

Algorithm 1: Rate evaluation

The rate allocation component may use any number of heuristics to determine the new $C_{est}$. It has to search through the space of feasible allocations for the new rate allocation. A simple algorithm using exponential increase/decrease in aggregate capacity is shown in Alg. 2. In Sect. 4.4 we propose a heuristic that uses binary search within measured upper and lower aggregate network capacity bounds to allow for faster convergence to the fair rate values.

Input: $C_{meas} = \sum_{i=1}^{N} r_i$, flow weight vector $[w_1, w_2, ..., w_N]$ 
Output: New token rate vector $[b_1, b_2, ..., b_N]$

1 increaseRates begin
2 \hspace{1cm} $C_{est} = \alpha \times C_{meas}$; /* $\alpha > 1$ */
3 \hspace{1cm} for every active flow $f_i$ do
4 \hspace{2cm} $b_i = \frac{w_i}{\sum_{i=1}^{N} w_i} \times C_{est}$;
5 end
6 end
7 decreaseRates begin
8 \hspace{1cm} $C_{est} = \beta \times C_{meas}$; /* $\beta < 1$ */
9 \hspace{1cm} for every active flow $f_i$ do
10 \hspace{2cm} $b_i = \frac{w_i}{\sum_{i=1}^{N} w_i} \times C_{est}$;
11 end
12 end

Algorithm 2: Rate allocation using a simple exponential increase/decrease in aggregate capacity

4.4 Design considerations

Rate increase/decrease heuristics: These heuristics help the rate allocation component explore the space of feasible allocations in search of the desired rate allocation. We prefer heuristics with quick convergence characteristics. Alg. 2 outlined a simple heuristic with exponential increase/decrease in capacity estimates. We now propose a binary search heuristic that uses bounds on aggregate capacity limits to provide faster convergence to fair rate values. It converges the aggregate capacity estimate, $C_{est}$, to current network conditions in logarithmic time (approx. $\log_{2} K$ steps, where $K$ is the set of feasible rate allocations per the fairness criterion). This heuristic works as follows: we first determine lower and upper bounds on feasible $C_{est}$ (In Appendix A we provide an outline of how the initial upper and lower bounds on $C_{est}$ can be determined for WMNs with many-to-one traffic patterns.) Then using binary search, we use the rate feedback in epoch $\tau_{1}$ to determine $C_{est}$ for epoch $\tau_{1+1}$. The pseudo-code for this is shown in Alg. 3.

Input: $C_{meas} = \sum_{i=1}^{N} r_i$, flow weight vector $[w_1, w_2, ..., w_N]$, and upper and lower aggregate capacity bounds $C_{up}$ and $C_{low}$ respectively.
Output: New token rate vector $[b_1, b_2, ..., b_N]$

1 increaseRates begin
2 \hspace{1cm} $C_{low} = C_{meas}$;
3 \hspace{1cm} $C_{est} = \frac{C_{meas}+C_{up}}{2}$
4 \hspace{1cm} for every active flow $f_i$ do
5 \hspace{2cm} $b_i = \frac{w_i}{\sum_{i=1}^{N} w_i} \times C_{est}$;
6 end
7 end
8 decreaseRates begin
9 \hspace{1cm} $C_{up} = C_{meas}$;
10 \hspace{1cm} $C_{est} = \frac{C_{meas}+C_{low}}{2}$
11 \hspace{1cm} for every active flow $f_i$ do
12 \hspace{2cm} $b_i = \frac{w_i}{\sum_{i=1}^{N} w_i} \times C_{est}$;
13 end
14 end

Algorithm 3: Rate allocation using binary search within aggregate capacity bounds

Dynamic flows: Our flow bundles are aggregates of microflows, and we expect them to be long-lived for durations lasting tens of seconds. However, when they do terminate, any unused capacity should be fairly reallocated. Similarly, when a new flow emerges, the rate allocation of existing flows need to be adjusted. For instance, the procedure decreaseRates may be called when a new flow is detected, as the new per-flow allocation will decrease in the presence of an additional flow. Similarly, increaseRates may be executed when a flow terminates.

Flow activation/termination can be detected in multiple ways. TCP stream activation and tear-down can be detected by the exchange of the TCP-specific three-way handshake messages. In our case where flow bundles constitute multiple TCP streams, we simply use the presence or absence of packets to determine the current state of stream activity. In contrast to prior approaches [4], this reduces the complexity as well as the state information maintained by the centralized controller. We evaluate FBRC with dynamic flows in Sect. 6.
5. FAIR SHARE COMPUTATION MODEL

Our prior work [11] showed the feasibility of centralized rate-control using a computation model that required knowledge of network topology and link interference. In Sect. 6 we benchmark the performance of FBRC against this model. Here we briefly describe this model for completeness.

The problem of modeling link interference so as to efficiently compute the fair-share capacity has been addressed previously (e.g., [8]). We have implemented a restricted version of the model developed by Li et al. [11]. Our simplifications constrain us to absolute fair sharing of the bandwidth, with single-rate routers. The Li et al. model treats the network as a graph, with mesh nodes as vertices connected via bi-directional wireless links. A link interferes with another link if either endpoint of one link is within transmission range of either endpoint of the other link. Thus, the set of all links that interfere with a given link, referred to as the collision domain of that link, are all those within two hops of either endpoint of the link. The model assumes that the links within a collision domain cannot transmit simultaneously. This actually over-estimates link contention. However, given that link interference, defined by transmission range rather than interference range, is under-estimated, the presumption (borne out by detailed simulation studies) is that the overall model is approximately correct.

It is then sufficient to determine the bottleneck collision domain, which will be a function of the usage of the links within each collision domain. Link usage is determined by routing and demand. When routing is fixed (as in static WMNs) and the backlogged sources have a persistent demand, we can then compute the load over each link, and in turn compute the load in each collision domain. Given the single-rate assumption, the bottleneck collision domain is simply that domain with the greatest load, and the fair share is determined simply by dividing the rate by the load.

6. PERFORMANCE EVALUATION

We have implemented FBRC in ns-2 [21]. Our module sits directly on top of the wireless MAC layer, and can rate limit both upstream and downstream flows. A gateway router with multiple wireless interfaces may use an FBRC module per interface to avoid synchronization between the flows on different interfaces. We rate limit data packets only; system housekeeping messages such as routing updates bypass FBRC and are not rate limited.

We simulated FBRC with the binary search heuristic from Sect. 4.4. The epoch duration δ was set to 10 sec; choosing a value larger than the control action of TCP is important for stability. Our unfairness threshold γ (Algo. 1, line 2) was set to 0.7. We simulated elastic flows using an infinite file transfer with TCP NewReno [5]. Here we present our results for equal fair sharing of the network capacity; our results for weighted fair sharing were similar and are omitted due to space constraints. We use Jain’s Fairness Index (JFI) [9] as a quantitative measure of aggregate fairness. We also list \( \frac{\text{min. flow rate}}{\text{fairrate}} \) and \( \frac{\text{max. flow rate}}{\text{fairrate}} \) to illustrate the imbalance between the minimum and the maximum throughput flows in a given experiment.

6.1 Long-lived elastic TCP flows

We evaluated FBRC on a number of different chain, grid, and random network topologies, with up to a maximum of 35 simultaneously active nodes transmitting via a single gateway. For a given topology, experiments were repeated 25 times with different random seeds and random flow activation sequences, and the results averaged. For performance benchmarks, we repeated the same set of experiments (i) without any rate limiting, and (ii) with gateway enforced rate limiting based on the computational model. Our results are summarized in Table 1. FBRC shows an approximately three-fold improvement in JFI over networks without any rate control, and is within 2% of the JFI obtained using the computational model approach.

6.2 Short-lived elastic TCP flows

We also evaluated the responsiveness of FBRC for dynamic elastic flows. Here we show our results for a 7-hop chain (Fig. 4). Initially, five flows are active. We terminate flows 1→0 and 5→0 at 200 sec, and flow 7→0 at 300 sec. Finally, at 400 sec., we reactivate flows 1→0 and 7→0. Fig. 7 plots throughput against time, averaged over 5 sec. We are interested in the time flows take to converge around their new fair rates. This convergence time is a function of the TCP state. A TCP agent begins in slow start, where its congestion window builds up exponentially over time. This allows flows 1→0 and 7→0 to rapidly approach their fair rate within one and three averaging intervals of our plot respectively. Thus FBRC allows new flows to quickly ramp up their rates to their allocated share of the network capacity.

6.3 Rate-constrained TCP flows

FBRC is optimized for continuously backlogged elastic data sources. Here we analyze its performance when rate-constrained TCP flows are introduced in the traffic mix.

The underlying rate increase/decrease heuristics deter-
Table 2: FBRC performance with rate-limited TCP.

<table>
<thead>
<tr>
<th>Node 4 rate</th>
<th>Node avg. optimal</th>
<th>Node 2 avg. optimal</th>
<th>Node 3 avg. optimal</th>
<th>Node 4 avg. optimal</th>
</tr>
</thead>
<tbody>
<tr>
<td>10 kb/s</td>
<td>0.70</td>
<td>0.70</td>
<td>0.80</td>
<td>1</td>
</tr>
<tr>
<td>20 kb/s</td>
<td>0.74</td>
<td>0.74</td>
<td>0.74</td>
<td>1</td>
</tr>
<tr>
<td>30 kb/s</td>
<td>0.79</td>
<td>0.79</td>
<td>0.79</td>
<td>1</td>
</tr>
<tr>
<td>40 kb/s</td>
<td>0.84</td>
<td>0.84</td>
<td>0.84</td>
<td>1</td>
</tr>
<tr>
<td>50 kb/s</td>
<td>0.90</td>
<td>0.90</td>
<td>0.90</td>
<td>1</td>
</tr>
<tr>
<td>60 kb/s</td>
<td>0.94</td>
<td>0.94</td>
<td>0.94</td>
<td>1</td>
</tr>
<tr>
<td>70 kb/s</td>
<td>1.04</td>
<td>1.03</td>
<td>1.03</td>
<td>1</td>
</tr>
</tbody>
</table>

70 kb/s is a close approximation to the fair rates enforced using a computational model based approach.

An inherent limitation of FBRC is that it works with adaptive transport protocols (e.g., TCP) that respond to notifications regarding current network conditions. For non-adaptive UDP-based streams, FBRC rate information would need to be signaled to source nodes to be enforced locally. This is an area for future work. Finally, we note that FBRC in the presence of rate-constrained flows may result in suboptimal use of the network capacity. This can be addressed by having FBRC maintain history about flow rates, and redistributing a flow’s unused allocation based on past observations. We hope to address this in our future work.

8. REFERENCES

APPENDIX

A. AGGREGATE CAPACITY BOUNDS

Consider a network with \( N \) mesh routers exchanging data via the gateway router. We assume that each router has a single wireless interface connected to an omni-directional antenna with unit gain. Let the wireless link capacity be \( W \) bits/s. For simplicity, we assume that each node transmits with equal power, obtaining a transmission range of \( r \) units. Note that our theoretical bounds assume that the gateway always has data to send/receive and that the wireless spectrum is fully utilized.

A.1 Upper network capacity bound

The aggregate transport capacity is maximized when the gateway is either receiving or transmitting data all the time. If the link capacity is \( W \) bits/s, the upper bound on the aggregate send/receive at the gateway is also \( W \) bits/s.

A.2 Lower network capacity bound

The upper bound on aggregate capacity is obtained when the active node(s) consume minimal resource (e.g., spectrum) to communicate with the gateway. This happens when the active nodes are within one-hop (i.e., within \( r \) units) of the gateway. Conversely, an efficient lower bound on aggregate capacity is obtained when all active nodes consume maximal resource to reach the gateway.

Let the interference range for a node be \( m \) units, where \( m \geq r \). Let \( d \) denote the distance that allows concurrent transmissions along a path. For CSMA/CA radios, \( d \geq m \).

A flow consumes increasing spectral resource with each additional hop, till the number of hops reach \( d \) and the resulting spatial reuse allows the transmissions to be pipelined. At this point, the flow is consuming maximal spectral resource as adding in any additional hops will still allow for concurrent, pipelined transmissions. Note that the spatial reuse depicted by \( d \) is a function of the carrier sense threshold of the radio. Zhai and Fang [25] show that maximum spatial reuse of a multihop flow is \( \frac{W}{d} \) in order to maintain a minimum of 10dB SINR required for correctly decoding a packet. This sets a lower bound of \( \frac{W}{d} \) on the theoretical aggregate send/receive capacity of the gateway. Practically achievable bounds may be lower due to packet losses, non-optimal carrier sense threshold, and other MAC-specific overheads.

http://www.isi.edu/nsnam/ns/