On the Throughput/Bit-Cost Tradeoff in CSMA Based Cooperative Networks

Georg Böcherer* and Rudolf Mathar*

*Institute for Theoretical Information Technology
RWTH Aachen University, 52056 Aachen, Germany
Email: {boecherer,mathar}@ti.rwth-aachen.de

Abstract—Wireless local area networks (WLAN) still suffer from a severe performance discrepancy between different users in the uplink. This is because of the spatially varying channel conditions provided by the wireless medium. Cooperative medium access control (MAC) protocols as for example CoopMAC were proposed to mitigate this problem. In this work, it is shown that cooperation implies for cooperating nodes a tradeoff between throughput and bit-cost, which is the energy needed to transmit one bit. The tradeoff depends on the degree of cooperation. For carrier sense multiple access (CSMA) based networks, the throughput/bit-cost tradeoff curve is theoretically derived. A new distributed CSMA protocol called fairMAC is proposed and it is theoretically shown that fairMAC can asymptotically achieve any operating point on the tradeoff curve when the packet lengths go to infinity. The theoretical results are validated through Monte Carlo simulations.

I. INTRODUCTION

The motivation for this work is the performance discrepancy for different users in WLAN uplinks as observed in [1]. Cooperation in wireless networks has drawn a lot of attention in order to mitigate throughput discrepancy between users in wireless networks. Based on the early results presented in [2], the authors in [3], [4] illustrate that cooperation between two co-located users can be beneficial for both users when transmitting over fading channels. Several works propose distributed protocols to coordinate cooperation at the MAC layer, for instance *r*DCF [5] and CoopMAC [6]. Both protocols enable two-hop transmission as an alternative to direct transmission for WLAN. These protocols also coordinate cooperation on the physical layer [7]. The benefits of cooperation for the whole network have been discussed in [8], [9]. In [5], [6], [10], the authors propose to select the best relay for each transmission separately. However, if one node is determined as the best relay for many nodes, its energy consumption will be very high compared to other nodes. In [11] we investigate distributed cooperative protocols for two users based on CSMA, where both users were constrained to achieve same throughput with same energy consumption, i.e., full fairness. This was achieved by individual transmission power adaption for each user. However the extension to scenarios with many users is rather unrealistic

since it would require centralized power allocation, which is difficult to implement in ad-hoc networks.

In this work, we choose a different approach and restrict all transmitters to the same average transmit power during transmission. We impose throughput fairness as in [11], i.e., on the long term, each node effectively transmits information at the same rate to the common access point (AP). For evaluation, we consider the effective throughput *and* the resulting bit-cost in terms of average energy per transmitted data.

- We identify a throughput/bit-cost tradeoff in cooperative networks: a potential helper increases his own *throughput* by cooperating, but he also increases his *bit-cost*.
- We analytically derive formulas for the throughput/bit-cost tradeoff curve that results from timesharing between CSMA based CoopMAC [6] and conventional CSMA based Direct Link, where all nodes transmit directly to the AP.
- We propose a distributed protocol called fairMAC and show, both theoretically and by Monte Carlo simulation, that fairMAC can asymptotically reach the tradeoff curve when the packet lengths go to infinity.

The remainder of this paper is organized as follows. In Section II, we provide our system model. In Section III, we introduce the main topics of our work in a simplified setup. We define the new protocol fairMAC in Section IV and we analyze it theoretically in Section V. Finally, we validate our theoretical results through simulation in Section VI.

II. SYSTEM MODEL

We consider a network of N nodes that seek to transmit their data to a common AP. For each pair of nodes k, l of the network, we associate with the transmission from k to l the achievable rate R_{kl} in bit/s. We denote by R_k the achievable rate for direct transmission from node k to the AP. We normalize the amount of data per packet to 1 bit. The packet length for a transmission from node k to the AP is then given by $1/R_k$. For the cooperative protocols CoopMAC and fairMAC (to be introduced in this work), some nodes have the possibility to transmit their packets to the AP via a helper. Following [6], the helper is chosen such that the overall transmission length is minimized: h can help k if and

only if

$$h = \arg\min_{l \in [1,N]} \frac{1}{R_{kl}} + \frac{1}{R_l} \quad \text{and} \quad \frac{1}{R_{kh}} + \frac{1}{R_h} < \frac{1}{R_k}. \quad (1)$$

If such an h exists for node k, we denote it by h_k. It is the best relay of k for two-hop transmission and transmitting from k via h_k to the AP takes less time than transmitting directly from k to the AP. We assume that node k knows the rate R_k and if it has a helper h_k according to (1), it also knows R_{kh_k}. We have a quasi-static environment in mind where a part of the nodes continuously experiences a channel much worse than other nodes. We therefore assume that the rates of the links remain constant over the period of interest.

The communications setup is throughout the paper as follows: each transmitted packet contains 1 bit of information. All nodes have an infinite amount of data that they want to transmit to a common AP. All nodes are restricted to the same transmit power $\mathcal{E}$ during transmission. The investigated strategies aim to guarantee the same effective throughput on the long term to all nodes independent of their transmission rates.

III. THROUGHPUT/BIT-COST TRADEOFF

In this introductory section, we give a short overview over the main topics of this work: we first define throughput and bit-cost. We then present the tradeoff between throughput and bit-cost in cooperative networks.

A. Throughput and Bit-Cost

The *throughput* S_k of node k is the average amount of data bits per time that node k successfully transmits. Only data belonging to k is taken into account; data that k forwards for other nodes does not contribute to the throughput S_k. Let $\bar{\mathcal{E}}_k$ denote the *average* power of node k ($\bar{\mathcal{E}}$ is given by transmit power $\mathcal{E} \times$ transmission time/overall time). In contrast to the throughput S_k, power spent while forwarding data of other nodes *does* contribute to $\bar{\mathcal{E}}_k$. We define the *bit-cost* B_k of k as

$$\mathsf{B}_k = \frac{\bar{\mathcal{E}}_k}{\mathsf{S}_k} \quad (2)$$

i.e., it measures the average amount of energy that node k has to spend to successfully transmit one own data bit.

For exposition and comparison, we consider in this section *Round Robin* as a centralized time division multiple access (TDMA) strategy. In a network of N nodes scheduled with Round Robin, the nodes transmit one after each other in a circular order. Denote by s_k the *travel time* of one bit of node k and denote by t_k the *transmission time* of node k, i.e., the overall time that node k is transmitting in one round. If node k is transmitting directly to the AP and does not forward data of other nodes, then $s_k = t_k = 1/R_k$. If node k is transmitting directly to the AP and is forwarding data of the number of H_k other nodes per round, then $s_k = 1/R_k$ and $t_k = (H_k+1)/R_k$. If node k transmits via node h_k, then $s_k = 1/R_{kh_k} + 1/R_{h_k}$ and $t_k = 1/R_{kh_k}$. Throughput and bit-cost of node k are thus
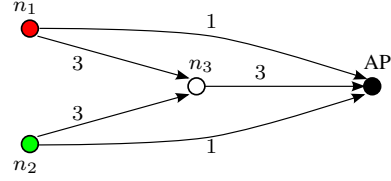

Fig. 1. A simple network with 3 nodes and one AP. According to (1), node n_3 is a potential helper for both node n_1 and node n_2.

given by

$$\mathsf{S}_k = \frac{1\text{bit}}{\sum_{k=1}^N s_k} \quad \text{and} \quad \mathsf{B}_k = \frac{\bar{\mathcal{E}}_k}{\mathsf{S}_k} = \frac{\frac{t_k \mathcal{E}}{\sum_{k=1}^N s_k}}{\frac{1\text{bit}}{\sum_{k=1}^N s_k}} = \frac{t_k \mathcal{E}}{1\text{bit}} \quad (3)$$

Note that $\mathsf{S}_k = \mathsf{S}_l$ for all $k,l = 1, \dots, N$. We can thus omit the index and simply refer to throughput S, but we have to keep in mind that S is the throughput per node and not the throughput sum over all nodes in the network.

B. A Toy Example

We now consider the simple network displayed in Figure 1. Three nodes n_1, n_2, and n_3 want to transmit to the same AP. All nodes use the transmit power of $\mathcal{E} = 1$ W. The rates are

$$R_{n_1} = R_{n_2} = 1\frac{\text{bit}}{\text{s}}, \quad R_{n_1 n_3} = R_{n_2 n_3} = R_{n_3} = 3\frac{\text{bit}}{\text{s}}. \quad (4)$$

For simplicity, we omit units in the following. Because of $1/3 + 1/3 < 1$, according to (1), n_3 is a potential helper for both n_1 and n_2. For clear exposure, we postpone distributed scheduling through random access to the following sections IV and V and schedule transmissions through Round Robin. The nodes n_1, n_2, and n_3 transmit one at a time in the fixed order $n_1, n_2, n_3, n_1, n_2, n_3, \dots$. In Direct Link, each node transmits one bit at a time directly to the AP, which takes the travel time 1 for nodes n_1 and n_2 and the travel time $1/3$ for node n_3. In CoopMAC, nodes n_1 and n_2 first transmit their bits to n_3, which takes the time $1/3$. After receiving a bit from n_1 or n_2, node n_3 immediately forwards the received bit to the AP, which again takes the time $1/3$. Thus, the travel time in CoopMAC for bits of n_1 and n_2 is $1/3 + 1/3 = 2/3$ and for n_3, it is $1/3$. We can now use (3) to calculate throughput and bit-cost of Round Robin based Direct Link and CoopMAC. For Direct Link, we get

$$\mathsf{S}^{\text{dir}} = \frac{1}{\frac{1}{1} + \frac{1}{1} + \frac{1}{3}} = \frac{3}{7}, \quad \mathsf{B}_{n_1}^{\text{dir}} = \mathsf{B}_{n_2}^{\text{dir}} = 1, \quad \mathsf{B}_{n_3}^{\text{dir}} = \frac{1}{3}. \quad (5)$$

For CoopMAC, we get

$$\mathsf{S}^{\text{coop}} = \frac{1}{\frac{2}{3} + \frac{2}{3} + \frac{1}{3}} = \frac{3}{5}, \quad \mathsf{B}_{n_1}^{\text{coop}} = \mathsf{B}_{n_2}^{\text{coop}} = \frac{1}{3}, \quad \mathsf{B}_{n_3}^{\text{coop}} = 1. \quad (6)$$

As we can see, cooperation increases throughput from $3/7$ to $3/5$ and decreases the *average* bit-cost from $7/9$ to $5/9$. However, the bit-cost of the helping node n_3 increases because of cooperation from $1/3$ to 1: *From the perspective of the*
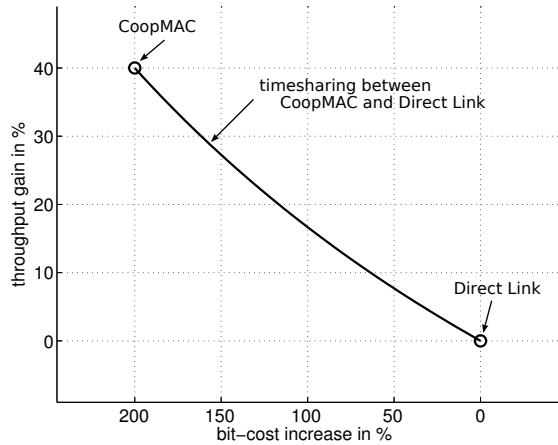
Fig. 2. The throughput/bit-cost tradeoff for the potential helper n_3 in the network displayed in Figure 1. The horizontal axis displays the bit-cost increase in percent compared to Direct Link. The vertical axis displays the throughput gain in percent compared to Direct Link. In the figure, the operating points of CoopMAC and Direct Link and the tradeoff curve that results from timesharing between these two strategies is displayed.

helping node n_3, there is a tradeoff between throughput and bit-cost. Through timesharing between CoopMAC and Direct Link, any other operating point in-between can be made available to n_3. We use CoopMAC for the fraction of time α and Direct Link for the fraction of time $1 - \alpha$. The average power of n_3 in CoopMAC and Direct Link is respectively

$$\bar{\mathcal{E}}_{n_3}^{\text{coop}} = \frac{3}{5} \quad \text{and} \quad \bar{\mathcal{E}}_{n_3}^{\text{dir}} = \frac{1}{7} \tag{7}$$

and following (3), we get for n_3 the tradeoff-curve parameterized by α

$$\mathsf{S}^\alpha = \alpha \mathsf{S}^{\text{coop}} + (1 - \alpha)\mathsf{S}^{\text{dir}}, \quad \mathsf{B}_{n_3}^\alpha = \frac{\alpha \bar{\mathcal{E}}_{n_3}^{\text{coop}} + (1 - \alpha)\bar{\mathcal{E}}_{n_3}^{\text{dir}}}{\alpha \mathsf{S}^{\text{coop}} + (1 - \alpha)\mathsf{S}^{\text{dir}}} \tag{8}$$

A plot can be found in Figure 2. The timesharing parameter α determines the degree of cooperation in the network: for $\alpha = 1$, the potential helpers are fully cooperative (CoopMAC) and for $\alpha = 0$, the potential helpers do not cooperate at all (Direct Link).

IV. CSMA BASED FAIRMAC PROTOCOL

For sake of clarity, we make some simplifying assumptions for the MAC layer. Since we are interested in high throughput for all nodes, we assume that all nodes operate in saturation mode, i.e., they are backlogged and we do not need to consider packet arrival processes in our analysis. We consider slotted CSMA with the two parameters slot length σ and transmit probability τ. In wireless networks, there are several reasons for packet losses. We include in our work packet losses because of interference (collision) but neglect other forms of packet losses. By assigning appropriate probabilities to other kinds of packet losses, they can be incorporated into our model, e.g., packet losses because of deep fade are considered in an ongoing research project at our institute. We further

assume that control headers and acknowledgments (ACK) are transmitted at a base rate and that they can be decoded by all nodes in the network. To remain general, we assume that data packets are large enough such that the specific size of control data is negligible. Finally, we assume that ACKs never get lost.

A. Reference Protocols

We start by defining the two reference protocols Direct Link and CoopMAC. While we used both terms for Round Robin based strategies in the previous section, they refer to CSMA based strategies here and hereafter, if not explicitly stated differently.

1) Direct Link [12]: When node k seeks to transmit a packet, it competes for the medium according to CSMA: if k senses the channel idle in time slot m, it initiates a transmission with probability τ in time slot $m + 1$. If no other node is transmitting at the same time, the AP can decode the packet and sends an ACK in return. Otherwise, a collision occurs; no ACK is sent by the AP; node k declares its packet lost and will try to transmit again the same packet later.

2) CoopMAC in base mode [6]: All nodes initiate the transmission of an own packet in the same way as in Direct Link. Assume that node k initiates a transmission. We have to distinguish two situations.

• Node k has no helper. The transmission is performed according to the Direct Link protocol.

• Node k has a helper h. In this case, k transmits its packet to h at rate R_{kh}. If h can decode the packet, it immediately forwards the packet to the AP at rate R_h. The AP sends an ACK to k. If h cannot decode the packet because of collision, it remains idle. Node k detects the collision by not receiving the ACK. Node k declares its packet lost and tries to transmit the same packet again via h later.

B. fairMAC

CoopMAC was designed to maximize throughput. However, the resulting bit-cost of potential helping nodes compared to other nodes can become very large, as we have seen in the previous section. Although a node addressed for help can in principal refuse to help, bit-cost control at helping nodes is not incorporated in CoopMAC. This is because the source k decides when the helper h has to help: h forwards immediately the packet from k. In fairMAC, this decision is taken by h: node h stores the data from k and transmits it in conjunction with one of his own future packets. This procedure is managed in a distributed manner at source k and helper h as follows.

• **Helping node** h manages an additional, infinite packet queue for the packets to be forwarded. When h receives a packet from k, h adds it to this queue and notifies k by sending a "preACK" to k. When node h initiates a transmission to the AP, it forms a joint packet consisting of own data from its buffer and data of up to Q packets from the forwarding queue. If there is no collision, the AP successfully decodes the joint packet and sends one "jointACK" to h and all other nodes with data in the joint packet. Node h receives the jointACK

and removes the corresponding packets from the forwarding queue.

• **Source node** k tracks the packet delay at helper h by a state variable p that indicates the number of pending packets. Each time k transmits a packet to h and receives a preACK, it increases p by one. When p passes the maximum number of pending packets P, k directly transmits its current packet to the AP. When k receives a jointACK from the AP, it decreases p by the number of its pending packets that helper h finally forwarded to the AP in the corresponding joint packet.

V. THEORETICAL ANALYSIS

In this section, we derive analytic formulas for the throughput and bit-cost of Direct Link and CoopMAC and show how these are related to the corresponding values of fairMAC. Our derivations are inspired by [12], but we follow the notation in [11].

A. Throughput and Bit-Cost of Direct Link and CoopMAC

The network situation over time can be split into phases. In each phase, the network can either be idle, there can be a successful transmission, or there can be a collision. In the average, one network phase is idle for the time $\bar{t}_i$, it consists in a successful transmission for the time of $\bar{t}_s$, and it consists in a collision for the time of $\bar{t}_c$. We normalized the amount of data of one successful transmission to 1 bit. As a result, both packet durations and slot time σ have to be normalized by the number of bits in a typical packet. For now, this observation is not of further importance, however, we will come back to this observation in Subsection V-B. The probability p_s that one specific node k transmits successfully in a given phase is given by

$$p_s = \tau(1-\tau)^{N-1}. \tag{9}$$

Therefore, the throughput S per node is given by

$$\mathsf{S} = \frac{p_s}{\bar{t}_s + \bar{t}_c + \bar{t}_i}. \tag{10}$$

We now explicitly calculate $\bar{t}_s$, $\bar{t}_i$, $\bar{t}_c$ in (10). The probability of an idle phase is

$$p_i = (1-\tau)^N. \tag{11}$$

The time $\bar{t}_i$ a phase is idle in the average is given by

$$\bar{t}_i = p_i \sigma. \tag{12}$$

The travel time as introduced in Subsection III-A is the duration one packet needs to travel from the source node to the destination. For node k, it is given by $s_k = 1/R_{kh_k} + 1/R_{h_k}$ if k has helper h and it is given by $s_k = 1/R_k$ if k transmits directly to the AP. The average time of successful transmission in one phase is now given by

$$\bar{t}_s = \sum_{k=1}^{N} p_s(s_k + \sigma). \tag{13}$$

Here, we have to add the slot length σ to the travel time s_k since every transmission is followed by an idle slot (no

node will transmit right after an ongoing transmission in CSMA since it first needs to sense an idle slot). It remains to calculate the average collision time $\bar{t}_c$. In CoopMAC, we do not need to consider forwarding transmissions from helpers to the AP, because helping nodes only forward packets if there was no collision in the first hop. Since forwarding happens immediately, there cannot be a collision in the second hop, see Subsection IV-A2). Relevant for the collision time is thus the *packet duration*, which we denote by u_k. If node k transmits via h_k, $u_k = 1/R_{kh_k}$ and if k transmits directly to the AP, $u_k = 1/R_k$. We assume without loss of generality that the set of packet lengths $\{u_k\}_{k=1,\ldots,N}$ is ordered, i.e., $k < l \Rightarrow u_k \le u_l$. The average collision time $\bar{t}_c$ is then given by

$$\bar{t}_c = \sum_{k=2}^{N} \underbrace{\tau(1-\tau)^{N-k}}_{(i)} \sum_{l=1}^{k-1} \underbrace{\binom{k-1}{l} \tau^l (1-\tau)^{k-1-l}}_{(ii)} (u_k + \sigma) \tag{14}$$

where the term (i) is the probability that node k transmits and nodes with packet length larger than u_k (and possibly some nodes with packet length equal to u_k) do not transmit, and where the term (ii) is the probability that exactly l nodes with packet length smaller than or equal to u_k transmit. Using (12), (13), and (14) in (10) allows us to explicitly calculate the average per node throughput S of CSMA based CoopMAC and Direct Link for a given network.

To calculate the bit-cost of node k, we need to differ between two kinds of transmission: first, transmitting own data to the AP (or the helper), and second, forwarding data of other nodes to the AP. As stated above, transmission of own data is involved in collisions and forwarding is not. Therefore, to successfully transmit one own bit, node k has to try τ/p_s times and to successfully forward a received packet, node k only needs to try once. As introduced in Subsection III-A, H_k denotes the number of nodes that get help from node k. In the average, node k forwards H_k packets per own successfully transmitted bit and the resulting bit-cost of node k is

$$\mathsf{B}_k = \left(H_k + \frac{\tau}{p_s}\right) u_k \mathcal{E} \tag{15}$$

where $\mathcal{E}$ is the transmission power, according to our system model from Section II. Note that in Direct Link, $H_k = 0$ for all nodes $k = 1, \ldots, N$ in the network.

B. Throughput and Bit-Cost of fairMAC

We now relate two specific configurations of fairMAC to CoopMAC and Direct Link. First, when the maximum number of pending packets P is finite and the maximum number Q of packets forwarded at a time is equal to zero (no cooperation at all), which we refer to by fairMAC$_0$. Second, when $P = Q = \infty$, which we refer to by fairMAC$_\infty$. The values $P = Q = \infty$ may appear unrealistic, however, as we will see in Section VI, the theoretical behavior of fairMAC$_\infty$ can already be observed for moderate values of P and Q, which makes the investigation of fairMAC$_\infty$ reasonable. We

denote the throughput and bit-cost of CoopMAC and Direct Link as can be calculated by (10) and (15) in the following by S^{coop}, B_k^{coop}, S^{direct}, B_k^{direct}, respectively. The corresponding values for fairMAC$_0$ and fairMAC$_\infty$ are denoted by S^{fair_0}, $B_k^{\text{fair}_0}$ and $S^{\text{fair}\infty}$, $B_k^{\text{fair}\infty}$.

Proposition 1. *fairMAC$_0$ reaches the Direct Link operating point, i.e.,*

$$S^{\text{fair}_0} = S^{\text{direct}} \quad and \quad B_k^{\text{fair}_0} = B_k^{\text{direct}}. \tag{16}$$

Proof: In fairMAC$_0$, nodes that try to transmit via their helper loose their first P packets, since these packets are transmitted to the corresponding helpers but then never forwarded because of $Q = 0$. After that, the number of pending packets is $p = P$ and all nodes will transmit all following packets directly to the AP, which happens exactly according to Direct Link (this can be seen from the protocol descriptions in Section IV by setting $p = P$ and $Q = 0$). On the long term, the impact of the lost P packets onto throughput and bit-cost gets infinitesimal small and the proposition follows. ■

We now prepare for the investigation of fairMAC$_\infty$. The two parameters transmission probability τ and slot length σ are network parameters, which take different values depending on which network setup we consider. We assign to the transmit probability a value $\tau \propto \sqrt{\sigma}$ and let then σ got to zero. Note that it was shown in [12] that $\tau \propto \sqrt{\sigma}$ also holds for that τ that maximizes throughput for a given σ; our assignment is thus reasonable. Since σ is normalized by the number of bits in a packet, letting σ go to zero in our formulas corresponds to letting the packet duration go to infinity in the corresponding real world system.

Proposition 2. *For $\tau \propto \sqrt{\sigma}$ and $\sigma \to 0$, CSMA based CoopMAC and Direct Link perform asymptotically as Round Robin based CoopMAC and Direct Link (see (3)), i.e.,*

$$S^* = \lim_{\sigma \to 0} S = \frac{1}{\sum_{k=1}^{N} s_k} \tag{17}$$

$$B_k^* = \lim_{\sigma \to 0} B_k = (H_k + 1) u_k \mathcal{E} \tag{18}$$

Proof: We write

$$S = \frac{p_s}{\bar{t}_s + \bar{t}_c + \bar{t}_i} = \frac{\tau(1-\tau)^{N-1}}{\bar{t}_s + \bar{t}_c + \bar{t}_i} \tag{19}$$

$$= \frac{(1-\tau)^{N-1}}{\frac{\bar{t}_s}{\tau} + \frac{\bar{t}_c}{\tau} + \frac{\bar{t}_i}{\tau}} \tag{20}$$

As $\tau \propto \sqrt{\sigma}$, $\tau \xrightarrow{\sigma \to 0} 0$ and $\frac{\sigma}{\tau} \xrightarrow{\sigma \to 0} 0$. Using these two limits, it follows through some basic arithmetic operations that for $\sigma \to 0$, the numerator of the right-hand side of (20) converges to 1 and that the denominator of (20) converges to $\frac{1}{\sum_{k=1}^{N} s_k}$. Thus (17) follows. For the bit-cost, we have from (15)

$$B_k = \left(H_k + \frac{\tau}{p_s}\right) u_k \mathcal{E} = \left(H_k + \frac{\tau}{\tau(1-\tau)^{N-1}}\right) u_k \mathcal{E} \tag{21}$$

$$= \left(H_k + \frac{1}{(1-\tau)^{N-1}}\right) u_k \mathcal{E}. \tag{22}$$

The right-hand side of the last line converges to $(H_k + 1) u_k \mathcal{E}$

for $\tau \to 0$. This concludes the proof. ■

To derive throughput and bit-cost expressions for fairMAC$_\infty$, we first calculate the packet lengths v_k in fairMAC$_\infty$. Denote by $\mathcal{D}$ the set of nodes that transmit their packets directly to the AP, denote by $\mathcal{H} \subseteq \mathcal{D}$ the set of nodes that help at least one other node, and denote by $\mathcal{C}$ the set of nodes that transmit their packets via a helper. In fairMAC$_\infty$, the packet lengths v_k of nodes $k \in \mathcal{D} \setminus \mathcal{H}$ are deterministic values given by $v_k = 1/R_k$. Since $P = \infty$, nodes $k \in \mathcal{C}$ will always transmit via their helper, and the packet lengths are also deterministic values given by $v_k = 1/R_{kh_k}$. The packet length v_k for nodes $k \in \mathcal{H}$ are random values given by $(X_k + 1)/R_k$ where X_k is the number of packets in the forwarding queue of node k right before k is transmitting. We can now see that deriving the expression for the throughput $S^{\text{fair}\infty}$ when σ is non-zero is intricate, because packets involved in collisions are of varying length. We defer these calculations to an extended version of this work. However, it can easily be seen that the term corresponding to the right-hand side of (17) is for fairMAC$_\infty$ given by

$$S^{*\text{fair}\infty} = \frac{1}{E[\sum_{k=1}^{\infty} v_k]} \tag{23}$$

Since the random variables $\{X_k\}_{k \in \mathcal{H}}$ are mutually independent, we can exchange summation and expectation in (23). The expectation of X_k is $E[X_k] = H_k$. We get

$$S^{*\text{fair}\infty} = \frac{1}{E\left[\sum_{k=1}^{N} v_k\right]} \tag{24}$$

$$= \frac{1}{\sum_{k \in \mathcal{D} \setminus \mathcal{H}} \frac{1}{R_k} + \sum_{k \in \mathcal{C}} \frac{1}{R_{kh_k}} + \sum_{k \in \mathcal{H}} E[v_k]} \tag{25}$$

$$= \frac{1}{\sum_{k \in \mathcal{D} \setminus \mathcal{H}} \frac{1}{R_k} + \sum_{k \in \mathcal{C}} \frac{1}{R_{kh_k}} + \sum_{k \in \mathcal{H}} (1 + H_k)\frac{1}{R_k}}. \tag{26}$$

We can now reorder the terms and get, continuing from the last line

$$\frac{1}{\sum_{k \in \mathcal{D} \setminus \mathcal{H}} \frac{1}{R_k} + \sum_{k \in \mathcal{C}} \frac{1}{R_{kh_k}} + \sum_{k \in \mathcal{H}} (1 + H_k)\frac{1}{R_k}} \tag{27}$$

$$= \frac{1}{\sum_{k \in \mathcal{D}} \frac{1}{R_k} + \sum_{k \in \mathcal{C}} \frac{1}{R_{kh_k}} + \sum_{k \in \mathcal{H}} H_k \frac{1}{R_k}} \tag{28}$$

$$= \frac{1}{\sum_{k \in \mathcal{D}} \frac{1}{R_k} + \sum_{k \in \mathcal{C}} \left(\frac{1}{R_{kh_k}} + \frac{1}{R_{h_k}}\right)} \tag{29}$$

$$= S^{*\text{coop}}. \tag{30}$$

Since $E[X_k] = H_k$, we can directly express the bit-cost of fairMAC$_\infty$ for $\sigma = 0$ by the packet lengths u_k in CoopMAC as

$$B_k^{*\text{fair}\infty} = E[v_k]\mathcal{E} = \begin{cases} 1/R_k \mathcal{E} = u_k \mathcal{E}, & k \in \mathcal{D} \\ 1/R_{kh_k} \mathcal{E} = u_k \mathcal{E}, & k \in \mathcal{C} \\ \frac{H_k+1}{R_k}\mathcal{E} = (H_k + 1) u_k \mathcal{E}, & k \in \mathcal{H} \end{cases} \tag{31}$$

$$= (H_k + 1)u_k\mathcal{E} \tag{32}$$
$$= \mathsf{B}_k^{*\mathrm{coop}}. \tag{33}$$

To see that equality in (32) holds, remember that $H_k = 0$ if node k does not help any other node. Combining result (30) and result (33) with Proposition 2, we have shown

Proposition 3. *fairMAC$_\infty$ asymptotically reaches the operating point of Round Robin based CoopMAC when $\sigma \to 0$, i.e.,*

$$\mathsf{S}^{*\mathrm{fair}_\infty} = \mathsf{S}^{*\mathrm{coop}} \quad and \quad \mathsf{B}_k^{*\mathrm{fair}_\infty} = \mathsf{B}_k^{*\mathrm{coop}}. \tag{34}$$

VI. NUMERICAL RESULTS AND DISCUSSION

We validate our results from Section V by simulation via an implementation of fairMAC as defined in IV in our custom network simulator written in object-oriented MATLAB. We compare the empirical values for throughput and bit-cost of fairMAC to the theoretical values of CoopMAC and Direct Link as obtained from (10) and (15). The tradeoff curve between these two is calculated by (8). We call it in the following the *timesharing curve*.

We simulate fairMAC for the network from Figure 1 and consider the throughput/bit-cost tradeoff at the potential helper n_3. We let the nodes compete $30\,000$ times for the channel. We set the maximum number P of pending packets constantly equal to $P = 10$ in all simulations. For the number Q of packets forwarded at a time by the helping node n_3, we evaluate fairMAC for the values $Q = 0, 1, 2, 4$. As can be seen in Figure 3, fairMAC reaches for $Q = 0$ the corresponding CSMA Direct Link operating point, both for the network parameters $(\sigma, \tau) = (0.0088, 0.045)$ and $(\sigma, \tau) = (0.0001, 0.0033)$. This validates Proposition 1. For the typical value $\sigma = 0.0088$ (see [11]), the CSMA timesharing curve is far away from the Round Robin timesharing curve. However, for the smaller value $\sigma = 0.0001$, the CSMA curve is close to the Round Robin curve. This validates Proposition 2. For $\sigma = 0.0088$, the operating points for $Q = 2$ and $Q = 4$ of fairMAC are below the corresponding CSMA time sharing curve. This is because in fairMAC, the helping nodes eventually transmit long packets (e.g, for $Q = 4$ helping nodes can forward up to 4 packets at a time). These long packets can be involved in collisions, which is expensive both in terms of throughput and bit-cost. For the smaller value $\sigma = 0.0001$, all fairMAC operating points get close both to the corresponding CSMA timesharing curve and the Round Robin timesharing curve. This validates Proposition 3. We conclude that our distributed protocol fairMAC can asymptotically reach the Round Robin timesharing curve as σ goes to zero, which corresponds in real world to packet lengths going to infinity.

VII. CONCLUSIONS

Under the throughput fairness constraint, we identified a tradeoff between throughput and bit-cost in CSMA based cooperative networks. From the helper node perspective, no cooperation (Direct Link) is optimum in terms of bit-cost while always cooperating (CoopMAC) is optimum in terms of throughput. We proposed the new distributed cooperative
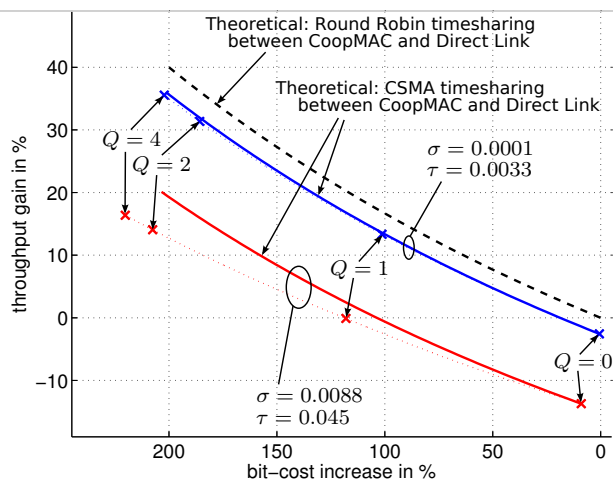

Fig. 3. The throughput/bit cost tradeoff of node n_3 in the network from Figure 1. The reference strategy is for all displayed values the theoretical value for Round Robin Direct Link. For a discussion of the plot, see Section VI.

CSMA protocol fairMAC and showed both theoretically and by simulation that fairMAC can reach the throughput/bit-cost tradeoff curve.

REFERENCES

[1] M. Heusse, F. Rousseau, G. Berger-Sabbatel, and A. Duda, "Performance anomaly of 802.11b," in *Proc. IEEE Conf. IEEE Comput. Commun. Soc. (INFOCOM)*, vol. 2, 2003, pp. 836–843.

[2] T. M. Cover and A. A. E. Gamal, "Capacity theorems for the relay channel," *IEEE Trans. Inf. Theory*, vol. 25, no. 5, pp. 572–584, Sep. 1979.

[3] A. Sendonaris, E. Erkip, and B. Aazhang, "User cooperation diversity—part I: System description," *IEEE Trans. Commun.*, vol. 51, no. 11, pp. 1927–1938, Nov. 2003.

[4] ——, "User cooperation diversity—part II: Implementation aspects and performance analysis," *IEEE Trans. Commun.*, vol. 51, pp. 1939–1948, 2003.

[5] H. Zhu and G. Cao, "rDCF: A relay-enabled medium access control protocol for wireless ad hoc networks," *IEEE Trans. Mobile Comput.*, vol. 5, no. 9, pp. 1201–1214, 2006.

[6] P. Liu, Z. Tao, S. Narayanan, T. Korakis, and S. S. Panwar, "CoopMAC: A cooperative MAC for wireless LANs," *IEEE J. Sel. Areas Commun.*, vol. 25, no. 2, pp. 340–354, Feb. 2007.

[7] P. Liu, Y. Liu, T. Korakis, A. Scaglione, E. Erkip, and S. Panwar, "Cooperative MAC for rate adaptive randomized distributed space-time coding," 2008, pp. 1–6.

[8] T. Korakis, Z. Tao, S. Makda, B. Gitelman, and S. Panwar, "It is better to give than to receive—implications of cooperation in a real environment," in *IFIP Networking*, 2007, pp. 427–438.

[9] S. Narayanan and S. S. Panwar, "To forward or not to forward—that is the question," *Wireless Personal Commun.*, vol. 43, no. 1, pp. 65–87, Oct. 2007.

[10] A. Bletsas, A. Khisti, D. Reed, and A. Lippman, "A simple cooperative diversity method based on network path selection," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 3, pp. 659–672, March 2006.

[11] G. Böcherer, A. de Baynast, and R. Mathar, "Cooperative protocols for random access networks," in *Allerton Conf. Commun., Contr., Comput.*, 2008.

[12] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 3, pp. 535–547, Mar. 2000.