Throughput and Fairness in CSMA/CA Wireless Networks

Hoon Chang
Computer Science Department
Columbia University
New York, NY 10027
Email: hchang@cs.columbia.edu

Vishal Misra
Computer Science Department
Columbia University
New York, NY 10027
Email: misra@cs.columbia.edu

Dan Rubenstein
Electrical Engineering Department
Columbia University
New York, NY 10027
Email: danr@ee.columbia.edu

***Abstract*— While physical layer capture has been observed in real implementations of wireless devices accessing the channel like 802.11, log-utility fair allocation algorithms based on accurate channel models describing the phenomenon have not been developed. In this paper, using a general physical channel model, we develop an allocation algorithm for log-utility fairness. To maximize the aggregate utility, our algorithm determines channel access attempt probabilities of nodes using partial derivatives of the utility. Our algorithm is verified through extended simulations. The results indicate that our algorithm could quickly achieve allocations close to the optimum with 8.6% accuracy error on average.**

I. General Physical Channel Model

For a general analytical model of node throughput and interference, consider first the error probability of transmissions. The successful reception of a transmitted frame is determined through two significant stages. Initially, the frame must be detected by the receiver. Following this, the frame must be successfully received in the presence of interference both from other transmissions and external noise sources. In literature, many models for the two capture stages have been proposed [1], [2], [3].

Given a set of simultaneously transmitting neighbors, we can estimate the error probability of frame transmissions based on a specific channel model. Assume only nodes in a given node set J are transmitting. In Power capture model [2], the detection probability of a transmission from node i is determined by the strength of the intended signal and interference signals at node i's receiver. After detecting the frame, the intended signal is successfully received when the aggregate interference is below a threshold [4].

Rather than relying on specific capture models, we introduce capture function $c_i(J)$. Given a transmitting neighbor node set J, the function computes the successful reception probability of node i's transmitted frames in consideration of interference signals from all nodes in J. Typically, $c_i(J)$ is approximated to an indicator function of J for wireless LANS.

With $c_i(J)$, the success probability of transmissions, q_i is given by:

$$q_i = \sum_{J \subset N_i} c_i(J) \prod_{j \in J} f_j \prod_{k \in N_i - J} (1 - f_k), \qquad (1)$$

where N is the whole node set in the network, $N_i = N - \{i\}$, N is the set of all nodes in the network and f_j is the attempt rate or per-slot transmission probability of node j. Note that even though the number of possible set J is $2^{|N_i|}$, q_i can be approximated by considering subsets of the entire J [5]. Since the capture probability c_i with large-sized J is usually zero, it is accurate enough to take into account only sets including 3 to 4 nodes, on the assumption in [5]. With the approximation, the computation time is a polynomial of $|N|$.

Now, we formulate node throughput. We assume all nodes are within a single sensing range and the nodes are fully synchronized as in [6]. In CSMA/CA fair scheduling protocols, if the channel is idle, nodes access the channel per time slot with probabilities determined by scheduling algorithms like [7]. If any node begins a transmission at a given time slot, the channel is occupied during the transmission time and a new time slot begins after that; otherwise, all nodes hold off their transmissions over a single time slot, which is much shorter than the transmission time. Node throughput in CSMA/CA systems is given by the channel busy probability of each slot as well as the attempt and error probabilities of transmissions. the throughput of node i, x_i is given by:

$$x_i = \frac{f_i q_i T_{TX}}{(1 - p_{TX}) T_{SL} + p_{TX} T_{TX}}, \qquad (2)$$

where p_{TX} is the probability where any nodes in the network transmit at a time slot, which is simply given by $p_{TX} = 1 - \prod_{j \in N} (1 - f_j)$, and T_{SL} and T_{TX} are the length of the time slot and transmission time. Note that Equation 2 is a general throughput formula for random access systems. For slotted-Aloha systems, where T_{SL} equals T_{TX}, Equation 2 exactly yields a formula of node throughput.

II. Obtaining Allocation to Maximize Log Utility

A. Formulation of Optimal Attempt Rate

We develop an algorithm to obtain attempt probabilities to maximize log utility [8] using the fixed-point iteration. Let S be the aggregate log utility. For CSMA/CA systems with a single carrier-sensing ranges, the aggregate utility, S is given by $\sum_i \log\left(\frac{f_i q_i T_{TX}}{(1-p_{TX})T_{SL} + p_{TX} T_{TX}}\right)$. From the concaveness of log utility, the optimal allocation makes all partial derivatives of the aggregate utility zero. Thus, we have the following:

$$\frac{\partial S}{\partial f_i} = \frac{1}{f_i} - \underbrace{\frac{|N|}{G - (1 - f_i)}}_{A} - \underbrace{\frac{|N| - \sum_{j \in N} \frac{q_{j|i}}{q_j}}{1 - f_i}}_{B} = 0, \qquad (3)$$

where G is $\frac{T_{TX}}{(T_{TX}-T_{SL})\prod_{j\neq i}(1-f_j)}$. Note that Part A in Equation 3 is omitted if any attempt rates of the nodes equals to 1 or $T_{TX} = T_{SL}$.

Let a function g return f_i satisfying Equation 3, given f_j for $j \neq i$. Since function g is continuous and maps a rate vector to another rate vector, g has a fixed point (Brouwer's fixed point theorem [9]). We can further show that g converges to the fixed point as an immediate consequence of Brouwer's theorem and the concaveness of log utility.

Now, we formulate function g. Since p_{TX} and q_j depends on f_i, Equation 3 is not easy to solve. To get the function to compute f_i, we first show that Part A in Equation 3 is approximated to a linear function of f_i as follows:

$$\text{Part A} \approx |N|(f_i\mu(\prod_{j\neq i}(1-f_j)) + \nu(\prod_{j\neq i}(1-f_j))). \quad (4)$$

We obtain μ and ν by applying the least squares method. Given f_j ($j \neq i$), we uniformly sample K points from the curve $1/(G-(1-f_i))$, which is a function of f_i, and find a linear function that closely approximates the sampled data to minimize the sum of the squares of the residuals between points generated by the function and corresponding sampled points. The computation time of this approximation is $O(K)$.

Part B in Equation 3 is approximated by:

$$\text{Part B} \approx \sum_j (\theta(q_{j|i}, q_{j|\bar{i}})f_i + \phi(q_{j|i}, q_{j|\bar{i}})). \quad (5)$$

After linear approximation, we have a quadratic formula for f_i from Equation 3 as follows:

$$\frac{1}{f_i} = af_i + b, \quad (6)$$

where $a = |N|\mu(\prod_{j\neq i}(1-f_j)) + \sum_j \theta(q_{j|i}, q_{j|\bar{i}})$ and $b = |N|\nu(\prod_{j\neq i}(1-f_j)) + \sum_j \phi(q_{j|i}, q_{j|\bar{i}})$.

From the quadratic formula, f_i is finally given by:

$$f_i = \begin{cases} \frac{-b+\sqrt{b^2+4a}}{2a} & \text{(if } a > 0\text{) and} \\ \min\left(\frac{1}{b}, 1\right) & \text{(if } a = 0\text{),} \end{cases} \quad (7)$$

where $\min(\infty, 1) = 1$. Assuming q_j and $q_{j|i}$ are known, the total computation time of f_i is $O(K)$.

We show the approximation error is small enough. First, consider Part A in Equation 3. It is easy to see that G in Part A of Equation 3 is greater than 1. A linear approximation to Part A with f_i is prone to larger error as G gets closer to 1. The value of $\prod_{j\neq i}(1-f_j)$ is maximized when all f_j is the minimum. From Equation 3, the possible minimum of f_i is obtained when $q_{j|i} = 0$. Using 802.11 operation parameters to compute T_{TX} and T_{SL}, G is at least 1.18 for all $|N| \geq 2$ sending 512-byte packets at 54 Mbps. For all $|N| \geq 5$, G is more than 1.30. The larger $|N|$ is, the bigger G we have.

As an indicator of how well our linear equation fits, we measure the square of residuals from the original. With $|N| = 2$, the square of the residuals is around 0.875008831 for f_i in the range of 0 to 0.5. When $|N| = 5$, the square value is beyond 0.927489457. For $f_i > 0.5$ and $|N| \geq 2$, the square value is more than 0.976722922. It ranges in value from 0 to 1 and the value 0.927489457 indicates that there is an close correlation in the estimated and actual value of Part A.

To improve the accurate further, we repeat approximation within different intervals of f_i. After the root of Equation 3 is first obtained, we find another best-fitting line for a segment of the curve in an interval around the value of the root. Then, we again compute a new root of f_i satisfying Equation 3. Repeating this process several times, a more accurate root of the formula is obtained.

In approximation for Part B, the square of the residuals is 0.968194017 for f_i in the range of 0 to 0.5. Since $q_{j|\bar{i}}$ and $q_{j|i}$ are not a function of f_i, Part B is continuous between $1/q_{j|i}$ and $1/q_{j|\bar{i}}$. It is trivial that the shorter distance between the two end points, the more like a line the graph looks. The square of the residuals is maximized when $q_{j|\bar{i}}$ and $q_{j|i}$ are farthest away from each other, where $q_{j|\bar{i}} = 1$ and $q_{j|i} = 0$ (because $q_{j|\bar{i}} \geq q_{j|i}$). Note that the optimal value of f_i is typically less than 0.5 in our simulations. However, for $f_i > 0.5$, we can compute accutate μ, ν, θ and ϕ by repeating the approximation with proper intervals.

B. Obtaining Optimal Point

With Equation 7, the fixed-point iteration generates the optimal attempt probability vector. We implement Algorithm 1 in a distributed way. Assuming node i knows neighbors that it interferes with (i.e., nodes whose successful transmission probability is affected by the behavior of i), nodes communicate with the neighbors over wired or wireless links. Note that access points in corporate and residential areas are typically connected to a wired local area network and can often cooperate with each other. The information of neighbors can be entered by the network administrator.

Algorithm 1 Distributed Fair Allocation Algorithm

```
Each node i performs the following steps. The node i
terminates this iteration process when the previous and
updated values of f_i differ within a threshold ϵ
for every round do
    Estimate the channel idle probability (∏_{j≠i} (1 − f_j)).
    Request for each neighbor j to send back the estimated
    error probabilities q_{j|i} and q_j.
    Compute a new attempt probability f_i with the error
    probabilities of the neighbors from Equation 7.
end for
```

In Algorithm 1, each node updates its attempt probability until it converges; nodes stop the process when the difference of the updated probability is less than a threshold. To calculate their attempt probabilities, nodes need to know Compute μ, ν, θ and ϕ that depends on the error probabilities $q_{j|i}$ and q_j of neighbor j and the channel idle probability $\prod_{j\neq i}(1-f_j)$, as in Equation 5.

The channel idle probability $\prod_{j\neq i}(1-f_j)$ can be measured at node i itself. To obtain error probabilities $q_{j|i}$ and q_j, node i only needs to request neighbors whose $q_{j|i}/q_j$ is neither one nor zero. That is, if node j is too far away from i, transmissions

TABLE I

OPERATION PARAMETERS

Physical Layer		Ant. Efficiency	0.8
Frequency	5GHz	Antenna Loss	0.5dB
Path Loss Model	Two Ray	Antenna Height	1.5m
Shadowing Model	Constant	MAC Layer	
Shadowing Mean	4dB	MAC Protocol	CSMA/CA Fair Algorithm
Fading Model	None		
Temperature	290K	Slot Time	9μs
Noise Factor	7	Tx Time	177μs
Tx Power	16dBm	Tx Speed	54Mbps
Rx Sensitivity	-87dB	MAC Header	28 bytes
Antenna	Omni.	ACK Frame	14 bytes

from i do not affect node j and thus $q_{j|i}$ becomes equal to q_j. If node j is close enough, then collisions occur and $q_{j|i} = 0$. Thus, node i requests the limited number of neighbors and all information for rate computation is locally obtained.

$q_{j|i}$ and q_j can be measured by real time. q_j is simply obtained at neighbor j. For $q_{j|i}$, we measure $q_{j|\bar{i}}$. Whenever needed, node i holds off its transmissions and ask the neighbors to measure the successful transmission probability, which is $q_{j|\bar{i}}$. Then, $q_{j|i}$ is computed by the measured $q_{j|\bar{i}}$, q_j and f_i.

III. SIMULATION RESULTS

A. Simulation Parameters

To verify our model and algorithm for log utility fairness, given a set of node pairs, consisting of a sender and receiver, we randomly place them in an area 100×100 m^2. We assume the channel is saturated; senders always have 512-byte data packets to transmit to their associated receivers. The distance between a sender and receiver is selected in a random way but the maximum is set to 30, 10 and 5 meters respectively. 10 different random node placements are considered for each number of node pairs.

Our simulator is based on `Qualnet` simulator, which provides a more accurate and realistic simulation environment than `ns-2` [10], [11]. Table I shows the operation parameters. From locations of nodes, we compute the strength of signals using the two-ray path loss model and the BER (bit error rate) of the received frame is obtained from calculation of `Qualnet`. The transmitter is in half-duplex mode and operating in the 802.11a 5-GHz channel. Receiving an uncorrupted frame, the receiver always responds with an acknowledgment.

MAC protocol is CSMA/CA with a fair algorithm. Assuming all nodes are within a single-carrier sensing range, the nodes are synchronized and channel accesses occur at every slot with attempt probabilities. The attempt probability is determined by a given fair algorithm. The UDP and IP protocols are used for transport and network layer.

B. Optimality Test

The accuracy of our algorithm is compared with the Newton-Raphson method [12], obtaining the optimal attempt rate. The accuracy error is given by the difference divided by the optimum. In total 1,980 rate computations, our accuracy error for attempt probabilities is around 12% on average. For node throughputs, the accuracy error is around 8.6% on average, as shown in Figure 1.

C. Convergence Time

We investigate the convergence time of our algorithm. Our algorithm is run until the difference between the previous and updated attempt rates is less than 10^{-7}. Time is measured in unit of processing rounds. For each maximum distance, we obtain the results varying the number of nodes from 6 to 16.

As shown in Figure 1(c), the convergence time is related to the topology of the network rather than the number of nodes. With the same maximum distance, varying the number of nodes does not significantly change the results. We also plot the convergence time of the Newton-Raphson Method for comparison. The overall convergence time of our algorithms for a median sized network should range within a few seconds with the same assumption in [13].

IV. CONCLUSIONS

In this paper, We show a fair algorithm to determine the attempt rate of nodes. For log-utility fairness, the optimal attempt rate is a function of the interference effect, defined as how much the successful transmission probability of neighbors are degraded by a node. A distributed algorithm is also presented to achieve log-utility fairness. It employs an accurate model to describe the effect of interference.

Our algorithm is implemented in a fully distributed way. Our algorithm runs at each node the fixed-point iteration to obtain the optimal attempt rate. Simulation results show that the convergence time is less then 30 rounds with various numbers of node pairs in a 100×100-m^2 area. The convergence time may be less than a few seconds for a median sized network. As a result of the accuracy, our algorithm achieves around 8.6% of accuracy error for node throughputs on average.

REFERENCES

[1] J. Arnbak, “Capacity of slotted aloha in rayleigh fading channels,” *IEEE Journal On Selected Areas in Communications*, vol. SAC-5, pp. 261–269, February 1987.

[2] D. Davis and S. Gronemeyer, “Performance of slotted aloha random access with delay capture and randomized time of arrival,” *IEEE Transactions on Communications*, vol. COM-28, pp. 703–710, 1980.

[3] K. Cheun and S. Kim, “Joint delay-power capture in spread-spectrum packet radio networks,” *IEEE Transaction on Communications*, vol. 46, no. 4, pp. 450–453, 1998.

[4] T. Rappaport, *Wireless Communications: Principles and Practice*. Prentice Hall, 1996.

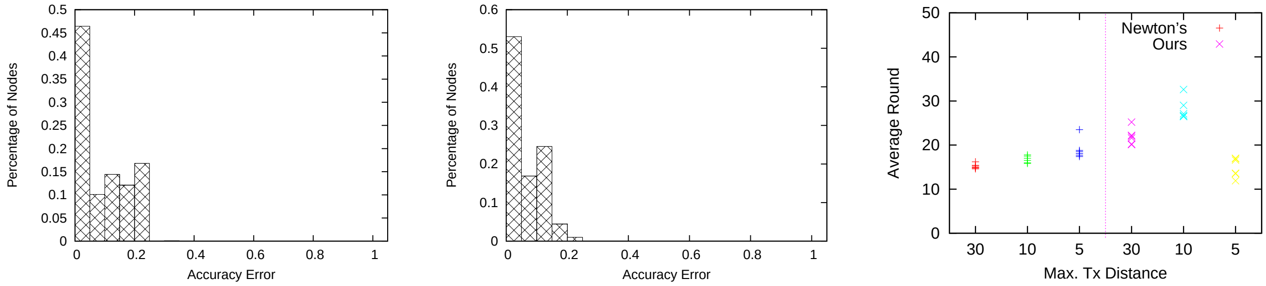
[5] H. Chang, V. Misra, and D. Rubenstein, “A general model and analysis of physical layer capture in 802.11 networks,” in *IEEE INFOCOM 2006*, April 2006.

[6] G. Bianchi, “Performance analysis of the IEEE 802.11 distributed coordination function,” *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 3, 2000.

[7] L. B. Jiang and S. C. Liew, “Proportional fairness in wireless LANs and ad hoc networks,” in *IEEE Wireless Communications and Networking Conference*, vol. 3, March 2005, pp. 1551–1556.

[8] F. Kelly, A. Maulloo, and D. Tan, “Rate control in communication networks: shadow prices, proportional fairness and stability,” in *Journal of the Operational Research Society*, vol. 49, 1998.

[9] W. S. Massey, *Algebraic Topology: An Introduction*. Springer-Verlag, 1977, vol. 56.



(a) Attempt Probability

(b) Node Throughput

(c) Convergence Time

Fig. 1. Accuracy of Assignments and Convergence Time.

[10] Scalable Network Technologies, "Introduction to QualNet," Qualnet User's Manual, 2004.

[11] M. Takai, J. Martin, and R. Bagrodia, "Effects of wireless physical layer modeling in mobile ad hoc networks," in *MobiHoc '01: Proceedings of the 2nd ACM international symposium on Mobile ad hoc networking & computing*. New York, NY, USA: ACM Press, 2001, pp. 87–94.

[12] D. P. Bertsekas, *Nonlinear Programming*, 2nd ed. Athena Scientific Press, 1999.

[13] X. Wang and K. Kar, "Cross-layer rate control for end-to-end proportional fairness in wireless networks with random access," in *MobiHoc '05*. New York, NY, USA: ACM Press, 2005, pp. 157–168.