ITERATIVE COLLISION RESOLUTION IN WIRELESS NETWORKS

An Undergraduate Research Scholars Thesis

by

KATHERINE CHRISTINE STUCKMAN

Submitted to Honors and Undergraduate Research
Texas A&M University
in partial fulfillment of the requirements for the designation as

UNDERGRADUATE RESEARCH SCHOLAR

Approved by
Research Advisor: Dr. Krishna Narayanan

April 2013

Major: Electrical Engineering

TABLE OF CONTENTS

ABSTRACT

Iterative Collision Resolution in Wireless Networks
(April 2013)

Katherine Christine Stuckman
Department of
Electrical and Computer Engineering
Texas A&M University

Research Advisor: Dr. Krishna Narayanan
Department of
Electrical and Computer Engineering

With the growing popularity of smart phones and tablets, development of multimedia applications is on the rise. Speedy transmission of this massive amount information is already pushing the limits of the capacity of wireless networks, and in upcoming years wireless data traffic is projected to continue increasing dramatically. Advances in wireless network throughput are necessary to keep up with society's data demands.

In an uncoordinated wireless communications system, transmissions collide and interfere as multiple users transmit data to a central receiver. Slotted-ALOHA, the conventional method that schedules user transmissions, has only 37% throughput efficiency. However, theoretical results in recent studies suggest that scheduling transmissions over a random number of timeslots and employing iterative collision resolution techniques achieves optimal throughput efficiency of approximately 100%. This research considers how real-world conditions affect these theoretical results. A MATLAB model was developed to create random graphs, representing users

transmitting packets over such timeslots, and the packets were resolved by this method. This model was simulated extensively, representing networks of up to 10,000 packets over 10,000 to 20,000 timeslots, and the number of packets resolved in each iteration was measured.

These simulations have generated empirical data that backs up the theoretical claim. Furthermore, by implementing low-complexity matrix algebra, an even greater percentage of successful trials can be obtained to further increase efficiency. These results demonstrate the potential of this method in handling uncoordinated transmissions in communications systems, even in the presence of finite conditions. This suggests that this method could eventually be employed in actual wireless systems.

NOMENCLATURE

$f_X(x)$	PDF defining probability that single user transmits in x timeslots
$P_l[v]$	Probability that the left-to-right message on a randomly chosen edge is an erasure in iteration l
$P_{l,i}[v]$	Probability that the left-to-right message on a randomly chosen edge, connected to a user node of degree i, is an erasure in iteration l
$P_l[u]$	Probability that the right-to-left message on a randomly chosen edge is an erasure in iteration l
$P_{l,i}[u]$	Probability that the right-to-left message on a randomly chosen edge, connected to a timeslot node of degree i, is an erasure in iteration l
λ_i	Fraction of edges connected to user nodes of degree i
ρ_i	Fraction of edges connected to timeslot nodes of degree i
ϵ	Probability of a single erasure
K	Number of users
M	Number of timeslots
V	MATLAB data structure containing data for K users
C	MATLAB data structure containing data for C timeslots
p	Empirical percentage of successes in successive trials

CHAPTER I

INTRODUCTION

Today's wireless technologies are struggling to keep up with the demand for real-time, multimedia data transmission. As data traffic continues to increase and congest wireless networks, it becomes increasingly urgent to research and implement new technologies that increase wireless capacity.

This research addressed the problem of multiple uncoordinated users attempting to transmit data to a central receiver in a wireless communication system. Since the users are uncoordinated, packets transmitted simultaneously can collide, causing interference to each other. Historically, communications services turn to a standard slotted ALOHA scheme, which outlines a random access protocol. In slotted ALOHA, each user transmits their data after waiting a random interval of time, with no synchronization with the other users. Because the users are not coordinated, this method often results in interference between users sending their packets simultaneously. When these collisions occur, the slotted ALOHA technique calls for the users to resend their data. This decreases the throughput and results in a throughput efficiency of $1/e = 0.37$ [1]. Despite this low efficiency, slotted ALOHA is used widely in wireless network communications by mobile phone, satellite, and Wi-Fi [1]. Especially when considering the importance of such communications networks in today's society, a more efficient protocol is desirable.

In [2], Narayanan and Pfister have proposed a method detailing a protocol in which users send their packet a number of times, as defined by a derived probability distribution, at random time intervals. A user will send a packet in x timeslots with probability $f_{\mathrm{X}}(x) = \frac{1}{X(x-1)}$ for $x = 2{,}3 \ldots \infty$.

This technique utilizes iterative collision resolution: instead of requiring users to resend their data in the case of interference, the central receiver can resolve all packets based on the collective information gathered over all the timeslots. Like slotted ALOHA, this algorithm does not require coordination between users. Despite this, use of the proposed probability distribution in this method can attain an efficiency very close to 1 when the number of users becomes asymptotically large. Even when considering the case where all users coordinate which packets to send during their timeslots, this is the upper-bound on the achievable efficiency.

Though this result is optimal, this is an asymptotic result that requires the number of packets and timeslots to be infinitely large. In this thesis, we consider the practical case, where the number of packets and timeslots are finite. Addressing these effects is critical before the algorithm can be considered for real-world application.

CHAPTER II

METHODS

A single trial can be simulated by generating a random Tanner graph representing K users sending packets over M time slots, as shown in Figure 1. In such a trial, each user is assigned a number of time slots, as defined by the probability distribution, and each time slot is chosen according to a uniform distribution.

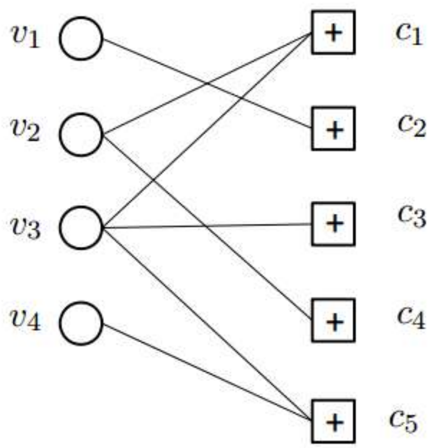

Figure 1 (Narayanan, Pfister)

Recall that the derived distribution considers the domain for possible number of timeslots a user can transmit to is $x = 2,3 \ldots \infty$. However, for a finite number of timeslots M, the domain becomes $x = 2,3 \ldots M$. To satisfy probability law by ensuring all probabilities sum to 1, the distribution must be normalized as a PMF. Doing so requires solving for a normalization coefficient C, resulting in the following normalized distribution for the number of transmission timeslots for a single user: $f_X(x) = \frac{A}{x(x-1)}$ for $x = 2,3 \ldots M$.

The iterative collision resolution algorithm involves iterations through all timeslots, checking the degree at each timeslot node. If the degree is 1, the single packet sent in this timeslot is known, so it can be effectively “subtracted” every time it appears in other timeslots. This is done by removing all edges connected to this packet, which will ideally result in more timeslots having the degree of user packets at value 1, thus allowing further packets to be resolved in subsequent iterations. If every user packet can be removed from timeslot user arrays, thus resulting in a degree of 0 for every timeslot, the resolution of the graphs is successful, indicating that all packets can be resolved in the particular Tanner graph generated.

By deriving density resolution for this scenario, the theoretical behavior of this model for finite K and M can be better understood. Based upon the density evolution for Low-Density Parity Check (LDPC) codes, a similar recursion can be derived for this model.

Asymptotic Analysis using Density Evolution

In the Tanner graph in Figure 1, let λ_i be the fraction of edges connected to user nodes of degree i, and let ρ_i be the fraction of edges connected to timeslot nodes of degree i. Given $f_{\mathrm{X}}(x)$, K, and M, these fractions can be determined.

λ_i can be determined directly from $f_{\mathrm{X}}(x)$ since this PMF gives the distribution of user node degrees. The fraction of user nodes that are of degree i is simply $f_{\mathrm{X}}(i)$. Since the distribution is identical for all K user nodes, the expected number of degree i user nodes in the graph is $Kf_{\mathrm{X}}(i)$, and with each of these nodes producing i edges, the expected number of edges in the graph that are connected to a user node of degree i is $Kif_{\mathrm{X}}(i)$. The expected number of edges produced by a node is $\sum_{j=2}^{M} jf_{\mathrm{X}}(j)$, so the expected number of edges in a entire graph of K users is $K\sum_{j=2}^{M} jf_{\mathrm{X}}(j)$. Therefore, the expected fraction of edges in the graph that are connected to a user node of degree i is $\lambda_i = (\frac{1}{(i-1)}) \,/\, (\sum_{j=2}^{M} \frac{1}{(j-1)})$.

Finding ρ_i requires this λ_i and the values of K and M. If $g_{\mathrm{X}}(x)$ is the PMF for the probability that a timeslot node has x incoming transmissions, then the expected total number of edges in the graph is $M\sum_{j=2}^{K} jg_{\mathrm{X}}(j)$. The expression $K\sum_{j=2}^{M} jf_{\mathrm{X}}(j)$ derived above is also the total number of edges in the graph, so $M\sum_{j=2}^{K} jg_{\mathrm{X}}(j) = K\sum_{j=2}^{M} jf_{\mathrm{X}}(j)$. This means that the average degree of a timeslot node is $\sum_{j=2}^{K} jg_{\mathrm{X}}(j) = \frac{K}{M}\sum_{j=2}^{M} jf_{\mathrm{X}}(j)$. In [2], this average degree of a timeslot node is denoted r_{avg}. Furthermore, [2] shows that the number of edges connected to a timeslot node of degree i is given by the distribution $\rho(x) = \mathrm{e}^{-(1-x)r_{avg}}$. Substituting the value for r_{avg} obtained above results in $\rho(x) = \mathrm{e}^{-(1-x)\frac{K}{M}\sum_{j=2}^{M} jf_{\mathrm{X}}(j)} = \rho(x) = \mathrm{e}^{-(1-x)\frac{K}{M}(\sum_{j=2}^{M}\frac{1}{(j-1)})}$. Finding the expression for ρ_i requires expanding $\rho(x)$ into its series representation:

$\rho(x) = \mathrm{e}^{-\frac{K}{M}(\sum_{j=2}^{M}\frac{1}{(j-1)})}[1 + \frac{K}{M}\,(\sum_{j=2}^{M}\frac{1}{(j-1)})x + \frac{\frac{K}{M}(\sum_{j=2}^{M}\frac{1}{(j-1)})^{2}}{2!}\mathrm{x}^2 + \,\dots]$. Extracting the ρ_i values results in $\rho_i = \mathrm{e}^{-\frac{K}{M}(\sum_{j=2}^{M}\frac{1}{(j-1)})}\frac{\frac{K}{M}(\sum_{j=2}^{M}\frac{1}{(j-1)})^{\mathrm{i}}}{\mathrm{i}!}$.

From here, LDPC coding theory can be used to determine the density evolution properties of this model [3]. Like the recursive probabilities of erasure in LDPC codes, let $P_l[v]$ be the probability that the left-to-right message on a randomly chosen edge is an erasure in iteration l, and let $P_{l,i}[v]$ be this same probability, for an edge connected to a user node of degree i. Similarly, let

$P_l[u]$ and $P_{l,i}[u]$ represent these probabilities for messages from right-to-left. ϵ is the probability of a single erasure.

We will consider a more general model where the timeslots can be erased with probability ϵ. The probability of no erasure on an edge connected to a node with degree i to be: $1 - P_{l,i}[u] = (1-\epsilon)(1 - P_l[v])^{\,i-1}$. This means that probability of erasure is $P_{l,i}[u] = 1 - (1 - \epsilon)(1 - P_l[v])^{i-1}$. By summing these probabilities, weighted by the probability of a random edge connected to a node of degree i, probability erasure in any node is $P_l[u] = \sum_i \rho_i P_{l,i}[u]$. Similarly in the next iteration, $P_{l+1,i}[v] = (P_l[u])^{i-1}$ and $P_l[v] = \sum_i \lambda_i P_{l,i}[v]$.

By substituting these values, we find:

$P_{l+1}[u] = \sum_i \rho_i P_{l+1,i}[u]$

$P_{l+1}[u] = \sum_i \rho_i (1 - (1-\epsilon)(1 - P_{l+1}[v])^{i-1})$

$P_{l+1}[u] = \sum_i \rho_i (1 - (1-\epsilon)\left(1 - \sum_j \lambda_j P_{l+1,j}[v]\right)^{i-1})$

$P_{l+1}[u] = \sum_i \rho_i (1 - (1-\epsilon)\left(1 - \sum_j \lambda_j (P_l[u])^{j-1}\right)^{i-1})$

Because this is not a model for an LDPC code, we can set the probability of erasure ϵ as 0. This resolved the equation above to $P_{l+1}[u] = \sum_i \rho_i (1 - \left(1 - \sum_j \lambda_j (P_l[u])^{j-1}\right)^{i-1})$.

By substituting the known values of λ_j and ρ_i derived previously, the recursion can be used to solve for $1 - P[u]$, the steady state probability of no erasure. Figure 2 shows the how the recursion converges over several iterations.

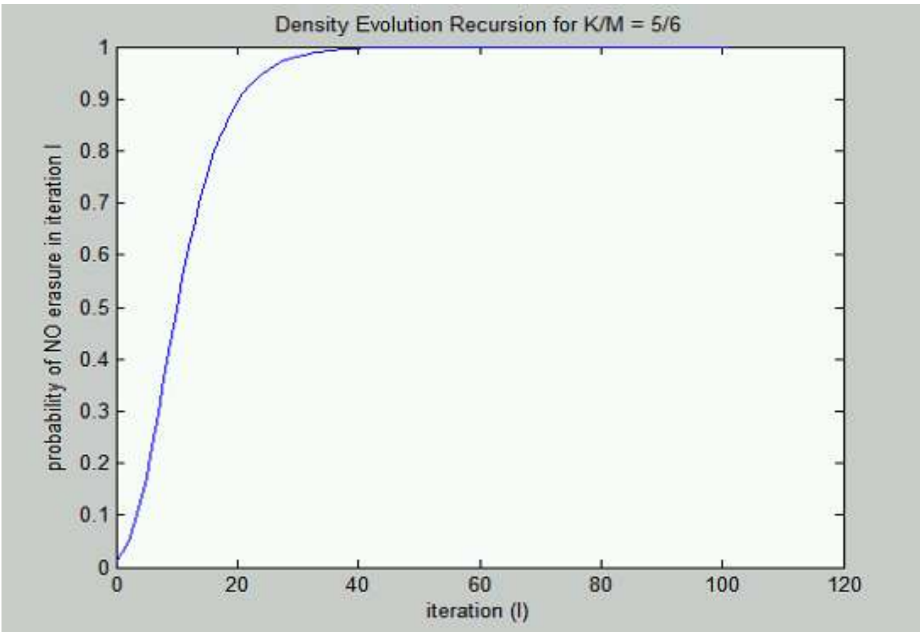

Figure 2

Figure 3 shows how the steady state probability of no erasure $1 - P[u]$ changes for varying K/M. This plot shows that probability of no erasure converges to 1 for $K/M \leq 1$.

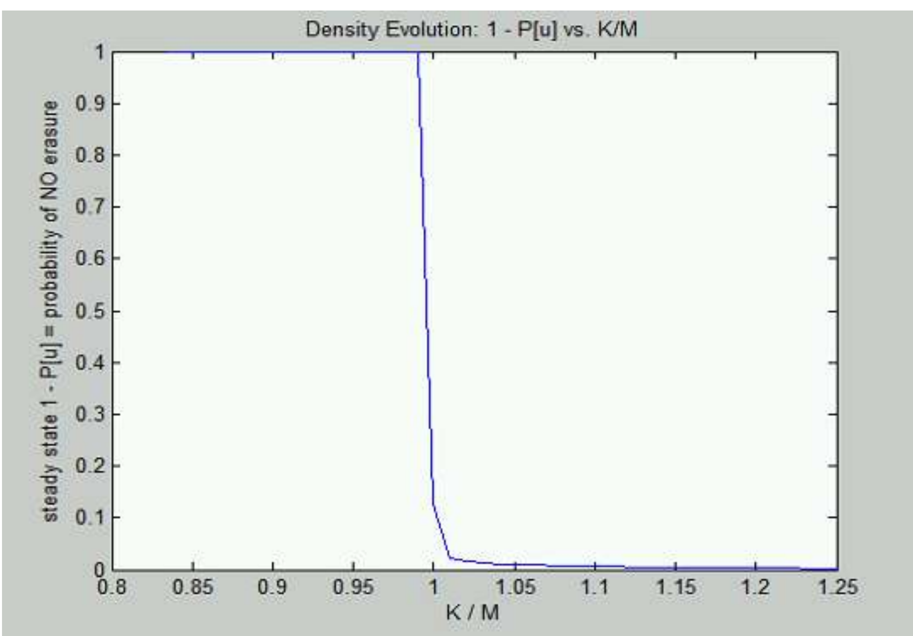

Figure 3

This density evolution gives the theoretical basis for this problem. The MATLAB simulations described next give empirical data to back up theoretical claims. The optimality in the theoretical case comes with the stipulation that the number of users K approaches infinity. Simulations were necessary to determine how the algorithm will behave with a finite number of users.

Simulations for Finite Lengths

A simulation of a single trial can be modeled by a tanner graph. In MATLAB, this graph is expressed as two data structures. Data structure V contains K MATLAB structs. Each of these structs represents a single user, storing (a) the degree, an integer signifying the number of timeslots for transmission, and (b) an array of these timeslots, each timeslot an integer value, representing the index of this user in struct C. Similarly, data structure C contains M MATLAB structs. Each of these structs represents a timeslot, storing (a) the degree, and integer signifying the number of users transmitting in this timeslot, and (b) an array of users, each user an integer value representing the index of this user in struct V.

Once this Tanner graph is generated, the trial must be deemed a success or failure. The graph undergoes the iterative interference cancellation algorithm described previously. This involves iterations and modifications to the V and C data structures as defined by the algorithm, deleting edges in the structures to indicate that a packet has been resolved. If the algorithm is successful in resolving all packets, the trial is deemed a success.

In these simulations, some of the mappings generated may be irresolvable: A trial will be deemed a failure if not all packets could be properly recovered. For a large number of trials with set value K and M, let p be the empirical fraction of successes.

First, the number of packets M is set to a range of values to model the relationship M vs. p. The ratio K/M necessary for a $p \approx 1$ is expected to converge to 1 as K and M increase. Attaining the desired efficiency of 1 requires that $M = K$.

An appropriate number for K in these trials is 10,000, with values of M ranging from 10,000 – 20,000. The first question was how many entire graphs are resolved for these varying values of M. This was simulated by simulating creation and resolution of random graphs. Next the average number of individual packets recovered in each trial was considered. Finally, simulations determined the distribution of the number of packets resolved was determined, and specifically, the number of graphs either (1) almost entirely resolved, with only 0.01% of packets remaining, and (2) the number of graphs that were highly unsuccessful, with only 0.01% of packets resolved.

For values of M with a high percentage of "Terrific" graphs, most graphs that are not completely resolved have less than 10 packets remaining. The remaining timeslots can be viewed as a manageable, binary system of equations, with the number of unknowns equal to the number of remaining packets. For some of these systems of equations, matrix algebra under binary operations allows the rest of the packets to be resolved. If the rank of the binary matrix is greater than or equal to the number of remaining packets, the system can be resolved and all packets can be recovered. Such a test was simulated in MATLAB for “Terrific” packets to determine if a higher percentage of graphs could be entirely resolved.

After the results of the binary matrix resolution tests proved unsuccessful for the large graphs, another factor was considered. In real-world applications, there will be a gain that factors into each packets signal. This gain can be thought of as a complex coefficient for each packet in the systems of equations for each timeslot. If this gain is known, the system can be solved as before, except using complex matrix algebra instead of binary algebra. Added to this complex gain is noise in the channel. In this study, the noise is ignored, so the resulting efficiency is an optimistic estimate.

CHAPTER III

RESULTS

Running trials for K = 10,000 users yielded increasing percentages of entire graphs resolved. However, some of these unresolved graphs may have had only a small fraction of packets remaining: even with the graph unresolved as a whole, several of the users may have had successful packet transmissions. The plot below considers the percentage of entire packets resolved, denoted in blue. It also denotes the fraction of user packets, out of K = 10,000 total packets, resolved in each trial, marked in red.

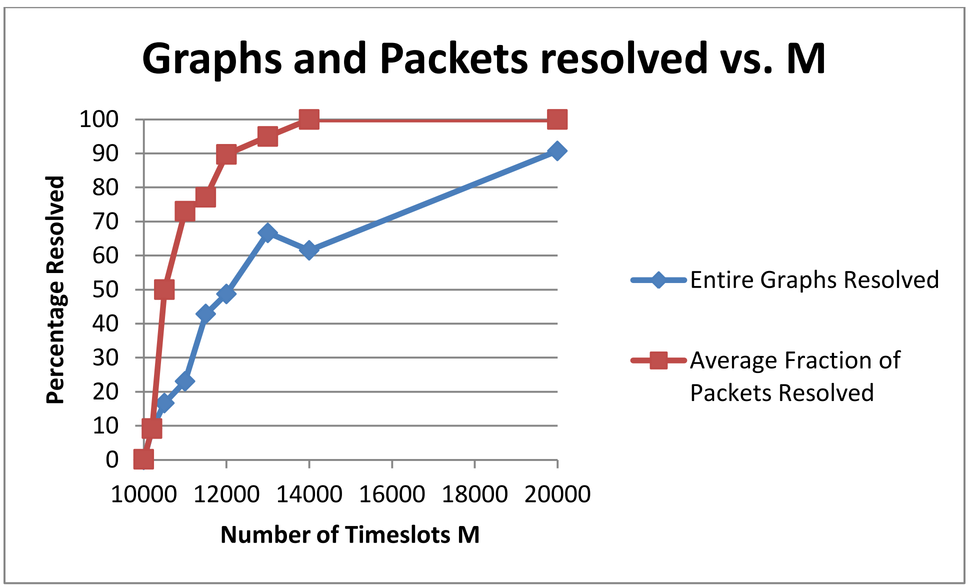

Figure 4

These results led to a fascination of the distribution of results in Figure 4, particularly how many trials resolved either almost all packets or alarmingly few packets. Below, the number of trials with at least 99.9% of packets resolved ("Terrific Graphs") and the number of trials with fewer than 0.01% of packets resolved ("Terrible Graphs") are displayed.

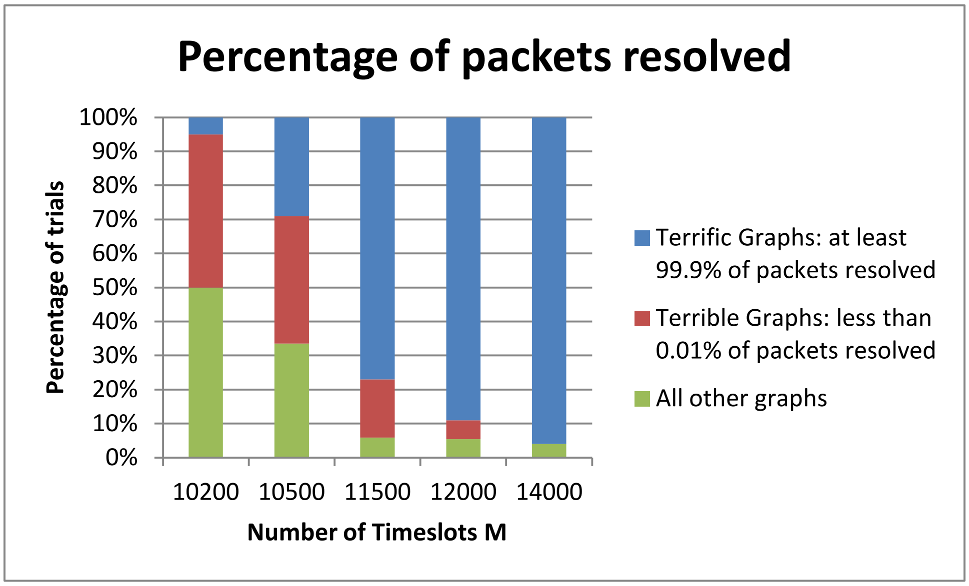

Figure 5

Results show that a significant number of trials fell into one of these two categories as M increases. Additionally, the number of "Terrific" graphs outweighs the number of "Terrible" graphs as M increases.

Next, the use of Matrix resolution was evaluated. Matrix resolution in large graphs was largely unsuccessful. However, very small graphs were evaluated as well, and many matrices could be resolved with binary matrix algebra. By altering graph sizes while keeping a constant K / M ratio, the success of the matrix resolution is shown in Figure 6.

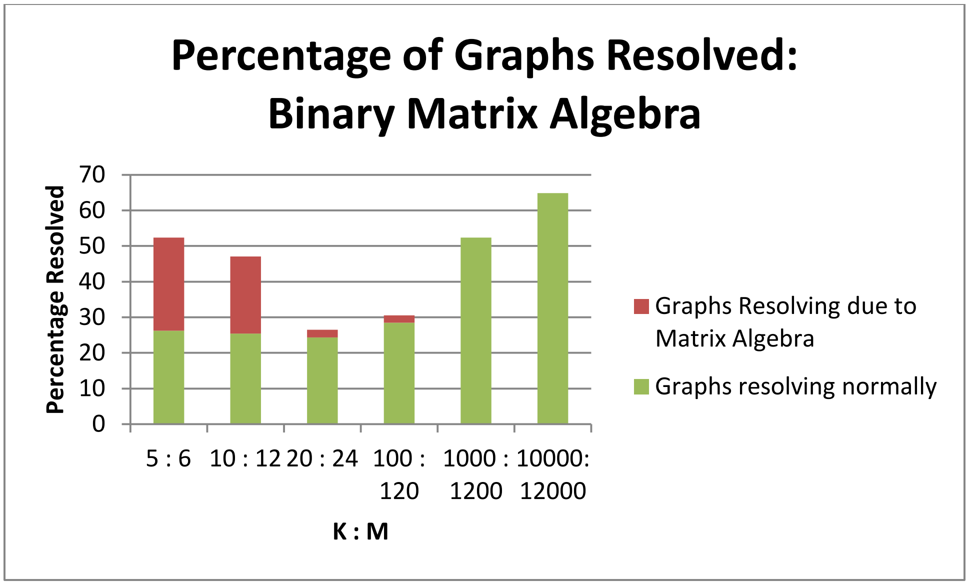

Figure 6

This graph illustrates that, while increasing the graph size improves the percentage of working iterative collision resolution trials, the use of matrix algebra in the small graphs drastically improves overall performance.

After considering the complex gain in the channel, the matrix resolution was performed again on a complex matrix rather than a binary matrix. The results of these trials are shown in Figure 7.

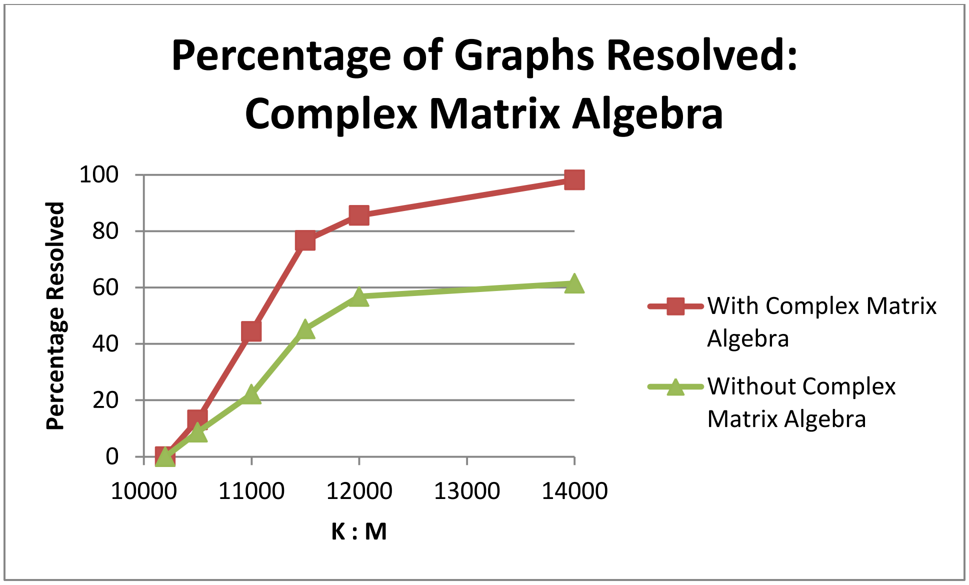

Figure 7

The effectiveness of the complex matrix resolution increases the percentage of completely resolved trials, and since these matrix inversions are performed on matrices no greater in size than 10 by 10, there is also low cost in complexity. Implementing matrix resolution by accounting for the known gains in the system can drastically improve the efficiency of this method.

However, the complex gain will be corrupt by some noise in the channel, and this model does not account for the coding schemes necessary to remove noise from the signal. This means that the simulation shown above is very optimistic, and implementing the channel codes necessary will also affect the efficiency of this model.

CONCLUSIONS

The data above shows that, in a network of K = 10,000 users, efficiency of approximately 10,000/14,000 = 0.71 is achieved. Recall that this achievable efficiency is much higher than the 37% of the Slotted-ALOHA used in existing systems. Furthermore, increasing K to larger values is expected to further increase efficiency. In general, further research and eventual implementation of this method could indeed drastically increase the throughput of wireless networks.

Additionally, the use of simple matrix algebra drastically increases the success of trials by approximately 30% when assuming no noise. Adding this simple mechanism to the existing model has drastic effects on the efficiency of the method, as these successful graphs would not require that any packets be resent in subsequent trials.

Additionally, in future work, the number of users K can be increased to 100,000 and 1,000,000. The same trials can be duplicated and improvement of the algorithm based upon this increase can be measured. This work will assist in determining an appropriate lower-bound for K such that $K/M \approx 1$ while maintaining $p \approx 1$.

Considering packet resolutions over multiple, consecutive trials may offer a way to further increase efficiency. Unresolved packets could be resent with a modified probability distribution that increases the number of transmissions. This requires more theoretical investigation to derive a modified distribution for resent packets, while upholding the theoretical efficiency of 1 by also modifying the distribution for all other users. This would increase the efficiency of the system, and would be a practical addition to this method in implementation.

Additionally, other effects of the physical layer can be further explored. As mentioned when exploring the complex signal gain, the effects of a noisy channel were not considered when defining this optimal transmission policy. Efficient coding and decoding methods should be used to correct errors caused by noise in the channel. This will require a joint design of the probability distribution and the error correction code. The measures of performance will be the amount of energy used and the complexity of the error correcting codes. By considering the packets received over multiple timeslots jointly, the central receiver may be able to use this collective information to correct errors. This method may allow for error correction techniques requiring less redundancy and less energy, than those required when considering each timeslot individually.

.

In conclusion, additional research into the practical limitations of this method must be conducted to fully understand its potential impact on wireless networks. Preliminary results uphold the theoretical promise of this method, and further investigation could increase wireless network capacity and thus hold immense value to today's society.

REFERENCES

[1] Schwartz, M.; Abramson, N. (2009). The Alohanet - surfing for wireless data [History of Communications]. Communications Magazine, IEEE , vol.47, no.12, pp.21,25.

[2] Narayanan, K.R.; Pfister, H.D. (2012). Iterative collision resolution for slotted ALOHA: An optimal uncoordinated transmission policy. Turbo Codes and Iterative Information Processing (ISTC), 2012 7th International Symposium on, 136,139, 27-31.

[3] Ryan, William, and Shu Lin (2009). Channel Codes: Classical and Modern. Cambridge: Cambridge UP. 201-207, 388-394.