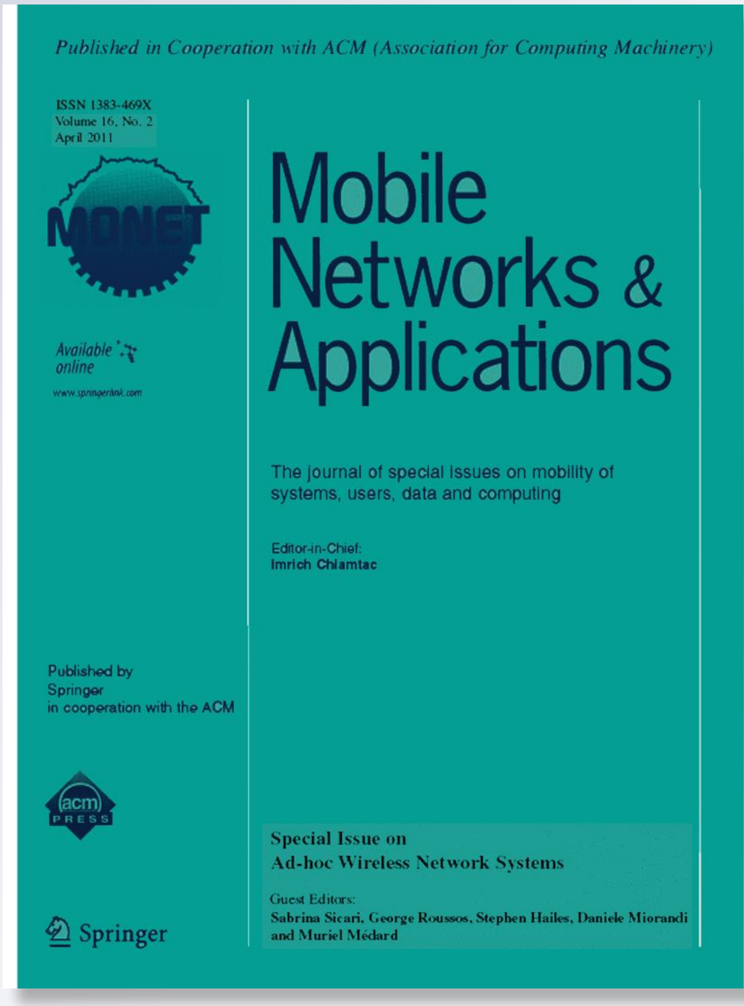
Mathematical Analysis of Throughput Bounds in Random Access with ZigZag Decoding

Mobile Networks and Applications
The Journal of SPECIAL ISSUES on Mobility of Systems, Users, Data and Computing

ISSN 1383-469X
Volume 16
Number 2

Mobile Netw Appl (2010)
16:255-266
DOI 10.1007/s11036-010-0290-2



Springer

Mobile Netw Appl (2011) 16:255–266
DOI 10.1007/s11036-010-0290-2

Mathematical Analysis of Throughput Bounds in Random Access with ZigZag Decoding

Jeongyeup Paek · Michael J. Neely

Published online: 17 December 2010

Abstract We investigate the throughput improvement that ZigZag decoding (Gollakota and Katabi 2008) can achieve in multi-user random access systems. ZigZag is a recently proposed 802.11 receiver design that allows successful reception of packets despite collision. Thus, the maximum achievable throughput of a wireless LAN can be significantly improved by using ZigZag decoding. We analyze the throughput bounds in four different idealized multi-access system models for the case when ZigZag decoding is used. We also provide results for the Aloha and CSMA models where exact closed form solutions are infeasible to calculate. Our analysis and simulation results show that ZigZag decoding can significantly improve the maximum throughput of the random access system.

An earlier version of this paper [12] appeared in the Proceedings of The 7th International ICST Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt 2009).

This material is supported in part by one or more of the following: the DARPA IT-MANET program grant W911NF-07-0028, the NSF grant OCE 0520324, the NSF Career grant CCF-0747525.

J. Paek (✉)
Department of Computer Science,
University of Southern California,
3710 S. McClintock Avenue, RTH 418,
Los Angeles, CA 90089, USA
e-mail: jpaek@usc.edu

M. J. Neely
Department of Electrical Engineering—Systems Division,
University of Southern California, 3740 McClintock Ave.,
EEB 520, Los Angeles, CA 90089-2565, USA
e-mail: mjneely@usc.edu

Keywords ZigZag decoding · random access · throughput analysis · CSMA

1 Introduction

In this paper, we investigate how much throughput improvement ZigZag decoding [5] can achieve. ZigZag is a recently proposed 802.11 receiver design that combats hidden terminals. In ZigZag, the receiver can decode two consecutive signals of two colliding packets and successfully receive both. In other words, if the same two packets collide twice (with a small bit offset difference), the receiver can receive both packets. ZigZag decoding is based on the observation that when two 802.11 senders collide, their retransmissions tend to collide again on same packets, and since 802.11 senders jitter every transmission by a short random interval, collisions start with a small random stretch of interference-free bits. Thus, the maximum achievable throughput of a wireless LAN can be significantly improved by using ZigZag decoding.

We look at four different idealized multi-access system models to investigate the performance benefits of ZigZag decoding; (1) N-user slotted random access system, (2) stabilized slotted Aloha, (3) slotted CSMA with mini-slots, and (4) unslotted CSMA. Prior work has already analyzed these simple systems and has given throughput bounds (see, for example, [2, 4] for summaries and reviews of prior work). We extend these well known results to the case when ZigZag decoding is used. We also provide results for the Aloha [1] and CSMA [7] models where exact closed-form solutions are infeasible to calculate. We concentrate on simple random access protocols and do not consider conflict resolution techniques such as [3, 8] or capture effects (e.g. [9, 10]) for the simplicity of analysis. Related recent work in [11] considers random Aloha-type protocols in networks with "soft collisions" and multi-packet reception capabilities (such as CDMA systems) but the CDMA approach is orthogonal to ZigZag decoding that we consider in the present paper.

Most of the classical assumptions for idealized slotted multi-access models used in prior work also hold in our work, unless otherwise stated. For example, our models have slotted time with slot boundaries at integer times ($t \in \{0, 1, 2, ...\}$), packets have fixed size and their transmission time equals exactly one slot, and if just one node sends a packet in a given slot, the packet is correctly decoded at the receiver. Also, we assume that each packet involved in a collision is retransmitted, until the packet gets successfully received. A node with a packet that must be retransmitted is said to be backlogged. We remove the slot boundary restriction later in Section 5 when we discuss the unslotted case.

However, to apply ZigZag into our analysis, we redefine the term *'Collision'* and make the following simplified assumptions for the slotted models: 'Collision' occurs on a slot when three or more users attempt transmission in a given time slot. In this case, no packets are delivered to the receiver. If exactly two users transmit packets in a slot, we say that this is a 'ZigZag' case which is decodable using ZigZag decoding.[1] Either an 'Idle,' 'Success,' 'ZigZag,' or 'Collision' event happens on every slot, (corresponding to whether 0, 1, 2, or more than 2 packets were transmitted in that slot, respectively) and this feedback is explicitly given ('0,' '1,' 'ZigZag,' 'C') to all users at the end of each slot. If a 'ZigZag' event occurs, that slot is automatically extended into two slots. The two colliding users know that they have collided in a ZigZag event, and retransmit the same packet in the next slot.[2] Other users also know about this and thus remain silent in the next slot. If a 'ZigZag' event occurs, exactly 2 packets can be perfectly received at the receiver during two time slots.[3] Hence the average throughput during this period is 1 packet/slot. In the unslotted case, everything remains the same except that the slot boundaries are not synchronized and may occur any time on the continuous time axis. Finally, we ignore other aspects of ZigZag decoding such as decoding failure or 3 packet decoding.[4]

While we use a slotted model, it is important to recall that ZigZag decoding relies on packet transmissions to arrive at the receiver so that the first bits of each packet are non-aligned. This is consistent with a slotted model if slot sizes are several bits larger than packet sizes, and if transmissions are randomized with respect to the initial bit alignment. A general model allows for some probability p_f for a ZigZag frame failure due to accidental perfect alignment. However, in this paper, we treat the ideal case of $p_f = 0$ for the sake of simplicity.[5]

2 Slotted random access

We first look at a simple idealized random access protocol for a N-user system, where each user has an infinite amount of data to send. In this protocol, we use ZigZag decoding as follows. The timeline is decomposed into *frames* of size either one or two slots, and these frames define renewal events (see Fig. 1). At the beginning of a frame, all N users independently transmit a packet with probability q ($0 \leq q \leq 1$), and exactly one of the following four events happen; (1) *Idle*: nobody transmits any packet, (2) *Success*: exactly one user transmits a packet, (3) ZigZag: exactly two users transmit a packet, or (4) *Collision*: three or more users attempt transmission. Then the receiver

[1] Gollakota and Katabi [5] describes the various collision patterns that the ZigZag applies to; overlapped collisions, flipped order, different packet sizes, capture effect, etc. We abstract all of these patterns into two packets being transmitted within two slots with small asynchrony across successive collisions for simplicity of analysis.

[2] ZigZag decoding is based on the observation that when two 802.11 senders collide, their retransmissions tend to collide again on same packets, and since 802.11 senders jitter every transmission by a short random interval, collisions start with a small random stretch of interference free bits. We assume that this random time interval is sufficiently small enough to be included within a slot along with the packet with negligible effect to the analysis.

[3] Gollakota and Katabi [5] describes the success rate of ZigZag decoding. Its collision detector has only 3.1% false positives, 1.9% false negatives, and 98.2% of the colliding packets are decodable when all of its components are enabled. We simplify the analysis by using 0%, 0%, and 100% for each of these ratios respectively.

[4] In a real system, there could be cases where ZigZag decoding may fail for some practical reasons such as insufficient SNR. Also, there could be cases where three or more colliding packets are decodable. Gollakota and Katabi [5] does describe these cases but we ignore them for simplicity of analysis.

[5] As we have mentioned in footnote 3, ZigZag decoding was able to decode 98.2% of the colliding packets in a real experiment, which corresponds do $p_f = 1.8\% \approx 0$.

Fig. 1 A timeline showing the various kinds of frames '0', '1', 'C', 'ZigZag'

gives one of the four following feedback messages at the end of the first slot of the frame:

$$\text{Feedback} = \begin{cases} \text{'0'} & \text{if idle (i.e., no user attempted transmission).} \\ \text{'1'} & \text{if success (i.e., exactly one user attempted transmission).} \\ \text{'ZigZag'} & \text{if exactly two users attempted transmission.} \\ \text{'C'} & \text{if three or more users attempted transmission.} \end{cases}$$

The frame has size one slot if the feedback is '0,' '1,' or 'C.' If the feedback is 'ZigZag,' then the frame has a size of two slots. Recall that in the second slot of a 'ZigZag' frame, the same two users transmit their packets again while all other users remain silent. Thus, in a 'ZigZag' frame, exactly two packets are successfully transmitted by the end of the two-slot frame.

In this system, the probability of '1' (success) and 'ZigZag' is given as follows:

$$P_1 = \binom{N}{1} q(1-q)^{N-1} = Nq(1-q)^{N-1}, \tag{1}$$

$$P_{\text{zigzag}} = \binom{N}{2} q^2(1-q)^{N-2} = \frac{N(N-1)}{2} q^2(1-q)^{N-2}. \tag{2}$$

Also, the average frame size of the system, and the average number of packets successfully delivered to the receiver in a frame is given by:

$$\begin{aligned} E\{\text{frame size}\} &= 2 \cdot P_{\text{zigzag}} + 1 \cdot (1 - P_{\text{zigzag}}) \\ &= 1 + P_{\text{zigzag}}, \end{aligned} \tag{3}$$

$$E\{\#\text{ success packets in a frame}\} = 1 \cdot P_1 + 2 \cdot P_{\text{zigzag}} \tag{4}$$

For each frame $k \in \{1, 2, ...\}$, let R_k and T_k respectively represent the number of successes in frame k and the size of frame k. Thus $\{R_k\}$ and $\{T_k\}$ are *i.i.d* sequences. Let t_n represent the ending time of the nth frame, and let $N(t_n)$ represent the total number of successes up to time t_n. Define μ as the time average throughput. By renewal/reward theory, we have:

$$\begin{aligned} \mu \equiv \lim_{n\to\infty} \frac{N(t_n)}{t_n} &= \frac{\lim_{n\to\infty} \frac{1}{n}\sum_{k=1}^{n} R_k}{\lim_{n\to\infty} \frac{1}{n}\sum_{k=1}^{n} T_k} = \frac{E\{R\}}{E\{T\}} \\ &= \frac{E\{\#\text{ success packets in a frame}\}}{E\{\text{frame size}\}} \quad \text{with prob. } 1 \end{aligned} \tag{5}$$

Thus,

$$\mu = \frac{P_1 + 2P_{\text{zigzag}}}{1 + P_{\text{zigzag}}} = \frac{2Nq(1-q)^{N-2}(1+(N-2)q)}{2 + N(N-1)q^2(1-q)^{N-2}} \tag{6}$$

To maximize throughput, we need to solve $\frac{\partial \mu}{\partial q} = 0$. But it is challenging to derive a closed-form formula for optimal q from this. Hence, we take a different approach to obtain the maximum achievable throughput.

It can be shown that the optimal transmission probability is $\Theta(1/N)$, and hence we would like to find the optimal throughput by guessing the optimal transmission probability as

$$q^\star = \frac{\alpha}{N+\delta} \tag{7}$$

and optimizing α and δ for the maximum real throughput μ as N goes to infinity. Intuitively, the significant order of the optimal transmission probability must be N^{-1} since any other order will result in throughput of zero as N goes to infinity; higher order will result in infinite repeated collision, and lower order will result in too little transmission relative to N.

Substituting $q^\star$ into throughput μ in Eq. 6 and as $N \to \infty$, the throughput becomes,

$$\mu_N^\star \equiv \frac{2N\frac{\alpha}{N+\delta}\left(1-\frac{\alpha}{N+\delta}\right)^{N-2}\left(1+(N-2)\frac{\alpha}{N+\delta}\right)}{2+N(N-1)\left(\frac{\alpha}{N+\delta}\right)^2\left(1-\frac{\alpha}{N+\delta}\right)^{N-2}} \tag{8}$$

$$\lim_{N\to\infty} \mu_N^\star = \frac{2\alpha(1+\alpha)e^{-\alpha}}{2+\alpha^2 e^{-\alpha}} \tag{9}$$

Optimizing α for maximum throughput $\mu^\star$ using numerical calculation results in $\alpha = 1.4995 \approx 1.5$. Also, in our model with ZIGZAG decoding, the transmission probability q should be 1 for $N \leq 2$ to maximize throughput, since two simultaneous transmissions can always result in average throughput of 1 packet/slot. Hence we set $\delta = -0.5$. Thus, as $N \to \infty$, the optimal transmission probability can be approximated as:

$$\boxed{q^\star = \frac{1.5}{N - 0.5}}, \tag{10}$$

and the bound on the maximum throughput is,

$$\boxed{\mu^\star \equiv \lim_{N\to\infty} \mu^\star_N \approx 0.6688.} \tag{11}$$

Now, we verify the result of the above derivation by numerically finding the maximum throughput at various N values. Using the expression of μ from Eq. 6, we first solve:

$$\frac{\partial \mu}{\partial q} = 0$$

to find the optimal transmission probability $q^\star$, and then use this to evaluate the optimal throughput $\mu^\star_N|_{q^\star}$.

Figure 2 depicts the numerically calculated maximum throughput with ZIGZAG decoding for various N values, along with the derived throughput bound when $N \to \infty$. The figure shows that the numerically calculated maximum throughput does converge to the derived throughput bound as N goes to infinity. Also, Fig. 3 shows the corresponding numerically calculated optimal packet transmission probability at various N values. This is close to the packet transmission probability $q^\star$ derived in Eq. 10. These two results together confirm that our derivation is accurate.

Thus, in a N-user slotted random access system with ZIGZAG decoding, the maximum achievable throughput is 0.6688. Comparing this result to the throughput of a simple random access system without ZIGZAG decoding ($e^{-1} = 0.3678$, presented in [2]) shows that ZIGZAG can improve the throughput by 81.8%.

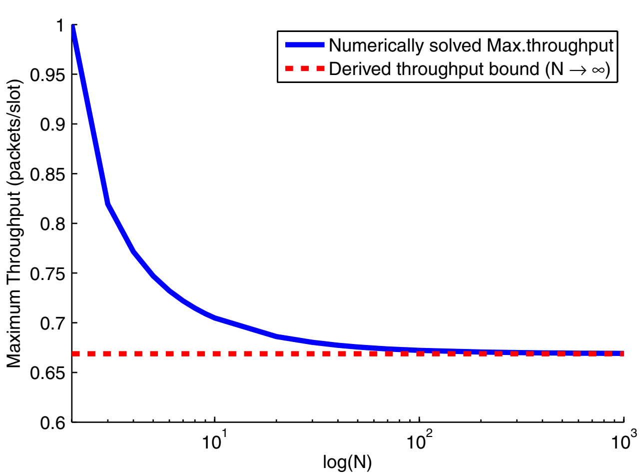


Fig. 2 Maximum throughput at various N, and the throughput bound when $N \to \infty$

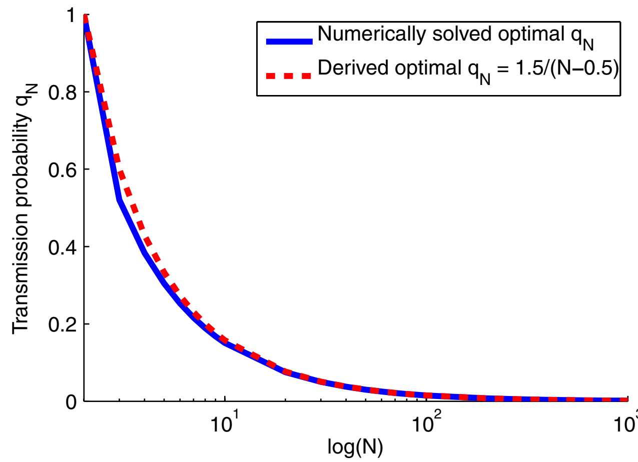


Fig. 3 Transmission probability $q^\star$ that achieves maximum throughput at various N

3 Stabilized slotted Aloha

In this section, we look at a stabilized version of Slotted Aloha [14] with ZIGZAG decoding. In this model, there are an infinite number of users each with at most one packet to send. New users arrive according to a Poisson process with rate λ packets/slot, and they transmit immediately on the slot in which they arrive (even if it is the second slot of a ZIGZAG frame). If the transmission results in a collision, then each user sending one of the colliding packets discovers the collision at the end of the slot and becomes a backlogged user. Backlogged users retransmit their packet independently every slot with some adaptive probability q_n, $(0 \leq q_n \leq 1)$, where n is the number of backlogged users in the system. If the transmitted packet is received at the receiver, then that backlogged user is removed from the system. Otherwise, it remains backlogged.

Let $N(k)$ be the number of backlogged users at the beginning of the k^{th} frame, and suppose that all backlogged users know the value of $N(k)$. At the beginning of the k^{th} frame, if $N(k) = n$, then all backlogged users transmit with probability q_n and new users arrive independently according to a Poisson distribution of rate λ. Feedback is given in the same way as before (see Section 2). The frame has size 1 if the feedback is '0,' '1,' or 'C.' In the case of a 'ZIGZAG' feedback, the frame has size 2, and the same two users who collided re-transmit the same packets in the second slot of the frame. All other backlogged users remain silent.

However, there is a chance that a new arrival on the second slot of the frame also transmits, in which case we get a feedback of 'C,' and we lose all information about the ZigZag event. That is, a ZigZag frame delivers exactly 2 packets in two slots if there are no new arrivals in the second slot of the ZigZag frame, but delivers exactly 0 packets in two slots if there is a new arrival in the second slot of the ZigZag frame (see Fig. 4). Recall that new arrivals that are not successfully delivered are added to the set of backlogged users at the end of the frame.

We first define the probabilities of '1' (success) and 'ZigZag', given that $N(k) = n$, as follows:

$$P_1(n) = \lambda e^{-\lambda}(1-q_n)^n + e^{-\lambda} n q_n (1-q_n)^{n-1}, \tag{12}$$

$$P_{\text{zigzag}}(n) = \frac{\lambda^2}{2} e^{-\lambda}(1-q_n)^n + \lambda e^{-\lambda}\binom{n}{1} q_n (1-q_n)^{n-1} + e^{-\lambda}\binom{n}{2} q_n^2 (1-q_n)^{n-1}. \tag{13}$$

P_1 is the probability of the '1' event where exactly 1 packet attempted transmission at the beginning of a frame. P_{zigzag} is the probability of the 'ZigZag' event, where exactly 2 packets attempted transmission at the beginning of a frame. This event *does not* imply that 2 packets are successfully delivered to the receiver. In fact, since a new arrival may attempt transmission in the second slot of a ZigZag frame and corrupt the ZigZag delivery (Fig. 4), the probability of 2 packets being successfully delivered is:

$$P_2(n) = Pr\{\text{No new arrival \& ZigZag event} \mid N(k)=n\} = e^{-\lambda} P_{\text{zigzag}}(n) \tag{14}$$

Again, the average frame size in this system is given by:

$$E\{\text{frame size} \mid N(k)=n\} = 2 \cdot P_{\text{zigzag}}(n) + 1 \cdot (1 - P_{\text{zigzag}}(n)) = 1 + P_{\text{zigzag}}(n)$$

and the average number of packets successfully delivered to the receiver in a frame is given by:

$$E\{\text{\# packets delivered in a frame} \mid N(k)=n\} = 1 \cdot P_1(n) + 2 \cdot P_2(n) = P_1(n) + 2e^{-\lambda} P_{\text{zigzag}}(n).$$

Now, consider the Discrete Time Markov Chain $N(k)$, and define the drift D_n as:

$$\begin{aligned} D_n &\equiv E\{N(k+1) - N(k) \mid N(k) = n\} \\ &= E\{\text{arrivals} \mid N(k)=n\} - E\{\text{departures} \mid N(k)=n\} \\ &= \lambda(1 + P_{\text{zigzag}}(n)) - (P_1(n) + 2P_2(n)) \\ &= \lambda(1 + P_{\text{zigzag}}(n)) - (P_1(n) + 2e^{-\lambda} P_{\text{zigzag}}(n)) \end{aligned} \tag{15}$$

By the Drift Theorem [6, 13], the system is stable if there is an $\varepsilon > 0$ such that the drift D_n satisfies $D_n \le -\varepsilon$ for large n (and unstable if $D_n > 0$ for large n). By Eq. 15, $D_n < 0$ is equivalent to the following expression:

$$\lambda < \frac{P_1(n) + 2e^{-\lambda} P_{\text{zigzag}}(n)}{1 + P_{\text{zigzag}}(n)}. \tag{16}$$

We now design the transmission probability $q_n^\star$ so that the right hand side of Eq. 16 converges to a constant $\mu^\star$ as $n \to \infty$. Thus, from Eq. 16, the system is stable (with negative drift for large n) whenever $\lambda < \mu^\star$. To this end, define $\mu_n^\star$ as the right hand side of Eq. 16, and define $q_n^\star$ as follows:

$$q_n^\star = \frac{\alpha - \lambda}{n - \delta}, \tag{17}$$

such that nq_n is bounded for large n. The constants α and δ in Eq. 17 will be optimally chosen. By substituting $q_n^\star$ into the right hand side of Eq. 16, and by approximating $(1-q_n)^n \approx (1-q_n)^{n-1} \approx (1-q_n)^{n-2}$ and sending $n \to \infty$, we get:

$$\mu^\star \equiv \lim_{n\to\infty} \mu_n^\star = \frac{\alpha e^{-\alpha} + \alpha^2 e^{-\alpha} e^{-\lambda}}{1 + \frac{\alpha^2}{2} e^{-\alpha}}. \tag{18}$$

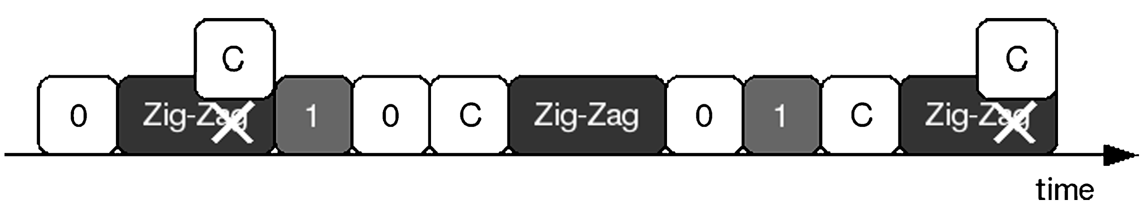


Fig. 4 A timeline of events in slotted Aloha. Two ZigZag frames are ruined by new arrivals on the second slot

To find the maximum throughput bound, we numerically solve the following two equations to optimize α:

$$\frac{\partial \mu^\star}{\partial \alpha} = 0, \quad \mu^\star = \lambda \tag{19}$$

to obtain the optimal $\alpha^\star = 1.310$. Hence, the optimal transmission probability $q_n^\star$ is given by:

$$q_n^\star = \frac{1.31 - \lambda}{n - \lambda - 0.69}, \tag{20}$$

where δ has been set so that $q_n^\star = 1$ when $n = 2$. This gives us the optimal maximum throughput bound of

$$\boxed{\mu^\star = \lim_{n\to\infty} \mu_n^\star = 0.5123.} \tag{21}$$

Thus in a stabilized slotted Aloha system, ZigZag decoding improves the maximum throughput from $e^{-1} = 0.3678$ to 0.5123. This is a 39% improvement compared to the system without ZigZag decoding. However, this improvement result is not as much as that of the N-user random access model in Section 2. The reason is that the system above suffers from a loss of throughput if a new user arrives in the second slot of a ZigZag frame. Thus, we can improve the system throughput by modifying the above random access protocol so that the new arrivals do not transmit immediately. They listen to hear feedback at the end of the slot in which they arrive, and are then added to the set of 'backlogged' packets without transmitting on the next slot if they hear a 'ZigZag' in the slot in which they arrived. In such a way, a ZigZag event will always end with two successful packet receptions.

Then the above analysis should be slightly modified such that:

$$(\text{Eq. } 14) \longrightarrow \hat{P}_2(n) = P_{\text{zigzag}}(n), \tag{22}$$

$$(\text{Eq. } 15) \longrightarrow \hat{D}_n = \lambda(1 + P_{\text{zigzag}}(n)) - (P_1(n) + 2P_{\text{zigzag}}(n)), \tag{23}$$

$$(\text{Eq. } 16) \longrightarrow \hat{\mu}_n = \frac{P_1(n) + 2P_{\text{zigzag}}(n)}{1 + P_{\text{zigzag}}(n)}, \tag{24}$$

$$(\text{Eq. } 18) \longrightarrow \hat{\mu}^\star = \frac{\alpha e^{-\alpha} + \alpha^2 e^{-\alpha}}{1 + \frac{\alpha^2}{2} e^{-\alpha}}, \tag{25}$$

and numerically solving Eq. 19 with these modified equations results in optimal $\hat{\alpha}^\star = 1.3558$. Thus, the optimal transmission probability $\hat{q}^\star$ is given as follows:

$$\hat{q}_n^\star = \frac{1.3558 - \lambda}{n - \lambda - 0.6442}, \tag{26}$$

where δ has been set so that $\hat{q}^\star = 1$ when $n = 2$. This gives the optimal maximum throughput bound of:

$$\boxed{\hat{\mu}^\star = \lim_{n\to\infty} \hat{\mu}_n^\star = 0.6688.} \tag{27}$$

The above result is identical to the result from the N-user random access system in Section 2; *ZigZag decoding improves maximum throughput from* 0.3678 *to* 0.6688*, an* 81.8% *improvement compared to the system without ZigZag decoding in stabilized slotted Aloha.*

Although our analysis do not consider conflict resolution techniques such as [3, 8], which provide a maximum throughput of 0.43 and 0.4871 respectively, we believe that ZigZag decoding will improve throughput bounds in those cases as well. Since the essence of ZigZag decoding is to allow successful reception of 2 colliding packets, it is complementary to conflict resolution techniques, and hence it will only increase the maximum throughput even when the conflict resolution is used.

Finally, we have also verified this maximum achievable throughput using packet-level simulations. In the simulation, we have used $\hat{q}_n^\star$ in Eq. 26 as the adaptive transmission probability (recall that n is the number of backlogged packets in the system), and each simulation ran for 100,000 packets with maximum backlog queue size of 500. Specifically, each simulation was run for a fixed value of λ, and the value of λ was increased for each new simulation to find the maximum λ for which there are no queue overflow events. The simulation resulted in maximum throughput of 0.6675. This result is almost identical to the optimal throughput obtained above, and confirms that our analysis is accurate.

4 CSMA slotted Aloha

In this section, we examine the maximum throughput of a slotted non-persistent CSMA [7] (Carrier Sense Multiple Access) system. This system is almost identical to the slotted Aloha system in the previous section (Section 3) with the following differences. The major difference between CSMA slotted Aloha and ordinary slotted Aloha is that idle slots in CSMA are 'Mini-slots' with a duration of β units of time ($\beta \leq 1$) (See Fig. 5). Newly arriving users during an idle slot (mini-slot) will attempt transmission in the next slot. If a new user with a packet arrives while transmissions are in progress, (either success, ZigZag, or collision) they are added to the set of backlogged users. Backlogged users wait until they hear a free mini-slot, then attempt transmission with probability q_n.

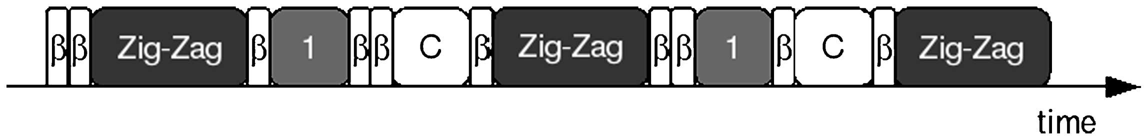


Fig. 5 A timeline showing the state transitions in CSMA Aloha with ZIGZAG

To analyze the CSMA Aloha, we can use a Markov chain again, using the number n of backlogged packets as the state and the end of idle slots as the state transition times. Note that each busy (success, ZIGZAG, or collision) frame must be followed by an idle slot, since nodes are allowed to start transmission only after detecting an idle slot.

In this system, there are four different types of states; idle, success-idle, collision-idle, and ZIGZAG-idle. The time between successive state transitions is either β (in the case of an idle slot), or $1+\beta$ (in the case of success or collision slot followed by an idle), or $2+\beta$ (in the case of ZIGZAG slots followed by an idle) (see Fig. 6).

First, we define the probabilities of each of four events:

$$P_0(n) = e^{-\lambda\beta}(1-q_n)^n, \tag{28}$$

$$P_1(n) = \lambda\beta e^{-\lambda\beta}(1-q_n)^n + e^{-\lambda\beta} n q_n (1-q_n)^{n-1}, \tag{29}$$

$$\begin{aligned} P_{\text{zigzag}}(n) = &\frac{(\lambda\beta)^2}{2} e^{-\lambda\beta}(1-q_n)^n \\ &+ \lambda\beta e^{-\lambda\beta}\binom{n}{1} q_n (1-q_n)^{n-1} \\ &+ e^{-\lambda\beta}\binom{n}{2} q_n^2 (1-q_n)^{n-2}, \end{aligned} \tag{30}$$

$$P_C(n) = 1 - P_0(n) - P_1(n) - P_{\text{zigzag}}(n). \tag{31}$$

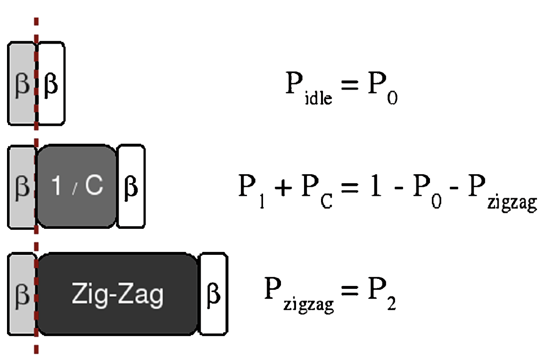


Fig. 6 Three possible state transition times in CSMA Aloha with ZIGZAG: β to 'β,' '1 or C,' 'ZIGZAG'

The expected time between successive state transitions and the expected number of arrivals and departures are given as:

$$\begin{aligned} E\{\text{transition time}\} &= \beta + 1\cdot(P_1(n) + P_C(n)) \\ &\quad + 2\cdot P_{\text{zigzag}}(n) \\ &= \beta + 1\cdot(1 - P_0(n) - P_{\text{zigzag}}(n)) \\ &\quad + 2\cdot P_{\text{zigzag}}(n) \\ &= \beta + 1 - P_0(n) + P_{\text{zigzag}}(n), \end{aligned} \tag{32}$$

$$\begin{aligned} E\{\text{arrivals}\} &= \lambda\cdot E\{\text{transition time}\} \\ &= \lambda(\beta + 1 - P_0(n) + P_{\text{zigzag}}(n)), \end{aligned} \tag{33}$$

$$E\{\text{departures}\} = P_1(n) + 2\cdot P_{\text{zigzag}}(n). \tag{34}$$

Then we can obtain the drift D_n using Eqs. 33 and 34 as follows:

$$\begin{aligned} D_n &\equiv E\{N(k+1) - N(k) | N(k) = n\} \\ &= E\{\text{arrivals}\} - E\{\text{departures}\} \\ &= \lambda(\beta + 1 - P_0(n) + P_{\text{zigzag}}(n)) \\ &\quad - (P_1(n) + 2P_{\text{zigzag}}(n)). \end{aligned} \tag{35}$$

From the equation above, we can see that the drift in state n is negative if the following holds:

$$\lambda < \frac{P_1(n) + 2P_{\text{zigzag}}(n)}{\beta + 1 - P_0(n) + P_{\text{zigzag}}(n)}. \tag{36}$$

To keep the system stable, the drift D_n must satisfy $D_n < 0$ for large n. Hence, the right hand side of Eq. 36 is the bound on the maximum throughput:

$$\mu^\star \equiv \lim_{n\to\infty}\left[\frac{P_1(n) + 2P_{\text{zigzag}}(n)}{\beta + 1 - P_0(n) + P_{\text{zigzag}}(n)}\right].$$

Suppose the packet transmission probability has the following structure:

$$q_n \equiv \frac{\alpha - \lambda\beta}{n}, \tag{37}$$

where α is a constant to be determined. In the above equation, $\alpha \equiv \lambda\beta + nq_n$ is the expected number of

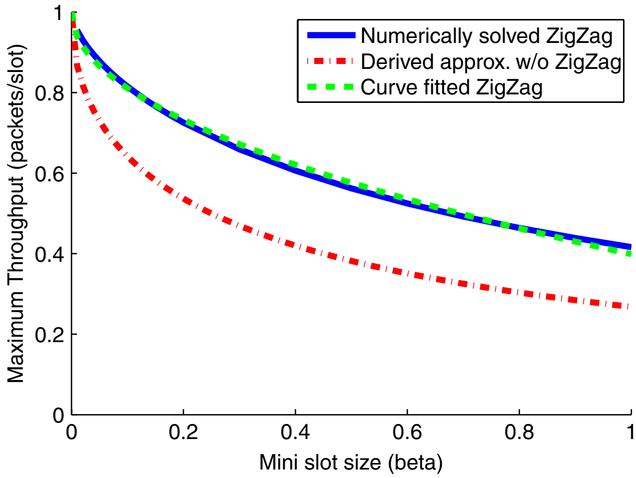


Fig. 7 Maximum throughput bound of the CSMA system with and without ZigZag at different β values

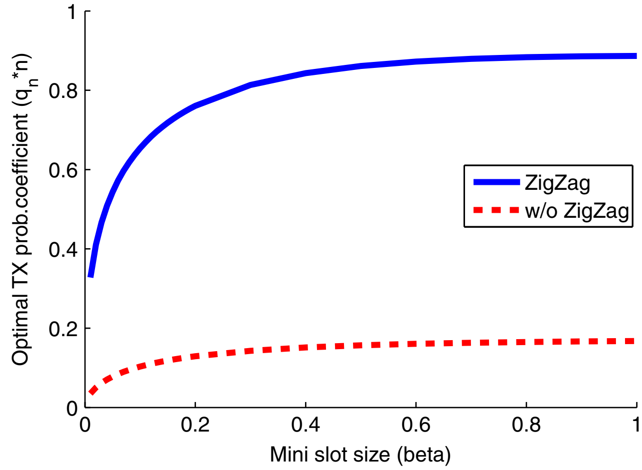


Fig. 8 Optimal aggregate transmission probability $q_n * n$ (with and without ZigZag) at different β values for the CSMA system

attempted transmissions following a transition to state n. For large n, we have the approximation:

$$(1-q_n)^n \approx (1-q_n)^{n-1} \approx (1-q_n)^{n-2} \approx e^{-nq_n}$$

We can thus approximate the state transition probabilities in Eqs. 28–30 as,

$$P_0 \approx e^{-\alpha}, \quad P_1 \approx \alpha e^{-\alpha}, \quad P_{\text{zigzag}} \approx \frac{\alpha^2}{2} e^{-\alpha}.$$

Sending $n \to \infty$ leads to the following expression for the maximum throughput $\mu^\star$:

$$\mu^\star \approx \frac{\alpha e^{-\alpha} + \alpha^2 e^{-\alpha}}{\beta + 1 - e^{-\alpha} + \frac{\alpha^2}{2} e^{-\alpha}}. \tag{38}$$

To find the bound on maximum throughput $\mu^\star$, we optimize α for large n by solving,

$$\frac{\partial \mu^\star}{\partial \alpha} = 0, \tag{39}$$

$$\mu^\star = \lambda. \tag{40}$$

Unfortunately, it is difficult to solve Eq. 39 to obtain an intuitive closed-form formula. Hence, we numerically solve Eqs. 39 and 40 together at various β values to obtain the maximum throughput bound.

Figure 7 depicts the numerically calculated maximum throughput with ZigZag along with the approximated throughput without ZigZag ($\beta + 1 - \sqrt{\beta^2 + 2\beta}$) at various β values.[6] Also, Fig. 8 depicts the numerically calculated optimal packet transmission probability at various β values. Curve fitting the numerically calculated maximum throughput values gives us the following expression for the maximum throughput with ZigZag:

$$\boxed{\mu^\star = 1 - 0.5966\sqrt{\beta} - 0.0045\beta.} \tag{41}$$

We have verified this maximum achievable throughput at various β values using packet-level simulations. In the simulation, we have used the numerically obtained optimal transmission probabilities (in Fig. 8), and each simulation ran for 100,000 packets with maximum backlog queue size of 500.

Figure 9 shows the maximum throughput achieved in the simulation with and without ZigZag decoding. The figure also includes the numerically calculated

[6] Analysis in [2] presents $1 - \sqrt{2\beta}$ as the approximate maximum throughput, but that came from approximating $\beta + 1 - \sqrt{\beta^2 + 2\beta}$.

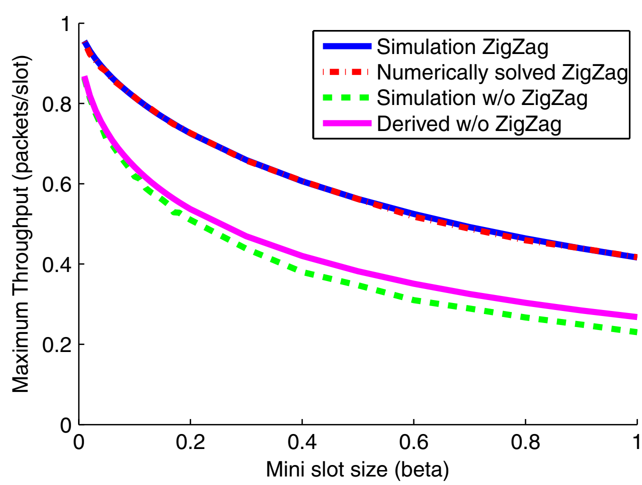


Fig. 9 Maximum throughput results (with and without ZigZag) from simulations at different β values for the CSMA system

throughput bounds in Fig. 7 for convenience of comparison. The simulation results clearly show that the maximum throughput achieved by using ZigZag decoding is greater than the case where ZigZag decoding is not used. For example when $\beta = 0.01$, the maximum throughput with ZigZag is 0.9645 and without ZigZag is 0.8682 which corresponds to 11.1% improvement. When $\beta = 0.1$, they are 0.8122, 0.6417, and 26.5% respectively. Furthermore, the figure also shows that the simulation results for both with and without ZigZag are almost identical to the numerically calculated maximum throughput for the case with ZigZag decoding and derived approximate maximum throughput without ZigZag ($\beta + 1 - \sqrt{\beta^2 + 2\beta}$), respectively. For the case without ZigZag decoding, the simulation result is very close to the calculated approximated throughput when β is small, but diverges a little when when β grows. This is because the derivation of $\mu \approx \beta + 1 - \sqrt{\beta^2 + 2\beta}$ used as assumption that $e^{-x} \approx 1 - x + \frac{x^2}{2}$ for $\beta \ll 1$.

5 CSMA un-slotted Aloha

In this section, we extend the discussion in Section 4 to find the maximum throughput of an *unslotted* CSMA Aloha system. This system is similar to the slotted CSMA system in the previous section (Section 4) with the following differences: Newly arriving users will attempt transmission immediately if the channel is sensed to be idle. We define β as the propagation and detection delay, such that if a user starts transmission at time t, another user will *not* detect that the channel is busy until $(t + \beta)$. For simplicity, we assume β to be the same for all pairs of users. If the channel is sensed to be busy, or if the transmission results in a collision, the new users are added to the set of backlogged ones. If a user senses the channel to be idle when it is actually busy, it will start transmission and concurrent transmissions will exist in the channel. A ZigZag event occurs when there are exactly two concurrent transmissions in the channel (see Fig. 10). We assume that a ZigZag event finishes within $2(1 + \beta)$ time from the beginning of first transmission, and the channel can be sensed as busy during $[t + \beta, t + 2(1 + \beta)]$. A collision event occurs when there are three or more concurrent transmissions in the channel. Backlogged users repeatedly attempt retransmission (until the transmission is successful) at randomly selected times separated by independent, exponentially distributed random delays τ, with probability density $x(n)e^{-\tau x(n)}$.

Now, consider an idle period where there are n backlogged users in the system. The time until the first transmission starts is an exponentially distributed random variable with rate

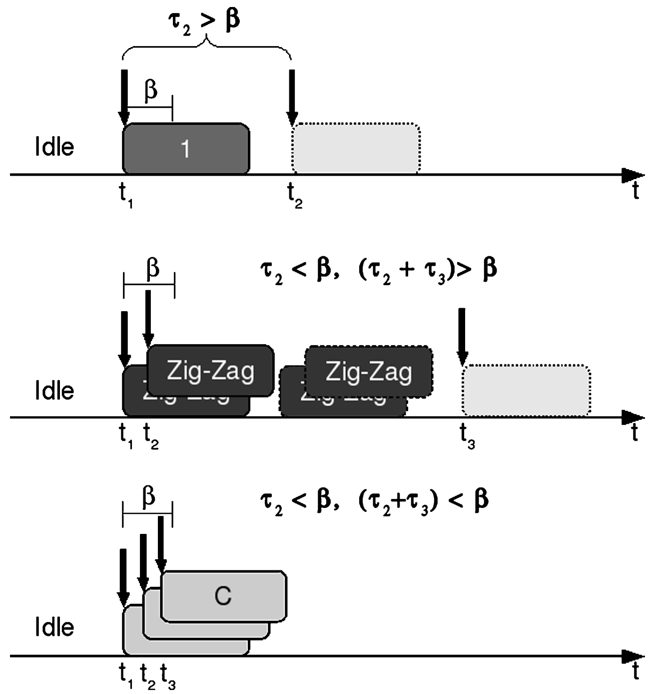


Fig. 10 Three possible state transition times in unslotted CSMA Aloha with ZigZag: idle to '1,' or 'C,' 'ZigZag' to idle

$$G(n) = \lambda + n \cdot x(n) \tag{42}$$

Note that $G(n)$ is the attempt rate in packets per unit time. Assuming $G(n) \approx G(n-1)$ for large n, the time from the first transmission until the next new arrival or backlogged node senses the channel is again an exponentially distributed random variable of rate $G(n)$. Successful transmission occurs if this second sensing is done after time β. A ZigZag event will occur if this second sensing is done within time β but the successive sensing is done after time β. A collision will occur otherwise. Thus, the probabilities of a successful transmission, a collision, and a ZigZag event can be approximated as:

$$P_1(n) \approx e^{-\beta G(n)}, \tag{43}$$

$$P_{\text{zigzag}}(n) \approx \beta G(n) e^{-\beta G(n)}, \tag{44}$$

$$P_C(n) = 1 - P_1(n) - P_{\text{zigzag}}(n) \approx 1 - e^{-\beta G(n)} - \beta G(n) e^{-\beta G(n)}. \tag{45}$$

Thus in this system, there are three different types of state transitions; idle-success, idle-collision, and idle-ZigZag. Note that each busy (success, ZigZag, or collision) frame must be followed by an idle period, since nodes are allowed to start transmission only after detecting the channel idle. Also, the system becomes idle after each busy frame (see Fig. 10). Then the expected time between successive state transitions, which is the time from the beginning of one idle period until the next, is given by

$$E\{\text{transition time}\} = \frac{1}{G(n)} + (\beta+1)(P_1(n) + P_C(n)) + 2(\beta+1)\cdot P_{\text{zigzag}}(n). \quad (46)$$

In this expression, the $1/G(n)$ term is the expected time until the first transmission starts. The time duration of a success or collision frame is approximately $1+\beta$, where β accounts for the time until the first transmission ends and the channel is detected as being idle again. Finally, the last term $2(1+\beta)$ accounts for the case of ZigZag frame, which we have assumed to finish within additional $1+\beta$ time.

Given the expected number of departures per state transition as,

$$E\{\text{departures}\} = P_1(n) + 2\cdot P_{\text{zigzag}}(n), \quad (47)$$

the throughput $\mu(n)$ of this system becomes:

$$\mu(n) \equiv \frac{E\{\text{departures}\}}{E\{\text{transition time}\}} \quad (48)$$

$$= \frac{P_1(n) + 2\cdot P_{\text{zigzag}}(n)}{\frac{1}{G(n)} + (\beta+1)\left\{(P_1(n)+P_C(n)) + 2P_{\text{zigzag}}(n)\right\}}. \quad (49)$$

To find the bound on maximum throughput, we again numerically optimize $G(n)$ for large n at various β values to obtain the maximum throughput $\mu^\star(n)$ using the same techniques used in the previous section. We also verified this maximum achievable throughput at various β values using packet-level simulations with the same setup used in the previous section. Figure 11 depicts several things: it plots (1) maximum throughput achieved in the simulation with and without ZigZag decoding, (2) numerically calculated maximum throughput with and without ZigZag, (3) derived optimal throughput without ZigZag $(\frac{e^{-\sqrt{\beta}}}{1+\sqrt{\beta}+\beta})$ [2], and (4) a curve fitted from the throughput results from the simulation with zigzag at various β values. Curve fitting the numerically calculated maximum throughput values gives us the following expression for the maximum throughput with ZigZag for the unslotted CSMA system:

$$\boxed{\mu^\star = 1 - 0.8795\sqrt{\beta} + 0.1550\beta.} \quad (50)$$

The figure shows that the simulation result matches the numerically calculated maximum throughput for both the cases with and without ZigZag decoding. The simulation results and the numerically solved values together clearly show that the maximum throughput achieved by using ZigZag decoding is greater than the case where ZigZag decoding is not used also for the unslotted CSMA system.

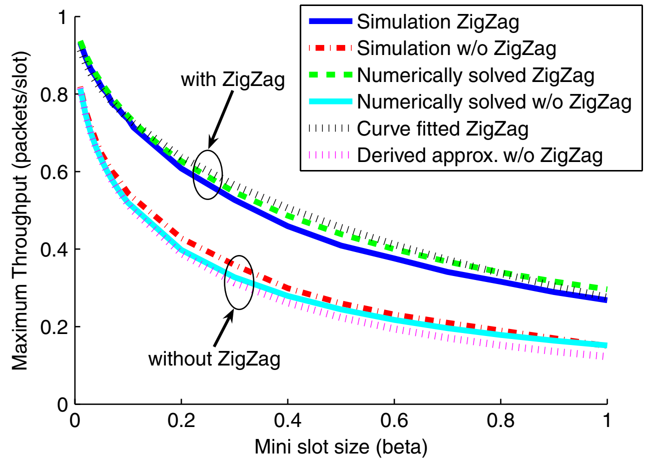


Fig. 11 Maximum throughput bound of the unslotted CSMA system with and without ZIGZAG at different β values

Finally, we compare our results to the real experimental results reported in [5]. In their work, the authors have implemented a ZigZag prototype in GNU radio, and evaluated it in a 14-node testbed. In this setup, 10% of the sender-receiver pairs were hidden terminals, 10% sensed each other partially, and the rest sensed each other perfectly. Their evaluation results showed that, averaging over all sender-receiver pairs, ZigZag improved the average throughput by 25.2% when compared to current 802.11. To compare this result to our analysis, we need to choose a β value to be used in Eq. 50. For their experiments, the authors have used 1,500-byte payload at bit rate of 500 kb/s, which corresponds to data transmission time of 24 ms. However, they have not described the details of what DIFS and BO (Backoff) values they have used. Considering

Table 1 Maximum throughput bounds with and without ZigZag decoding

Model	W/o ZigZag	With ZigZag	% gain
Random access	0.3678	0.6688	81.8
Aloha	0.3678	0.6688	81.8
CSMA ($\beta = 0.1$)	0.6417	0.8122	26.5
CSMA ($\beta = 0.05$)	0.7298	0.8759	20.0
CSMA Unslotted ($\beta = 0.1$)	0.5193	0.7430	43.0
CSMA Unslotted ($\beta = 0.05$)	0.6298	0.8287	31.5

the fact that DIFS is 50 μs and BO is 0∼620 iμs for 802.11b, we choose $\beta \approx 0.03$. Then, for $\beta = 0.03$, the maximum throughput for Unslotted CSMA is 0.6999 without ZigZag and 0.8746 with ZigZag, resulting in an throughput gain of 24.9%. Thus, we observe that our results closely matches the real experimental results reported in [5].

6 Conclusion

We have presented throughput bounds for multi-user random access systems with ZigZag decoding [5]. Our analysis and simulation results have shown that ZigZag decoding can significantly improve the maximum throughput of random access systems. Table 1 summarizes the main results. Although the analysis has been done using simplified models, we believe that the same trend will hold in real systems as well. We believe that our results with ZigZag decoding motivates a MAC design with more aggressive transmission behaviour, exploiting concurrent transmissions to maximize throughput.

References

1. Abramson N (1970) The aloha system: another alternative for computer communications. In: AFIPS '70 (fall): proceedings of the November 17–19, 1970, fall joint computer conference, pp 281–285
2. Bertsekas D, Gallager R (1992) Data networks, 2nd edn. Prentice-Hall, Inc., Upper Saddle River, NJ, USA
3. Capetanakis J (1979) Tree algorithms for packet broadcast channels. IEEE Trans Inform Theory 25(5):505–515
4. Ephremides A, Hajek B (1998) Information theory and communication networks: an unconsummated union. IEEE Trans Inform Theory 44:2416–2434
5. Gollakota S, Katabi D (2008) Zigzag decoding: combating hidden terminals in wireless networks. In: SIGCOMM '08: proceedings of the ACM SIGCOMM 2008 conference on data communication
6. Kaplan M (1979) A sufficient condition for nonergodicity of a Markov chain. IEEE Trans Inform Theory 25:470
7. Kleinrock L, Tobagi F (1975) Packet switching in radio channels: part i–carrier sense multiple-access modes and their throughput-delay characteristics. IEEE Trans Commun 23(12):1400–1416
8. Mosely J, Humblet P (1985) A class of efficient contention resolution algorithms for multiple access channels. IEEE Trans Commun 33(2):145–151
9. Mutsuura K, Okada H, Ohtsuki K, Tezuka Y (1989) A new control scheme with capture effect for random access packet communications. In: IEEE international conference on communications (ICC'89), vol 2, pp 938–944
10. Namislo C (1984) Analysis of mobile radio slotted aloha networks. IEEE J Sel Areas Commun 2(4):583–588
11. Ngo MH, Krishnamurthy V, Tong L (2008) Optimal channel-aware aloha protocol for random access in wlans with multipacket reception and decentralized channel state information. IEEE Trans Signal Process 56(6):2575–2588
12. Paek J, Neely M (2009) Mathematical analysis of throughput bounds in random access with zigzag decoding. In: 7th international symposium on modeling and optimization in mobile, ad hoc, and wireless networks (WiOpt'09)
13. Pakes AG (1969) Some conditions for ergodicity and recurrence of Markov chains. Oper Res 17(6):1059–1061
14. Roberts LG (1975) Aloha packet system with and without slots and capture. SIGCOMM Comput Commun Rev 5(2):28–42

Jeongyeup Paek recieved his B.S. degree from Seoul National University in 2003 and his M.S. degree from University of Southern California in 2005, both in Electrical Engineering. He then received his Ph.D. degree in Computer Science from the University of Southern California in 2010. He is currently a postidoctoral research fellow in the Embedded Networks Laboratory at the University of Southern California. Jeongyeup Paek's prior research has focused on topics in wireless sensor network systems including reliable transport protocols, architecture for tiered embedded networks, and real-world sensor network applications. His recent research interests are in the areas of mobile smartphones with focus on energy-efficient localization and software architecture.

Michael J. Neely received B.S. degrees in both Electrical Engineering and Mathematics from the University of Maryland, College Park, in 1997. He then received a 3 year Department of Defense NDSEG Fellowship for graduate study at the

Massachusetts Institute of Technology, where he received an M.S. degree in EECS in 1999 and a Ph.D. in 2003. During the Summer of 2002, he worked as an intern in the Distributed Sensor Networks group at Draper Labs in Cambridge. He is currently an Assistant Professor in the Communication Sciences Institute (CSI), within the Electrical Engineering Department at the University of Southern California. His research interests are in the areas of stochastic network optimization and queueing theory, with applications to wireless, satellite, mobile ad-hoc networks, and switching systems. Michael received the NSF Career award in 2008. He is a member of Tau Beta Pi and Phi Beta Kappa.