Increasing the Throughput of Spread-Aloha Protocols via Long PN Spreading Codes*

A. Ribeiro, Y. Yu, G. B. Giannakis (contact author) and N. D. Sidiropoulos+
Dept. of ECE, University of Minnesota
200 Union Street, Minneapolis, MN 55455, USA †
+ Dept. of ECE, Technical University of Crete, Chania - Crete, Greece

***Abstract*— Random access Aloha protocols have well documented merits in terms of simplicity and favorable delay-throughput trade-off under moderate bursty traffic loads. Short spreading codes have been used in conjunction with random access to endow Aloha with benefits originating from spread-spectrum communications. Instead of short, symbol-periodic spreading, this paper considers long pseudo-random (PN) packet-periodic sequences in the context of spread-Aloha and establishes that long PN codes increase the maximum stable throughput by reducing the probability of collisions. Relying on a dominant system approach, we analyze the resultant throughput and demonstrate that increasing the PN code length quickly transforms the collision-limited channel to an interference-limited one. In particular, we investigate how throughput depends on user load and packet length. Finally, we discuss synchronization issues and provide corroborating numerical results.**

Keywords — Random Access, Spread Spectrum Communications.

I. Introduction

Incorporation of spread spectrum (SS) techniques in random access (RA) channels is known to permeate benefits of SS communications to medium access control protocols [6], [5], [1]. Provided their queues are not empty, users in the Spread Aloha protocol of [1] share a common short Pseudo random-Noise (PN) sequence of length equal to the spreading gain N, to transmit packets at random time instances. Each user's PN sequence is thus randomly shifted with respect to a common reference time. Since these random shifts are highly unlikely to be identical, the delta-like autocorrelation property of PN sequences enables statistical separation of users at the access point. In this sense, the N possible shifts constitute a resource in SS-RA for which users contend. The throughput (normalized by the spreading gain N) turns out to equal that of a conventional Aloha system, namely $1/(2e)$ [1]. Moreover, synchronization turns to be particularly simple, since it suffices to have at the base station a continuous correlator sampling the received waveform at the chip rate in order to separate the individual users' symbols.

* Work in this paper was prepared through collaborative participation in the Communications and Networks Consortium sponsored by the U. S. Army Research Laboratory under the Collaborative Technology Alliance Program, Cooperative Agreement DAAD19-01-2-0011. The U. S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation thereon.

† Tel/fax: (612)626-7781/625-4583; Email: {aribeiro, yyu, nikos, georgios}@ece.umn.edu.

Instead of using short, symbol-periodic codes, this paper considers a Spread Aloha system in which users share a *long* PN sequence of period $P = LN$, where L denotes packet length L, and as in [1], N stands for the spreading gain. Thus spreading is packet-periodic as opposed to symbol-periodic. This seemingly minor difference will turn out to critically impact the collision probability and considerably increase the throughput of Spread Aloha to the extent that it is essentially limited by interference. While long PN sequences have been already considered for SS-RA [4], [7], the contribution of this paper is threefold: i) finite-population throughput analysis based on a dominant system approach [8]; see also [2]; ii) incorporation of the effect of Forward Error Control (FEC) coding on the throughput of Aloha with packet-periodic spreading, and iii) the recognition that Spread Aloha based on packet-periodic long PN sequences yields simpler synchronization of SS-RA protocols as an important side-benefit.

The rest of the paper is organized as follows. In Section II, we introduce our protocol and discuss how statistical user separation is achieved. In Section III, we analyze its throughput and derive expressions showing that throughput is limited by interference plus collisions that emerge only when different users' packets arrive synchronously. In Section IV we discuss pertinent synchronization issues, while in Section V we numerically evaluate throughput in different settings, and optimize packet length to maximize throughput. Finally, Section VI concludes the paper.

II. Protocol description

Consider the SS-RA system depicted in Fig. 1. Each of the $J+1$ users has an infinitely long buffer for storing packets of fixed length L that arrive with rate λ packets per packet duration. These packets are to be transmitted to a certain access point (call it from now on base-station (BS)). The arrival processes are assumed independent and identically distributed (i.i.d.) across users so that the total arrival rate is $(J+1)\lambda$ packets per packet duration. Prior to their transmission, packets are spread using a long-PN sequence of period NL. Letting $\mathbf{d}^{(j)} := \{d^{(j)}(n)\}_{n=0}^{L-1}$ denote data in a packet of user U_j, the transmitted chip sequence is:

$$x^{(j)}(Nl+n) = d^{(j)}(l)c(Nl+n), \;\; l = 0, 1, \ldots, L-1, \quad n = 0, 1, \ldots, N-1. \quad (1)$$

where $\mathbf{x}^{(j)} := \{x^{(j)}(n)\}_{n=0}^{NL-1}$ is a vector representing the transmitted block of the j^{th} user, and $\mathbf{c} := \{c(n)\}_{n=0}^{NL-1}$ is the

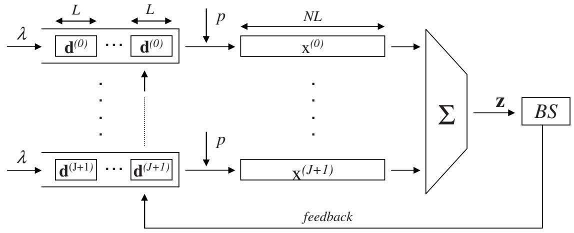

Fig. 1. The packet $\mathbf{d}$ of length L is spread before transmission to construct the transmitted block $\mathbf{x}$ of length NL. Arrivals are at a rate of λ packets per second and departures occur with probability p.

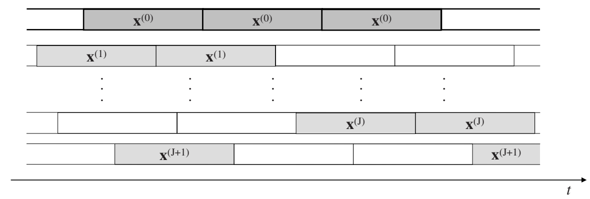

Fig. 2. Combination of rules [R3'] and [R4] effectively partitions the time axis into asynchronous packet slots

long PN sequence *shared* by all users. Here, $c(\cdot)$ should be interpreted as the cyclic extension of $\{c(n)\}_{n=0}^{NL-1}$.

Transmission of queued packets obeys the following rules that describe the proposed protocol:

[R1] Packets are spread before transmission according to (1).

[R2] User nodes are allowed to be chip asynchronous, and no attempt is made to synchronize them.

[R3] When queued packets are available, each user node transmits them with probability p.

[R4] If a user node decides not to transmit, it waits for NL chips before following [R3].

Notice that [R3] controls the transmission rate that will be adjusted to maximize throughput. The purpose of [R4], on the other hand, will become clear in Section III. In addition, [R1] and [R2] require further elaboration.

While [R2] facilitates practical implementation, it will also play an instrumental role in improving throughput in SS-RA when combined with [R1]. To see this, let T_j be the transmission starting time of user U_j measured in chip intervals with respect to a common time reference. When expressed in terms of this common time reference, (1) becomes

$$x^{(j)}(Nl+n) = d^{(j)}(l-T_j)c(Nl+n-T_j)p(Nl+n-T_j), \quad (2)$$

where $p(t)$ is a unit-amplitude square pulse with nonzero support over $t \in (0, NL)$. The signal $\mathbf{z} := \{z(n)\}_{n=0}^{NL-1}$ received at the BS comprises the superposition of (up to) J transmissions and has entries

$$z(Nl+n) = \sum_{j=0}^{J} d^{(j)}(l-T_j)c(Nl+n-T_j)p(Nl+n-T_j) + n(Nl+n), \quad (3)$$

where $n(Nl+n)$ denotes zero mean Additive White Gaussian Noise (AWGN) with variance $\mathrm{E}[n^2(Nl+n)] = N_0$. To recover the packet transmitted by the user of interest (here U_0), we despread the sequence in (3) using a properly delayed version of the long PN sequence $\mathbf{c}$. The resultant decision vector $\mathbf{r}^{(0)} := \{r^{(0)}(l)\}_{l=0}^{L-1}$ has entries

$$\begin{aligned} r^{(0)}(l) &= \frac{1}{N}\sum_{n=0}^{N-1} z(Nl+n)c(Nl+n-T_0) \\ &= d^{(0)}(l-T_0) + i^{(0)}(l) + \tilde{n}(Nl+n), \end{aligned} \quad (4)$$

where the AWGN $\tilde{n}(Nl+n)$ satisfies $\mathrm{E}[\tilde{n}^2(Nl+n)] = N_0/N$; and the interference term $i^{(0)}(l)$ caused by users $\{U_j\}_{j=1}^{J}$ is given by

$$i^{(0)}(l) = \sum_{j=1}^{J} d^{(j)}(l-T_j) \times \sum_{n=0}^{N-1} \left[\begin{array}{c} c(Nl+n-T_j)c(Nl+n-T_0) \\ p(Nl+n-T_j)p(Nl+n-T_0) \end{array} \right]. \quad (5)$$

Since the autocorrelation of long PN sequences is approximately $\delta(t)$, it follows that $\mathrm{E}[c(Nl+n-T_j)c(Nl+n-T_0)] \approx 0$, for $T_j \neq T_0$ [3]. Hence, taking expectations in (4), we obtain [c.f. (4) and (5)],

$$\mathrm{E}[\mathbf{r}^{(0)}] = \mathbf{d}^{(0)}. \quad (6)$$

In a nutshell, [R1] and [R2] ensure that our protocol achieves statistical separation of different users' packets whose probability of error is determined by the interference $\mathbf{i}^{(0)}$. Notice though that there is also a chance to have $T_j = T_0$ for some j(s). Both this and the interference term will determine the throughput of the proposed protocol that we analyze next.

III. Throughput Analysis

As mentioned in the previous section, the throughput is affected by two factors. The first is due to collisions corresponding to the event that $T_j = T_0$, while the second amounts to the interference-plus-noise induced packet error probability.

To analyze throughput of the protocol we introduced in Section II, we will adopt a dominant system approach [8] that provides a bound on the maximum stable throughput; see also [2]. Even when having empty buffers, users in the dominant system transmit a dummy packet; in this case, [R3] can be replaced by:

[R3'] Every user node transmits with probability p. If its queue is empty it transmits a dummy packet.

Rule [R3'] renders the system stationary and leads to a number of simplifications. Indeed, the probability $p = P_a(U_j)$ that any given user node U_j is active, becomes independent of its queue state. Moreover, applying [R4] to the dominant system, effectively partitions the time axis into asynchronous packet slots as depicted in Fig. 2. If the first decision on the packet of user U_0 is taken at time T_0, then subsequent decisions will be made at times $T_0 + k(NL)$, $k \in \mathbf{N}$, regardless of the history of prior decisions.

Since frames of users U_j and U_0 will collide if and only if they start at the same time, a necessary condition for collision to occur is: $T_0 \bmod (NL) = T_j \bmod (NL)$; and given that U_0 is active, the probability that U_j's frame collides with U_0's will be

$$P_c(i|0) = \Pr\{T_0 \bmod (NL) = T_j \bmod (NL)\} P_a(U_j) = \frac{p}{NL}. \quad (7)$$

From (7), the probability that U_0's frame collides with a frame transmitted by any of the other J users follows as

$$P_c(0) = 1 - \left(1 - \frac{p}{NL}\right)^J, \quad (8)$$

where we recall that $P_c(0)$ is conditioned on U_0 being active. Notice that the collision probability $P_c(0)$ in (8) is inversely proportional to the packet length measured in chips NL. As this is typically a very large number, $P_c(0)$ is expected to be small.

Let us now turn to the more likely throughput-limiting factor which is that a packet may be detected erroneously due to the interference present. To this end, consider the decision variable in (4). For a sufficiently large number of interfering packets, this variable can be accurately modeled as Gaussian with mean given by (6) and variance

$$\text{var}[r^{(0)}(l)] = \text{var}[i^{(0)}(l)] + \text{var}[\tilde{n}(Nl+n)]. \quad (9)$$

The noise variance is clearly $\text{var}[\tilde{n}(Nl+n)] = N_0/N$. To estimate the latter, let as temporarily suppose that all users are received with equal power $\text{E}[(d^{(j)}(l))^2] = P_0$ (i.e., a power-controlled scenario). Consider that m (out of the J possible) users are active at the same time with U_0, and that this number does not change during the duration of a packet. Then, only m out of the J products $p(Nl+n-T_j)p(Nl+n-T_0)$ are nonzero and the interference variance conditioned on the number of active users is

$$\text{var}[i^{(0)}(l)|m] = \frac{mP_0}{N}. \quad (10)$$

The conditional signal to interference plus noise ratio (SINR) will thus be given as [c.f. (9) and (10)]

$$\gamma(m) = \frac{P_0}{mP_0/N + N_0/N} \approx N/m, \quad (11)$$

where the approximation is valid for $mP_0 \gg N_0$.

Translating the SINR $\gamma(m)$ to the packet error probability $P_e(0|m)$ of user 0 given that m users are active, requires specifying the packet transmission format. In particular, $P_e(0|m)$ depends on the type of FEC used. Convolutional Coding (CC) is typically employed for FEC in this context. As a simple illustrative example, for rate $1/2$ CC with constraint length 3 and generator polynomials $g_0 = 7$, $g_1 = 5$ (in octal) the packet error probability is bounded by [9, sec. 5.4]

$$P_e(0|n) < \frac{L(e^{-\gamma(n)})^5}{1 - 2e^{-\gamma(n)}} \approx \frac{Le^{-5N/n}}{1 - 2e^{-N/n}}. \quad (12)$$

We remark that the specific CC used is just an illustrative example; in fact, the CC chosen not a particularly powerful one since constraint lengths up to 9 are typically used in practice.

In the *dominant* system, the probability $P_e(0)$ that U_0's transmission is lost due to interference can be computed by conditioning on the number of interfering packets and averaging as follows

$$P_e(0) = \sum_{n=0}^{J} \binom{J}{n} p^n (1-p)^{J-n} P_e(0|n), \quad (13)$$

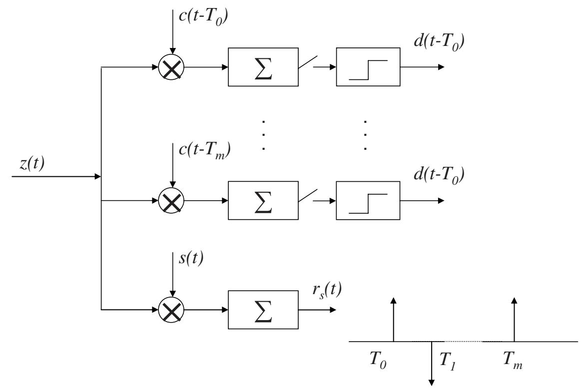

Fig. 3. Separate despreading circuitry is required for each path, but the presence of packets can be detected with a single continuous correlator

because all queues are effectively continuously backlogged. This makes it possible to average the above conditional probability with the binomial distribution, irrespective of the state of other queues. Combining (8) with (13), the probability that a packet transmitted by U_0 is lost can be written as

$$P_l(0) = P_c(0) + P_e(0)[1 - P_c(0)]. \quad (14)$$

Using the dominant system approach ([8]; see also [2]), it can be shown that the per-user throughput of our SS-RA protocol is given by

$$(J+1)\lambda < \mu := p[1 - P_l(0)]. \quad (15)$$

It is interesting to consider how μ in (15) varies with the packet length. As $L \to 1$, the packet error probability $P_e(0)$ in (13) is very small, and the throughput is only limited by collisions. In such a case, the throughput is approximately given by

$$\mu \approx p\left(1 - \frac{p}{NL}\right)^J, \quad (16)$$

and the proposed protocol behaves like slotted Aloha.

As $L \to \infty$, it is the collision probability in (8) that becomes negligible, and μ is well approximated by

$$\mu \approx p\left(1 - \sum_{n=0}^{J} \binom{J}{n} p^n (1-p)^{J-n} P_e(0|n)\right), \quad (17)$$

where $P_e(j|n)$ can be obtained as in (12).

IV. Synchronization Issues

User despreading is performed as depicted in Fig 3. For each of the active users, the BS multiplies the received sequence by the corresponding delayed version of the shared code $\mathbf{c}$, adds N chips and compares to a threshold. However, obtaining the required set of delays $\{T_j\}_{j=0}^m$ remains an issue to be addressed.

To this end, let us notice that the first symbol in every packet is always spread by the same set of chips. Upon defining the (short) periodic sequence

$$s(t) = c(t \bmod N), \quad (18)$$

which amounts to periodically repeating the first symbol chips, the output of a continuous correlator matched to $s(t)$ can be

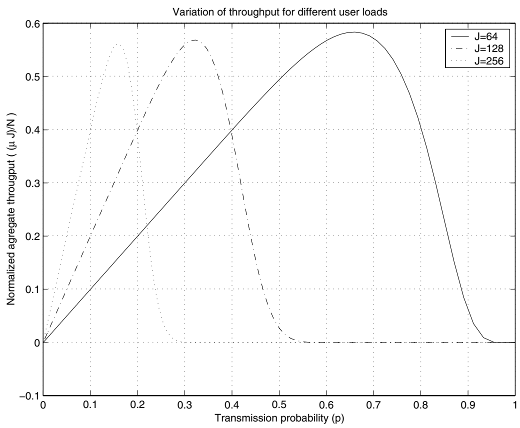

Fig. 4. The maximum normalized aggregate throughput is more than three times that of Spread Aloha, and changes slightly with user load ($N = 64$, $L = 100$).

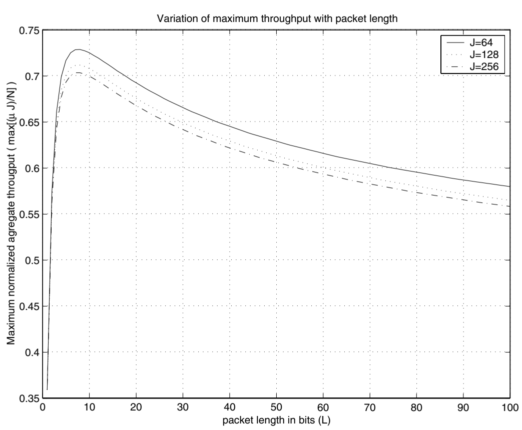

Fig. 5. There exists a certain packet length that maximizes the maximum aggregate throughput.

used to detect the beginning of a packet. Indeed, this correlator output is given by

$$r_s(t) = \sum_{k=t-N}^{t} c(k)z(k). \tag{19}$$

The event $|r_s(t)| > \tau$ can be used to identify the starting time of the $(m+1)^{st}$ packet at $T_{m+1} = t$. Thus, while we require as many correlators as the number of expected active users plus 1 (i.e., $m+1$ correlators), a simple synchronization procedure allows one to detect packet starting times. This establishes that the simple synchronization properties of Spread Aloha are retained by our SS-RA protocol that is based on long PN codes.

V. Numerical examples

With μ as in (15), Fig. 4 depicts the normalized aggregate throughput $\mu J/N$, for $N = 64$, $L = 100$ and different values of user loads J in the packet error probability expression (12). It can be seen that the normalized throughput reaches a maximum of roughly 0.5, more than three times the normalized throughput of Spread Aloha in [1]. Moreover, as is usually the case for random access protocols, the maximum throughput changes only slightly with the number of users.

Fig. 5 illustrates throughput dependence on packet length L. It can be seen that throughput is maximized for $L \approx 9$ corresponding to a normalized throughput of $\mu J/N \approx 0.7$. Presence of this optimum value is not surprising, since for small packet lengths the limiting factor is the collision probability given by (8), which decreases as L increases. However, inspection of (12) reveals that increasing packet lengths increases the probability that the packet is lost due to SINR-induced errors. It is worth noting that especially with the overhead present, the optimum packet length $L \approx 9$ is not realistic in practice. But since the optimum packet size depends on the type of code used, more realistic (stronger) convolutional codes are expected to move this small optimum packet size to larger (and thus practically reasonable) values.

VI. Conclusions

We have introduced a random access protocol similar to Spread Aloha, but based on long packet-periodic (as opposed to symbol-periodic) PN sequences. The latter renders interference (as opposed to collisions) the dominant limiting factor and effects a threefold increase in the maximum stable throughput.

The throughput was analyzed by adopting a dominant system approach, and a certain functional form of the packet error probability, corresponding to a simple convolutional code of constraint length 3. Using more powerful codes is an interesting future research topic as is investigating the effects of fading and node collaboration in SS-RA throughput[1].

References

[1] N. Abramson, "Multiple Access in Wireless Digital Networks," *Proceedings of the IEEE*, vol. 82, pp. 1360–1370, September 1994.

[2] G. Dimic, N. D. Sidiropoulos, and R. Zhang, "Signal processing and queuing tools for MAC-PHYS cross-layer design," *IEEE Signal Processing Magazine*, vol. 21, pp. 40–50, September 2004.

[3] E. Dinan and B. Jabbari, "Spreading codes for direct sequence CDMA and wideband CDMA cellular networks," *IEEE Communications Magazine*, vol. 36, pp. 48–54, Sept. 1998.

[4] K. Joseph and D. Raychaudhuri, "Throughput of unslotted direct-sequence spread-spectrum multiple-access channels with block FEC coding," *IEEE Transactions of Communications*, vol. 41, pp. 1373–1378, September 1993.

[5] R. K. Morrow and J. S. Lehnert, "Packet Throughput in Slotted Aloha DS/SSMA Radio Systems with Random Signature Sequences," *IEEE Transactions on Communications*, vol. 40, pp. 1223–1230, July 1992.

[6] M. B. Pursley and D. J. Taipale, "Error Probabilities for Spread-Spectrum Packet Radio with Convolutional Codes and Viterbi Decoding," *IEEE Transactions on Communications*, vol. 35, pp. 1–12, January 1987.

[7] T. Sato, H. Okada, T. Yamazato, M. Katayama, and A. Ogawa, "Throughput analysis of DS/SSMA unslotted ALOHA system with fixed packet length," *IEEE Journal of Slected Areas in Communications*, vol. 14, pp. 750–756, May 1996.

[8] B. Tsybakov and V. Mikhailov, "Ergodicity of a Slotted Aloha System," *ProbL Inf. Transm.*, vol. 15, April 1980. (translated from Russian original in ProbL Peredachi Inf. 15, 4 (Oct.-Dec. 1979), 72-87).

[9] A. J. Viterbi, *CDMA Principles of Spread Spectrum Communication*. Addison-Wesley Wireless Comms. Series, 1995.

[1] The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U. S. Government.