A code construction approaching capacity 1 for random access with multiplicity feedback*

Miklós Ruszinkó†

Computer and Automation Institute
of the Hungarian Academy of Sciences,
P.O.B. 63,
H–1518 Budapest, Hungary
h2942Rus@ella.hu

Peter Vanroose‡

K.U.Leuven,
division E.S.A.T.–MI2
Kard.Mercierlaan 94
B–3001 Heverlee, Belgium
Peter.Vanroose@esat.kuleuven.ac.be

Abstract

Pippenger [9] showed in a probabilistic way that the capacity of a collision channel with multiplicity feedback is one. In the current paper we give a constructive proof of this result.

1 The collision channel model

Consider the following communication situation, which is commonly called the (slotted) *multiple-access collision channel:* Message packets of fixed length can be transmitted over the channel, provided that they are packet-synchronized; a time slot corresponds to the transmission length of one packet, which we take as the time unit. The destination for the packet contents is a single common receiver. Senders of different packets cannot interchange information, so there are an (in principle) unlimited number of independent and non cooperating channel input users. Thus, the total packet traffic can be modeled as a Poisson process in time with, say, intensity λ.

When two or more users send a packet in the same time slot, these packets "collide" and the packet information is lost, i.e., the receiver cannot determine the packet contents, and

*This work was supported in part by the scientific exchange program between the Royal Belgian Academy of Sciences and the Hungarian Academy of Sciences.

†Miklós Ruszinkó is supported by the Hungarian National Foundation for Scientific Research, Grant T4271, and a grant of the "Magyar Tudományért" Foundation

‡Peter Vanroose is supported by a postdoctoral research grant of the Belgian National Fund for Scientific Research (NFWO)

retransmission will be necessary. However, collisions give some information: all users, also those who were not transmitting, are informed immediately of the multiplicity of the collision. Thus, multiplicity 1 means successful transmission by a single user, while multiplicity > 1 means that retransmission will be necessary.

A *conflict resolution protocol* is a retransmission scheme for the packets in a collision. Such a scheme must insure the eventual successful transmission of all these packets. We consider *blocked access* which means that during the resolution of one particular collision, all users which were *not* involved in it are not allowed to transmit. This "blocking period" is called an *epoch*. Thus, an epoch consists of the time slots needed by the conflict resolution protocol to (at least partially) resolve one collision. The users involved in the initial collision will be called the *active users* of that epoch.

Clearly, because of the Poisson arrival of messages, i.e., of new users, message packets waiting for transmission will accumulate during the epoch. These packets will all be transmitted in the following epoch, together with packets which were not resolved in the epoch. It is of course important that the maximum transmission delay, i.e., the time between an initial collision and the end of the epoch in which the last of its packets is resolved, must be finite with probability one. (Because of the probabilistic (Poisson) model for the message arrivals, this is the best one can do.) To this end, it suffices that the number of active users in subsequent epochs form a stable Markov chain [6].

The supremum of the set of intensities λ for which a certain protocol still gives raise to a stable Markov chain of active users is called its *throughput*. It is the relative fraction of time slots with successful transmission. A throughput of 1 means full channel utilization; the difference between the epoch length and the number of packets resolved, divided by the epoch length, is arbitrarily small. The *capacity* of a certain collision channel is the supremum of achievable throughputs, over all possible protocols. Clearly, the capacity can never be larger than 1.

The study of collision channels, also called random-access channels, started with Abramson's ALOHA system [5], which is non-slotted and uses only binary feedback (collision/no collision). An unstable throughput of $(2e)^{-1} \approx 0.18394$ was achieved. Later work for the *slotted* channel gave better throughputs (0.375 [6], 0.429 [7], 0.48775 [11]). The best upper bound to the capacity is currently 0.587 [12], but see also [9]. Also the collision channel without feedback [13, 15] and the collision channel with bounded multiplicity feedback [10] were studied.

The collision channel with multiplicity feedback was first considered by Tsybakov [8], who gave a protocol with throughput 0.533. Pippenger [9] proved that the capacity of the collision channel with multiplicity feedback is 1. Thus, if an arbitrary number of users are allowed to send fixed-length packets over a slotted multiple-access channel independently, and if the packet generation is a Poisson process with intensity $\lambda < 1$, then all packets can be sent with finite delay (with probability one). Pippenger's proof uses a random coding argument and is thus non-constructive: it does not produce a conflict resolution protocol with throughput 1.

In this paper we will present a protocol which has throughput arbitrarily close to 1, thus giving a constructive proof for Pippenger's result.

2 Definitions, notations, preliminaries

We will denote natural numbers by small letters n, n_1, k, t, m, a_i, x_i, y_i. Row vectors will be denoted by bold letters: $\boldsymbol{x}$, $\boldsymbol{y}$, $\boldsymbol{y_1}$, while for matrices we will use capital letters A, A_1, I. The script capital $\mathcal{L}$ will be used for a set of integers. $\mathbb{E}()$ will denote expectation.

The number t will denote the number of active users in a certain epoch, i.e., the number of users involved in the initial collision for the epoch. The number m will denote the number of active users for which the packet is resolved in the current epoch. Thus, $t-m$ users must retransmit in the next epoch. The total number of time slots in an epoch will be n.

An important achievement of Pippenger's paper is that he reduces the problem of determining the maximal achievable throughput to a combinatorial problem, related to the coin-weighing problem of Erdös and Rényi [2]. This coin-weighing problem was generalized and solved by Lindström [3, 4], by means of a set of *detecting matrices* given by the following

Construction: [4] for each nonnegative integer t and any fixed k:
a binary $t \times n$ matrix A can be constructed explicitly, with $n = \mathcal{O}(t/\log_k t)$, such that the n-tuple $\boldsymbol{y} = \boldsymbol{x} \cdot A$ uniquely determines the k-ary t-tuple $\boldsymbol{x}$.

In 1989 Györfi [14] pointed out a possible way to a constructive proof of Pippenger's result, based on the existence of a slightly different type of detecting matrices, as formulated in the following

Problem: [14]
For each integer $t > 0$, find a binary $t \times n$ matrix A, where $n = \mathcal{O}(t/\log t)$, such that the n-tuple $\boldsymbol{y} = [x_1 x_2 \cdots x_t] \cdot A$ uniquely determines the nonnegative integers $x_1, x_2, \ldots, x_t \in \{0, 1, \ldots, t\}$ satisfying $\sum_{i=1}^{t} x_i \leq t$.

It follows from Pippenger's proof that such a family of matrices must exist. Unfortunately, no constructions have been found until now.

We follow a somewhat different track in order to construct a protocol with throughput 1. Instead of a completely parallel search procedure, using a detecting matrix A which only depends on the number of active users, our matrix A will depend on feedback obtained during an initial part of the protocol. We will use Lindström's construction for the matrix A.

3 The collision channel protocol

All channel users are assumed to follow what is going on on the channel, so everybody not involved in a certain initial collision has to wait until the end of the epoch before transmitting his information packet (blocked access). All users can find out the end of the epoch, because the protocol is known to everybody. Therefore, it is sufficient to describe the protocol for one epoch. We do this in five consecutive steps, where for each step j we calculate the number n_j of time slots needed:

1. In the initial time slot, each active user sends his packet. Thus,

$$n_1 = 1,$$

and after this, all users know the number of transmitted packets, i.e., the number of active users, because that number is the feedback they receive. If this number is 0 or 1, the epoch ends.

Suppose there are $t \geq 2$ active users, each of which selects, randomly and uniformly, a number from $\{1, 2, \ldots, t\}$. These numbers $i_1, \ldots, i_t$ are called the identifiers. Let $x_i := \#\{j : i_j = i\}$ be the number of active users which chose the identifier i. Denote by m the number of users with an identifier for which $x_i = 1$. Only these m users will be able to successfully transmit. To this end, the users must determine the values x_i. Clearly, m is a random variable, depending on t.

2. Group the identifiers in intervals of length $\log t$: $\mathcal{L}_k := \{i_j | (k-1)\log t < i_j \leq k \log t\}$. In the k-th time slot, users with an identifier in $\mathcal{L}_k$ resend their packet. Thus,

$$n_2 = \lceil \frac{t}{\log t} \rceil.$$

3. Consider first the groups $\mathcal{L}_k$ for which the feedback in the k-th slot exceeds $(\log t)^2$. There are certainly less than $\frac{t}{(\log t)^2}$ such groups. In a reserved way all users with an identifier belonging to these groups resend their packet (thus learning their x_i), after which they leave the system.

$$n_3 < \frac{t}{(\log t)^2} \log t = \frac{t}{\log t}.$$

4. For the remaining users, certainly $x_i \leq (\log t)^2$, so construct a detecting matrix A with alphabet size $k = \lfloor (\log t)^2 \rfloor$, and determine $\boldsymbol{x}$ from $\boldsymbol{y} = \boldsymbol{x}A$, where $\boldsymbol{y}$ is the feedback sequence of length

$$n_4 \leq \mathcal{O}(\frac{t}{\log_k t}) = \mathcal{O}(t \frac{\log \log t}{\log t}).$$

5. Now, all users know the multiplicities x_i by which the identifiers i were chosen. In a reserved way, only the active users with identifiers $x_i = 1$ resend their packets:

$$n_5 \leq m.$$

The users which left the system in step three with a collision ($x_1 > 1$), and the users in step five with $x_i > 1$ must try again in the next epoch.

Thus, exactly m packets are successfully transmitted in $n = n_1 + \cdots + n_5$ time slots. The expected epoch length, given t, is

$$\mathbb{E}(n|t) \leq \mathbb{E}(m|t) + 1 + 2\lceil\frac{t}{\log t}\rceil + \mathcal{O}(t\frac{\log\log t}{\log t}) = \mathbb{E}(m|t) + \mathcal{O}(t\frac{\log\log t}{\log t}).$$

$\mathbb{E}(m|t) = t \cdot (1 - 1/t)^{t-1}$, since it is t times the probability that i_1 differs from all $i_2, \ldots, i_t$.

It is easy to see that the sequence $\{N_l\}$ of numbers of active users in subsequent epochs form an irreducible and aperiodic Markov chain. These two conditions, together with positive recurrence, imply stability. Foster's criterion [1] guarantees positive recurrence if $\mathbb{E}(N_{l+1}|N_l = k)$ is bounded for small k, and if $\mathbb{E}(N_{l+1} - N_l|N_l = k)$ is negative for large k.

By [6, Theorem 4.2], the throughput of this scheme is arbitrarily close to the asymptotic

$$\begin{aligned}
R &= \lim_{t\to\infty} \sup \frac{\mathbb{E}(m|t)}{\mathbb{E}(n|t)} \\
&= \lim_{t\to\infty} \frac{t \cdot (1 - 1/t)^{t-1}}{\mathbb{E}(n - m|t) + t \cdot (1 - 1/t)^{t-1}} \\
&\geq \lim_{t\to\infty} \frac{te^{-1}}{\mathcal{O}(t \log\log t/ \log t) + te^{-1}} \\
&= \lim_{t\to\infty} \frac{1}{e \cdot \mathcal{O}(\log\log t/ \log t) + 1} \\
&= 1.
\end{aligned}$$

4 Acknowledgment

We would like to acknowledge A. Méhes, whose master thesis (*Multiple Access Channels*, University of Budapest, 1993) gives a very good overview of the collision channel and of combinatorial and information-theoretic methods used to prove capacity results.

References

[1] F.G. Foster, "On stochastic matrices associated with certain queueing problems", *Ann. Math. Stat.* 24, pp. 355–360, 1953.

[2] P. Erdös & A. Rényi, "On two problems of information theory", *Publ. Math. Inst. Hung. Ac. Sci.* 8, pp. 241–254, 1963.

[3] B. Lindström, "On a combinatory detection problem I", *Publ. Math. Inst. Hung. Ac. Sci.* 9, pp. 195–207, 1964.

[4] B. Lindström, "On a combinatorial problem in number theory", *Can. Math. Bull.* 8(4), pp. 477–490, 1965.

[5] N. Abramson, "The ALOHA system — another alternative for computer communications", in *AFIPS Conf. Proc.*, Fall Joint Computer Conf., vol. 37, pp. 281–285, 1970.

[6] B.S. Tsybakov & V.A. Mikhailov, "Free synchronous packet access in a broadcast channel with feedback", *Problemy Peredachi Informatsii* vol. 14 no. 4, pp. 32–59, 1978.

[7] J.I. Capetanakis, "Tree algorithms for packet broadcast channels", *IEEE Transactions on Information Theory*, IT–25(6), pp. 505–515, 1979.

[8] B.S. Tsybakov, "Resolution of a conflict of known multiplicity", *Problemy Peredachi Informatsii* vol. 16 no. 2, pp. 69–82, 1980.

[9] N. Pippenger, "Bounds on the performance of protocols for a multiple access broadcast channel", *IEEE Transactions on Information Theory*, IT–27(2), pp. 145–151, 1981.

[10] L. Georgiades & P. Papantoni-Kazakos, "A collision resolution protocol for random access channels with energy detectors", *IEEE Transactions on Communications*, COM–30(11), pp. 2413–2420, 1982.

[11] J. Mosely & P.A. Humblet, "A class of efficient contention resolution algorithms for multiple-access channels", *IEEE Transactions on Communications*, COM–33(2), pp. 145–151, 1985.

[12] B.S. Tsybakov, "Survey of USSR contributions to random multiple-access communications", *IEEE Transactions on Information Theory*, IT–31(2), pp. 143–164, 1985.

[13] J.L. Massey & P. Mathys, "The collision channel without feedback", *IEEE Transactions on Information Theory*, IT–31(2), pp. 192–204, 1985.

[14] L. Györfi, "A code for multiple access adder channel", in *Open problems of the Fourth Joint Swedish-Soviet International Workshop on Information Theory*, Gotland, Sweden, Aug. 1989, pp. 13–15.

[15] L. Györfi & I. Vajda, "Constructions of protocol sequences for multiple access collision channels without feedback", *IEEE Transactions on Information Theory*, IT–39(5), pp. 1762–1765, 1993.