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.