The Influence of Memory on Creating Order

Ulrich Tamm
Dept. of Mathematics
University of Bielefeld
P. O. Box 100131
33501 Bielefeld, Germany

Abstract. A simple strategy for creating order in a special multi-user model is analyzed. In each time unit $s \geq 2$ bits arrive at an organizer who in the same time unit has to choose one bit as his output, preferably he will choose a 1, which may be one of the arriving bits or a 1 stored in some memory device. Contrasting to the original model for creating order introduced in [1] the number of bits in the memory device now may vary in time. We shall study the influence of the maximum size of the memory device on the expected occurrence of the first 0 in the sequence produced by the organizer.

I. Introduction

Ahlswede, Ye, and Zhang [1] introduced the following model for creating order in sequence spaces. We are given a box containing (a fixed number) β balls labelled by letters from an alphabet of size α. In each time unit a person $\mathcal{O}$ – denoted as organizer – takes out one ball of the box which is replaced by a new ball thrown into the box by a second person $\mathcal{I}$. The aim of the organizer is to reduce the space of possible output sequences. As a measure for the efficiency of the ordering process the number of possible output sequences and the entropy of the output space have been studied (cf. also [2], [3]). A related model in which the output sequence is regarded as a message from person $\mathcal{I}$ to a decoder $\mathcal{D}$ was considered by several authors studying the permuting channel, e. g. [4], [5], and [6].

In this paper a multi–user version of the original model for creating order is discussed. Now there are $s \geq 2$ persons $\mathcal{I}_1, \ldots, \mathcal{I}_s$, say, throwing balls labelled either 0 or 1 into the box. We shall present the model using a slightly different terminology. In each time unit s sources $\mathcal{I}_1, \ldots, \mathcal{I}_s$ produce one bit each. These s bits arrive at an organizer who in the same time unit has to choose one bit for output. This bit may be one of the s arriving bits or a bit stored in some memory device (the box), in which the bits not used so far may be stored. The organizer follows a simple strategy: if it is possible the output must be a 1. So if one of the arriving bits is a 1, the organizer will put out a 1 for sure. The bits not used for output he may store in the memory device. If all the s sources produce a 0, then the organizer will take a look at the memory. If there is still a 1 contained he will put out a 1, otherwise he must put out a 0.

We assume that in the beginning the memory device is empty. At some point it may occur that no further 1 can be stored, since the device is full of 1's (a 0 may be replaced by a 1). In this case there is a maximum size or capacity of M bits which cannot be superceded. Of theoretical interest is also the not very realistic model, in which the memory device can store infinitely many bits.

Observe that contrasting to the original model for creating order the size of the memory device (or box) now may vary in time. A natural question is: how much influence does the maximum size of the memory have on the behaviour of the sequence of bits arranged by the organizer? Of course, in the strategy considered the organizers aim is to produce the all–one sequence and we shall study how well he can manage to achieve this goal. As a new measure for the influence of the memory we consider the expected value of the first occurence of a 0 in this sequence. We shall denote this expectation as

$$E_0 = \sum_{t=1}^{\infty} t \cdot \text{Prob(first 0 at time t)}. \tag{1}$$

Further, we shall denote by

$$\underline{x}^i = (x_1^i, x_2^i, \ldots), \tag{2}$$

the sequences of bits produced by the sources $\mathcal{I}_i$, respectively and by

$$\underline{z} = (z_1, z_2, \ldots) \tag{3}$$

the sequence arranged by the organizer according to the above strategy. The results below are mainly derived from properties of the sequence

$$\underline{u} = (u_1, \ldots u_s, u_{s+1}, \ldots, u_{2s}, u_{2s+1}, \ldots) = (x_1^1, \ldots, x_1^s, x_2^1, \ldots, x_2^s, x_3^1, \ldots) \tag{4}$$

which is obtained by merging the sequences $\underline{x}^i, i = 1, \ldots, s$ into one sequence $\underline{u}$, where the bits in the positions $\equiv i \bmod s$ are those produced by $\mathcal{I}_i$. (Observe that the order of the incoming bits at each time unit does not matter, since the organizer waits until all the s bits have arrived. So $\underline{u}$ may also be defined by any other merging procedure which assigns the postions $(s-1)t+1, \ldots, st$ to the bits arriving in time unit t).
Our first result is obtained for two memoryless, symmetrical sources, i. e., $\mathcal{I}_1$ and $\mathcal{I}_2$ produce 1 and 0 with equal probability $\frac{1}{2}$.In this case it is

$$\text{Prob(first 0 at time } t-1) = \frac{1}{4^t} \cdot a(0, t-1), \tag{5}$$

where $a(0, t-1)$ denotes the number of sequences $\underline{u}$ produced by the two sources leading to the all-one sequence as output with actual memory size (= number of 1's in the device) 0 at time $t-1$. Clearly an empty memory at time $t-1$ is a necessary condition for the occurrence of the first 0 at time unit t, which then happens with probability $\frac{1}{4}$ (4^{t-1} is the number of all possible sequences $\underline{u}$ until time $t-1$).
Analogously, the numbers $a(m, t)$ are defined for actual memory size $m = 1, \ldots, M, t = 1, 2, \ldots$. Let us dnote by $\vec{u}_t = (u_{2t-1}, u_{2t})$ the two bits arriving at time t. Now observe that in each time unit the source inputs $\vec{u}_t = 10$ and $\vec{u}_t = 01$ do not change the actual size of the memory, $\vec{u}_t = 00$ decreases the memory size by one bit (and is forbidden for $m = 0$, since in this case the organizer must put out a 0) and $\vec{u}_t = 11$ increases the memory size by one bit if $m < M$ (and does not change the memory size if the capacity $m = M$ is reached. So we have a random walk in the memory size expressed by the recursion formulae for the numbers $a(m, t)$

$$\begin{pmatrix} a(0,t) \\ \vdots \\ a(M,t) \end{pmatrix} = A_{M+1} \cdot \begin{pmatrix} a(0,t-1) \\ \vdots \\ a(M,t-1) \end{pmatrix} \tag{6}$$

where

$$A_n = \begin{pmatrix} 2 & 1 & 0 & \ldots & 0 & 0 & 0 \\ 1 & 2 & 1 & \ldots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \ldots & 1 & 2 & 1 \\ 0 & 0 & 0 & \ldots & 0 & 1 & 3 \end{pmatrix}, \tag{7}$$

with initial matrices $A_1 = (3)$, $A_2 = \begin{pmatrix} 2 & 1 \\ 1 & 3 \end{pmatrix}$.

Theorem 1: If $\mathcal{I}_1$ and $\mathcal{I}_2$ are memoryless, symmetrical sources, then
i)

$$E_0 = \frac{1}{4} \sum_{i=1}^{M+1} c_i \Big(1 - \frac{\lambda_i^{(M+1)}}{4}\Big)^{-2} \tag{8}$$