ELSEVIER

Statistics & Probability Letters 42 (1999) 219–227

STATISTICS & PROBABILITY LETTERS

Sharp entropy bounds for discrete statistical simulation

Dan Romik *,[1]

School of Mathematical Sciences, Tel-Aviv University, Tel-Aviv 69978, Israel

Received April 1998; received in revised form June 1998

Abstract

We define a general procedure for simulating a given discrete distribution using a sequence of i.i.d. random variables. This procedure is used to prove that a natural information-theoretic bound on the number of samples required to simulate the distribution can be arbitrarily approached in a limiting sense. © 1999 Elsevier Science B.V. All rights reserved

Keywords: Entropy; Statistical simulation; Random number generation; Discrete distributions

1. Introduction

The subject of the simulation of one discrete distribution using another, and in particular the expected time it takes, has been investigated by several authors (e.g. Blum, 1986; Elias, 1972; Knuth and Yao, 1976; Stout and Warren, 1984; see Section 4 for discussion of the different approaches in these papers). It is perhaps not surprising that the entropies of the simulated and simulating distribution show up in these investigations. Knuth and Yao (1976) explore the simulation of distributions using fair coins, and show that the expected number of coin tosses in such a simulation is always at least the entropy of the simulated distribution, and that it is possible to simulate the distribution using on the average at most two tosses more than the entropy. This makes information-theoretic sense: a coin toss produces one bit of information, and the distribution $q_1, q_2, \ldots, q_d$ "contains" $H(q_1, q_2, \ldots, q_d)$ bits (where $H(q_1, q_2, \ldots, q_d) := -\sum_{i=1}^{d} q_i \log q_i$ is the entropy of the distribution $q_1, q_2, \ldots, q_d$); it is thus reasonable to expect, that the simulation should require at least $H(q_1, q_2, \ldots, q_d)$ tosses on the average, and that it should be possible to approach that number, in the sense that for any $\varepsilon > 0$, there exists an n such that it is possible to simulate n independent copies of $q_1, q_2, \ldots, q_d$ in such a way that the expected number of tosses, divided by n, is not greater than $H(q_1, q_2, \ldots, q_d) + \varepsilon$. Although not stated explicitly in Knuth and Yao's paper, this follows immediately from the result stated above.

Consider now a more general situation: simulation of the distribution $q_1, q_2, \ldots, q_d$ using another discrete distribution, which will be modelled by a sequence of i.i.d. r.v.s $X_1, X_2, X_3, \ldots$ distributed over a finite alphabet

* E-mail: romik@math.tau.ac.il.

[1] This paper is part of the author's M.Sc. thesis, written under the supervision of Prof. David Gilat.

PII: S0167-7152(98)00174-6

$A = \{\alpha_1, \alpha_2, \ldots, \alpha_k\}$. How many samples of the X_i's are required on the average to simulate $q_1, q_2, \ldots, q_d$? Again thinking information-theoretically, since each sample "contains" $H(X_1)$ (the entropy of X_1) bits, and we need $H(q_1, q_2, \ldots, q_d)$ bits, it is reasonable to expect, and it is the goal of this paper to prove, that the following result holds:

Theorem 1. (i) *For any simulation method of $q_1, q_2, \ldots, q_d$ using the process $X_1, X_2, \ldots$, the expected number of samples of the X_i's is not smaller than $H(q_1, q_2, \ldots, q_d)/H(X_1)$;*

(ii) *It is possible to arbitrarily approach that ratio, in the following exact sense: for any $\varepsilon > 0$, there exists an n such that it is possible to simulate n independent copies of $q_1, q_2, \ldots, q_d$ in such a way that the expected number of samples, divided by n, is not greater than $H(q_1, q_2, \ldots, q_d)/H(X_1) + \varepsilon$.*

Theorem 1 can be thought of as a theorem about the meaning of entropy as much as a theorem about simulation: compare it with the Noiseless Coding Theorem (see Abramson, 1963, pp. 72–73):

The Noiseless Coding Theorem. (i) *For any immediate code for the distribution $q_1, q_2, \ldots, q_d$ over the alphabet $\{0, 1, \ldots, k-1\}$, the expected word length is not smaller than $H(q_1, q_2, \ldots, q_d)/\log k$,*

(ii) *It is possible to arbitrarily approach this ratio, in the sense that for any $\varepsilon > 0$, there exists an n such that it is possible to code n independent copies of $q_1, q_2, \ldots, q_d$ using the alphabet $\{0, 1, \ldots, k-1\}$ in such a way that the expected word length, divided by n, is not greater than $H(q_1, q_2, \ldots, q_d)/\log k + \varepsilon$.*

The Noiseless Coding Theorem is of fundamental importance in information theory; it is the basis for the standard interpretation of the entropy of a discrete distribution as the minimal number of letters required on the average to code the distribution with the ordinary "computer science" $\{0, 1\}$-bits, or, more generally, as the minimal number of letters in the alphabet $\{0, 1, \ldots, k-1\}$ required on the average to code the distribution, provided that units are chosen so that one letter is worth 1 unit of information (i.e. $\log k = 1$), and in the extended sense where coding of multiple independent copies of the distribution is considered.

Similarly, Theorem 1 gives a new interpretation of the entropy of a distribution in terms of simulation rather than of coding: it says that the entropy of a distribution is the minimal number of fair coin tosses required on the average to simulate the distribution, or more generally, it is the minimal number of samples of any other distribution X_1 required on the average to simulate the distribution, again provided units are chosen so that one sample is worth 1 unit of information, i.e. $H(X_1) = 1$, and again in the sense where simulation of multiple independent copies of the distribution is considered.

In this paper, after giving a formal definition of a simulation method in Section 2, we shall see that equality in Theorem 1 ($E(N) \cdot H(X_1) = H(q_1, \ldots, q_d)$) is attained if and only if the simulation method "preserves information" in a natural sense that will be defined. This leads immediately to a proof of part (i). Finally, in Section 3, after briefly discussing the situation for simulations using fair coins, we generalize some of the ideas to simulations using general i.i.d. processes, culminating in the proof of part (ii).

2. Definitions and a basic equation

Let $X_1, X_2, \ldots$ be a discrete process, distributed over an alphabet A, which we shall think of as defined on the coordinate space $(\Omega, \mathscr{F}, P)$, where $\Omega = A^N$, $\mathscr{F}$ is the σ-field generated by the finite cylinder sets (see below), P is the probability measure induced by the process, and $X_n(a_1 a_2 a_3 \ldots) = a_n$. Let $q_1, q_2, \ldots, q_d$ be a discrete distribution, which for the sake of concreteness we assume to be over the alphabet $B = \{\beta_1, \beta_2, \ldots, \beta_d\}$ (both d and A are allowed to be infinite). We wish to define the concept of a simulation method of $q_1, \ldots, q_d$ using the process $X_1, X_2, \ldots$. Intuitively, this means that we sample the X_i's (graphically this can be visualized as going down the "tree" $\bigcup_{t=1}^{\infty} A^t$) until we decide to stop according to some predetermined rule and announce

the result (one of the symbols β_j), and the method produces each β_j with respective probability q_j. We start with some basic notations and definitions:

Terminology. Elements of the set $\bigcup_{t=1}^{\infty} A^t$ will be called *finite A-words*, or briefly *words* (A is the only alphabet whose words will be considered). For a word $w \in \bigcup_{t=1}^{\infty} A^t$, let $l(w)$ denote the length of w, and $p(w)$ denote the probability of w according to the process $X_1, X_2, \ldots$, i.e. $p(a_1 a_2 \ldots a_l) = P(X_1 = a_1, \ldots, X_l = a_l)$. To each $w \in \bigcup_{t=1}^{\infty} A^t$ there corresponds a *cylinder set* $\{\omega = a_1 a_2 a_3 \ldots \in \Omega\colon\ a_1 a_2 \ldots a_{l(w)} = w\}$. We shall frequently identify a word and its corresponding cylinder set, denoting both by w. A (possibly infinite) set $C \subset \bigcup_{t=1}^{\infty} A^t$ will be called a *code*; an *immediate* code is a code no word of which is a prefix of another (in terms of the corresponding cylinder sets, this is equivalent to saying that different cylinder sets of words in the code are disjoint). An (almost surely finite) *stopping time* for the process $X_1, X_2, \ldots$ is a random variable $N : \Omega \to \{1, 2, 3, \ldots\} \cup \{\infty\}$ such that for each n, the event $\{N = n\}$ is a union of (cylinder sets of) words of length n, and such that $P(N < \infty) = 1$. Finally, we shall use the letter H loosely to denote entropy – of partitions, random variables and distributions.

With these preliminaries, we can now define:

Definition. A *simulation method* for the distribution $q_1, \ldots, q_d$ using the process $X_1, X_2, \ldots$ is a triplet (N, C, f), where: N is a stopping time for the process $X_1, X_2, \ldots$; C is an immediate code such that $P(C) = \sum_{w \in C} p(w) = 1$ and for each $w \in C$, $N = l(w)$ on w; and $f : C \to B$ is a function satisfying

$$(*) \quad q_j = \sum_{w \in f^{-1}(\beta_j)} p(w), \quad j = 1, 2, \ldots, d.$$

Remark. The intuitive meaning of the definition is the following: the stopping time N represents the decision as to when to stop sampling the X_i's; the code C is a partition which represents the information available to us after stopping, and the function f determines the result of the simulation after stopping, and should be such that the simulated distribution is indeed $q_1, \ldots, q_d$, whence the requirement $(*)$. Note that, given any stopping time N, it determines uniquely (up to words of probability 0) an immediate code C with total probability 1 such that for each $w \in C$, $N = l(w)$ on w: C is simply the totality of words w with this property – for each n, the event $\{N = n\}$ is a union of words of length n, and C is comprised of those words for $n = 1, 2, \ldots$; hence $P(C) = P(N < \infty) = 1$. Conversely, any immediate code C with total probability 1 determines a stopping time N defined as $l(w)$ on each $w \in C$. Also, note that a stopping time N with an associated code C, but without a function f, can be thought of as a simulation method for the code itself, i.e. one can take $B = C$, $f = id_C$ (the identity function on C), and the distribution thus simulated is $\{p(w)\colon w \in C\}$.

We now assume for the rest of this paper that $X_1, X_2, \ldots$ are i.i.d. r.v.s. To avoid trivialities, we assume that $p(\alpha) > 0$ for all $\alpha \in A$. The following theorem establishes the fundamental equation for simulation using an i.i.d. process.

Theorem 2. *In any simulation method, we have* $E(N) \cdot H(X_1) = H(C)$ *(with the understanding that each side of the equation is finite iff the other side is finite).*

Proof. Assume first that N is bounded (or, equivalently, that C is finite) by t. Then, using standard properties of the entropy function (see Abramson, 1963), and using the fact that the partition of $X_1, \ldots, X_t$ refines the partition C, we have

$$tH(X_1) = H(X_1, \ldots, X_t) = H(X_1, \ldots, X_t, C) = H(C) + H(X_1, \ldots, X_t | C)$$

$$= H(C) + \sum_{w\in C} p(w)H(X_1,\ldots,X_t|w)$$

$$= H(C) + \sum_{w\in C} p(w)H(X_{l(w)+1},\ldots,X_t)$$

$$= H(C) + \sum_{w\in C} p(w)H(X_1)(t - l(w))$$

$$= H(C) + tH(X_1) - H(X_1)\sum_{w\in C} p(w)l(w)$$

$$= H(C) - E(N)\cdot H(X_1) + tH(X_1),$$

as claimed.

To prove the general case, consider for each t the simulation defined by the stopping time $N\wedge t$, denoting by C_t the associated code. It is easy to see that if C is the original code, then $C_t = \{w \in C: l(w)\leqslant t\}\cup\{w\in A^t : w$ is a prefix of a word in $C\}$. For each t, by the special case proved above we know that $E(N\wedge t)\cdot H(X_1)=H(C_t)$. Letting t tend to infinity, we see that $E(N\wedge t)\nearrow E(N)$, also since $-\sum_{w\in C,l(w)\leqslant t} p(w)\log p(w)\leqslant H(C_t)\leqslant H(C)$ we see that $H(C_t)\to H(C)$, and this gives the desired result. □

Remark. An alternative method for proving Theorem 2 would be to apply Wald's equation for the sequence of i.i.d. random variables $I_n = -\log p(X_n)$, where $p(X_n)$ is defined as $p(\alpha)$ on the event $\{X_n = \alpha\}$. This immediately yields $E(N)\cdot H(X_1) = H(C)$.

Corollary 1 (part (i) of Theorem 1)**.** *In any simulation, $E(N)\cdot H(X_1)\geqslant H(q_1,\ldots,q_d)$. Furthermore, the difference $E(N)\cdot H(X_1) - H(q_1,\ldots,q_d)$ can be identified as the (average) amount of information lost when different words in the code are combined by the function f into one symbol β_j. i.e.*

$$E(N)\cdot H(X_1) - H(q_1,\ldots,q_d) = \sum_{j=1}^{d} q_j H\left(\left\{\frac{p(w)}{q_j}: w\in f^{-1}(\beta_j)\right\}\right).$$

Proof.

$$\begin{aligned} E(N)\cdot H(X_1) - H(q_1,\ldots,q_d) &= H(C) - H(q_1,\ldots,q_d)\\ &= -\sum_{w\in C} p(w)\log p(w) + \sum_{j=1}^{d} q_j\log q_j\\ &= -\sum_{j=1}^{d} q_j \sum_{w\in f^{-1}(\beta_j)} \frac{p(w)}{q_j}\log p(w) + \sum_{j=1}^{d} q_j\log q_j\\ &= \sum_{j=1}^{d} q_j\left(-\sum_{w\in f^{-1}(\beta_j)} \frac{p(w)}{q_j}\log\frac{p(w)}{q_j}\right)\\ &= \sum_{j=1}^{d} q_j H\left(\left\{\frac{p(w)}{q_j}: w\in f^{-1}(\beta_j)\right\}\right)\geqslant 0. \end{aligned}$$

□

Remark. Note that this means that $E(N)=H(q_1,\ldots,q_d)/H(X_1)$ will hold if and only if the simulation method does not "discard" information, i.e. if the function f is injective. A simulation method with this property might aptly be termed *information-preserving*.

Example. As an example of the application of Theorem 2, let us calculate the entropy of a geometric random variable with parameter p: to simulate such a variable using $(p,1-p)$ coins, the natural way would be: toss the coins until you get the side corresponding to p, and the number of tosses would then be the result. The variable being simulated is the same as the stopping time for the simulation. The code is $C=\{0,10,110,1110,\ldots\}$ where the probability of 0 is p, and $f(w)=l(w)=N(w)$. This simulation method preserves information, and therefore $H(N)=E(N)\cdot H(X_1)=(1/p)H(p,1-p)$. So the entropy of a geometric r.v. with parameter p is equal to $(1/p)H(p,1-p)$, because to simulate it using a $(p,1-p)$ coin takes on the average $(1/p)$ tosses – a satisfying result.

We turn now to the proof of part (ii) of Theorem 1. To prove the claim, we need to find a "good" simulation method of $q_1,\ldots,q_d$ using the i.i.d. process $X_1,X_2,\ldots$.

3. A simulation method

In order to develop a simulation method for discrete distributions using a general i.i.d. source, let us first consider briefly the case where $X_1,X_2,\ldots$ are fair coins, i.e. $A=\{0,1\}$ and $p(w)=2^{-l(w)}$. Fair coins are literally the "standard currency" of information theory. Thus as we might expect, this special case is simpler to treat than the general one – there is no need to "convert" our currency to the standard one first. By considering fair coins first, we will gain insights that will prove helpful in treating the general situation.

Knuth and Yao (1976) give a simple characterization of the optimal simulation method for $q_1,\ldots,q_d$ using fair coins, and show that it is possible to construct this simulation method. However, this method does not lend itself to generalization to other i.i.d. sources. We describe a different method using fair coins which does; the method is as follows: define $t_0=0$, $t_1=q_1$, $t_2=q_1+q_2,\ldots,t_d=q_1+\cdots+q_d=1$. Thus $[0,1)=[t_0,t_1)\cup[t_1,t_2)\cup\cdots\cup[t_{d-1},t_d)$. The samples of the coin generate the binary expansion of a number $x\in[0,1)$, that is, $U=\sum_{n=1}^{\infty}X_n2^{-n}$ is distributed uniformly in $[0,1)$. The algorithm is, toss the coin until you know which of the intervals $[t_{j-1},t_j)$ the number U ends up in, and then the result is the corresponding β_j. In effect, this means: represent each interval $[t_{j-1},t_j)$ (whose length is q_j) as a disjoint union of dyadic intervals of maximal size.

We have used here binary expansion in the unit interval. To generalize this idea, it is necessary to understand more clearly how it works. In measure-theoretic terminology, one can say that we are using an isomorphism between the interval $[0,1)$ (with Lebesgue measure) and the space $\{0,1\}^N$ with the probability measure of fair coin tosses to build the simulation; we know how to partition $[0,1)$ into sets of respective measures q_j, and how to represent each of these sets as a union of dyadic intervals (which correspond to cylinder sets, or finite words, in $\{0,1\}^N$), and this knowledge, when phrased in terms of $\{0,1\}^N$, leads exactly to the desired simulation method. Thus, to generalize this to simulation using any i.i.d. process, we need to find a measure-theoretic isomorphism between $[0,1)$ (with Lebesgue measure) and $\Omega=A^N$ with the measure P induced by the process. Furthermore, this isomorphism, in order to be of use to us, has to admit a representation of any interval $[a,b)$ (or more precisely, the image thereof) as a disjoint union of cylinder sets.

The idea is as follows: Let $k=|A|$. Define an increasing sequence of partitions $\mathscr{P}_n$ of the interval $[0,1)$ (increasing in the order of refinement), such that for each n, the partition $\mathscr{P}_n=\{I_w^n: w\in A^n\}$ consists of k^n intervals indexed by the words of length n, the length of each I_w^n is equal to $p(w)$, and $I_w^n=\bigcup_{\alpha\in A}I_{w\alpha}^{n+1}$ (disjoint union). The isomorphism will match each I_w^n to the cylinder set corresponding to w, and it is easy to see that the requirements above are both sufficient and necessary for it to be an isomorphism. The construction of the

partitions $\mathscr{P}_n$ is by induction: To define $\mathscr{P}_1$, divide $[0,1)$ into intervals I_α^1 ($\alpha \in A$) of respective lengths $p(\alpha)$; next, assuming that $\mathscr{P}_n$ is defined, define $\mathscr{P}_{n+1}$ by dividing each interval I_w^n into intervals $I_{w\alpha}^{n+1}$ ($\alpha \in A$) of respective lengths $p(w\alpha)$. This is possible because $p(w) = \sum_{\alpha\in A} p(w\alpha)$. Note that there are many ways to carry out this construction; it will be convenient to single out one specific way, as follows: if $A = \{\alpha_i\}_{i=1}^k$, then the division of I_w^n into a union of $I_{w\alpha}^{n+1}$ is performed such that $I_{w\alpha_1}^{n+1}$ is the leftmost interval, $I_{w\alpha_2}^{n+1}$ the second leftmost, etc., and $I_{w\alpha_k}^{n+1}$ the rightmost interval (or, if $k=\infty$, then there is no rightmost interval).

Now that the partitions are defined, we can simulate as before: sample the X_i's, generating more and more letters of an infinite word $\omega = a_1a_2a_3a_4\ldots$; each finite subword $a_1\ldots a_n$ determines an interval $I_{a_1\ldots a_n}^n$, and the infinite word ω determines a single point $x \in [0,1)$ which is the intersection of all the intervals: $\{x\} = \bigcap_{n=1}^\infty I_{a_1\ldots a_n}^n$ (in fact, this is the isomorphism that we have defined as a point transformation: $\omega \leftrightarrow x$). The algorithm is to sample the process until you know which of the intervals $[t_{j-1}, t_j)$ the point x will end up in (where as before $t_j = q_1 + q_2 + \cdots + q_j$), and then the result is the corresponding β_j. In other words, the simulation stops as soon as the interval $I_{a_1\ldots a_n}^n$ is contained in one of the intervals $[t_{j-1}, t_j)$. Thus, as before, each $[t_{j-1}, t_j)$ is represented as a disjoint union of "cylinder sets" I_w^n.

Note that just as any infinite word determines a point $x \in [0,1)$, any such point x determines uniquely an infinite word $\omega = a_1a_2a_3\ldots \in A^N$, by the requirement that $x \in \bigcap_{n=1}^\infty I_{a_1\ldots a_n}^n$. We call ω the *expansion* of x with respect to the sequence of partitions $\mathscr{P}_n$ – this is a generalization of the ordinary binary expansion. This remark now enables us to describe the code of this simulation: denote by $\prec$ the order relation on A^N induced by the usual order relation on $[0,1)$ via the isomorphism $x \to$ expansion of x; this is simply the lexicographical order on A^N induced by the order $\alpha_1 \prec \alpha_2 \prec \cdots \prec \alpha_k$ on A, and can also be thought of as a partial order on $\bigcup_{t=1}^\infty A^t$, two words being comparable if one is not a prefix of the other. For each $j = 0, 1, \ldots, d$ let $\omega_j = a_{j,1}a_{j,2}a_{j,3}\ldots \in A^N$ be the expansion of t_j. Then for a word $w \in \bigcup_{t=1}^\infty A^t$, we will have $I_w^{l(w)} \subset [t_{j-1}, t_j)$ if and only if $\omega_{j-1} \prec w \prec \omega_j$. For w to be in the code and to give the result β_j, this has to hold, and should not hold for the word w' defined as w truncated by one letter: w' must not be comparable to either ω_{j-1} or ω_j, i.e. either $w' = a_{j-1,1}a_{j-1,2}\ldots a_{j-1,l(w)-1}$ or $w' = a_{j,1}a_{j,2}\ldots a_{j,l(w)-1}$; in the first case we must have $a_{j-1,\,l(w)} \prec \alpha$, where $w = w'\alpha$, and in the latter $\alpha \prec a_{j,l(w)}$; in both cases

$$a_{j-1,1}a_{j-1,2}\ldots a_{j-1,l(w)-1} \prec a_{j,1}a_{j,2}\ldots a_{j,\,l(w)-1}$$

must hold. This can be summarized in:

Definition. For each $j = 1, \ldots, d$, let v_j be the maximal number such that

$$a_{j-1,1}a_{j-1,2}\ldots a_{j-1,v_j} = a_{j,1}a_{j,2}\ldots a_{j,v_j}.$$

Then the *Unit Interval* simulation method is defined by the code $C = \bigcup_{j=1}^d C_j$, where for each j,

$$C_j = \bigcup_{n=v_j+1}^{\infty} \{a_{j-1,1}a_{j-1,2}\ldots a_{j-1,n-1}\alpha \in A^n\colon a_{j-1,n} \prec \alpha\} \cup \bigcup_{n=v_j+1}^{\infty} \{a_{j,1}a_{j,2}\ldots a_{j,n-1}\alpha \in A^n\colon \alpha \prec a_{j,n}\}$$

and by the function $f : C \to B$ defined as β_j on C_j ($j = 1, 2, \ldots, d$).

Lemma 1. *The Unit Interval simulation method simulates* $q_1, q_2, \ldots, q_d$.

Proof.

$$P(C_j) = P(\{\omega \in \Omega\colon \omega_{j-1} \prec \omega \prec \omega_j\}) = \text{Lebesgue measure of } (t_{j-1}, t_j) = q_j. \qquad \square$$

We are now in a position to prove:

Theorem 3. *If $k < \infty$, then there exists a constant c, depending on the distribution of X_1, such that for any distribution $\{q_j\}_{j=1}^d$, if N is the stopping time of the Unit Interval simulation method, then*

$$E(N) \cdot H(X_1) \leqslant H(q_1, \ldots, q_d) + c.$$

Proof. We assume that $H(q_1, \ldots, q_d) < \infty$, since the claim is vacuous otherwise. For each j, denote

$$\bigcup_{n=v_j+1}^{\infty} \{a_{j-1,1}a_{j-1,2}\ldots a_{j-1,n-1}\alpha \in A^n \colon a_{j-1,n} \prec \alpha\} = \{w_{j,n}\}_{n=1}^{\infty},$$

where $w_{j,n}$ are ordered by increasing length, and similarly

$$\bigcup_{n=v_j+1}^{\infty} \{a_{j,1}a_{j,2}\ldots a_{j,n-1}\alpha \in A^n \colon \alpha \prec a_{j,n}\} = \{w'_{j,n}\}_{n=1}^{\infty},$$

so that $C_j = \{w_{j,n}\}_{n=1}^{\infty} \cup \{w'_{j,n}\}_{n=1}^{\infty}$. It is clear that $l(w_{j,n}) \geqslant l(w_{j,1}) + \lfloor (n-1)/(k-1) \rfloor$, where $\lfloor x \rfloor$ is the greatest integer $\leqslant x$. Since

$$w_{j,1} = a_{j-1,1}a_{j-1,2}\ldots a_{j-1,l(w_{j,1})-1}\alpha, \quad w_{j,n} = a_{j-1,1}a_{j-1,2}\ldots a_{j-1,l(w_{j,n})-1}\beta$$

for some $\alpha, \beta \in A$, we have

$$p(w_{j,n}) \leqslant c \cdot e^{l(w_{j,n})-l(w_{j,1})+1} \cdot p(w_{j,1}) \leqslant c \cdot e^{l(w_{j,n})-l(w_{,1})+1} \cdot q_j,$$

where $e = \max_{\alpha \in A} p(\alpha)$, $c = (\min_{\alpha \in A} p(\alpha))^{-1}$, and since $l(w_{j,n}) - l(w_{j,1})$ grows at least linearly in n, there exist other constants $r > 0, 0 < s < 1$ such that for all n, $p(w_{j,n}) \leqslant r \cdot s^n \cdot q_j$. Likewise, we have $p(w'_{j,n}) \leqslant r \cdot s^n \cdot q_j$. Note that the constants r, s depend only on the distribution of X_1, and not on j or on the distribution $q_1, \ldots, q_d$. Now, let $C_j = \{u_{j,n}\}_{n=1}^{\infty}$, where $u_{j,n}$ are obtained by interlacing $w_{j,n}$ and $w'_{j,n}$, i.e. $u_{j,2n-1} = w_{j,n}$ and $u_{j,2n} = w'_{j,n}$. Clearly, by replacing r and s by other constants, the inequality

$$p(u_{j,n}) \leqslant r \cdot s^n \cdot q_j$$

will hold for all j, n, again with constants depending only on the distribution of X_1.

By Corollary 1, the difference $E(N) \cdot H(X_1) - H(q_1, \ldots, q_d)$ represents the loss of information when for every j, $\{u_{j,n}\}_{n=1}^{\infty}$ are combined into β_j, namely

$$E(N) \cdot H(X_1) - H(q_1, \ldots, q_d) = -\sum_{j=1}^{d} q_j \sum_{n=1}^{\infty} \frac{p(u_{j,n})}{q_j} \log \frac{p(u_{j,n})}{q_j}.$$

Since this is a convex combination of numbers, it is enough to show that for every j, $-\sum_{n=1}^{\infty} p(u_{j,n})/q_j \log p(u_{j,n})/q_j$ is bounded by a constant which depends neither on j nor on the distribution $q_1, \ldots, q_d$. Using the well-known inequality $-\sum_n a_n \log a_n \leqslant -\sum_n a_n \log b_n$, which holds provided that $\sum_n a_n = 1, \sum_n b_n \leqslant 1$ (see Abramson, 1963, p.16), we have

$$-\sum_{n=1}^{\infty} \frac{p(u_{j,n})}{q_j} \log \frac{p(u_{j,n})}{q_j} \leqslant -\sum_{n=1}^{\infty} \frac{p(u_{j,n})}{q_j} \log 2^{-n} = \sum_{n=1}^{\infty} \frac{p(u_{j,n})}{q_j} \cdot n \leqslant \sum_{n=1}^{\infty} r \cdot s^n \cdot n,$$

and thus we can take as our constant $c = \sum_{n=1}^{\infty} r \cdot s^n \cdot n < \infty$. □

Corollary 2 (Part (ii) of Theorem 1)**.** *For any distribution $q_1, \ldots, q_d$ (with finite entropy) and for any $\varepsilon > 0$, there exists an n such that it is possible to simulate n independent copies of $q_1, \ldots, q_d$ using the process $X_1, X_2, \ldots$ with a stopping time N, such that $(1/n)E(N) \leqslant H(q_1, \ldots, q_d)/H(X_1) + \varepsilon$.*

Proof. Consider first the case $k < \infty$: take $n \geqslant c/\varepsilon \cdot H(X_1)$, where c is the constant from Theorem 3. Then using the Unit Interval simulation method to simulate n independent copies of $q_1, \ldots, q_d$, we have

$$E(N) \cdot H(X_1) \leqslant n \cdot H(q_1, \ldots, q_d) + c \leqslant n \cdot H(q_1, \ldots, q_d) + n \cdot \varepsilon \cdot H(X_1),$$

and therefore $(1/n)E(N) \leqslant H(q_1, \ldots, q_d)/H(X_1) + \varepsilon$. For the case $k = \infty$: again there are two possibilities, $H(X_1) < \infty$ and $H(X_1) = \infty$. If $H(X_1) = -\sum_{j=1}^{\infty} p(\alpha_j) \log p(\alpha_j) < \infty$, then: for some small enough $\delta > 0$ to be determined later, there exists an m such that the process $X_1', X_2', \ldots$ obtained from $X_1, X_2, \ldots$ by identifying the symbols $\alpha_{m+1}, \alpha_{m+2}, \alpha_{m+3}, \ldots$ into one symbol α_m^* has entropy $H(X_1') \geqslant H(X_1) - \delta$. The process $X_1', X_2', \ldots$ has a finite alphabet $A^* = \{\alpha_1, \alpha_2, \ldots, \alpha_{m-1}, \alpha_m^*\}$, and therefore by what we proved, there exists an n such that it is possible to simulate n independent copies of $q_1, \ldots, q_d$ using $X_1', X_2', \ldots$ with a stopping time N such that

$$\frac{1}{n}E(N) \leqslant \frac{H(q_1, \ldots, q_d)}{H(X_1')} + \frac{\varepsilon}{2} \leqslant \frac{H(q_1, \ldots, q_d)}{H(X_1) - \delta} + \frac{\varepsilon}{2},$$

and if δ is chosen to be sufficiently small, then $(1/n)E(N) \leqslant H(q_1, \ldots, q_d)/H(X_1) + \varepsilon$ will hold; but simulation using $X_1', X_2', \ldots$ can be thought of as a simulation using $X_1, X_2, \ldots$, since the former are a function of the latter. The case where $H(X_1) = \infty$ is dealt with in a similar manner. □

4. Additional comments and questions

1. The approach towards simulation taken in this paper is similar to that taken by Knuth and Yao (1976), although different terminology has been used. Other authors (Blum, 1986; Elias, 1972; Stout and Warren, 1984) have considered the generation of a *random* number of independent copies of the distribution $q_1, \ldots, q_d$ using a *fixed* number n of i.i.d. r.v.s $X_1, X_2, \ldots, X_n$, concentrating mainly on the case where the X_i's are biased coins and the goal is to generate fair coins. In the framework set here, this can be described as a simulation method using biased coins with a constant stopping time, and the simulated distribution is of a pair of r.v.s (A, B) such that, given that $\{A = m\}$, B is distributed uniformly on $\{1, 2, 3, \ldots, 2^m\}$. The natural measure for the efficiency of such a method is the (average) number of fair coins produced per biased coin used, namely $E(A)/n$. Using Corollary 1, we see that

$$n \cdot H(X_1) = E(N) \cdot H(X_1) \geqslant H(A, B) = H(A) + H(B|A) = H(A) + E(A) \geqslant E(A),$$

and therefore $E(A)/n \leqslant H(X_1)$, a result similar in nature to part (i) of Theorem 1. Elias (1972) shows that it is possible to approach the bound $H(X_1)$ in the limit as $n \to \infty$.

2. So far we have discussed simulation using i.i.d. processes. The natural question now arises, whether Theorem 1 stays true for simulation using more general Markov and stationary ergodic processes, when $H(X_1)$ is replaced by the entropy of the process. It should be noted that part (i) in its present form cannot stay true in such a setting: for example, let $X_1, X_2, \ldots$ be a non-i.i.d. stationary Markov process, then one can simulate the stationary distribution X_1 with a stopping time of 1 in the obvious way, and then $H(X) = H(X) \cdot E(N) < H(X_1)$. However, we can expect an *asymptotic* lower bound on $E(N)$ to remain true. We conjecture the following:

Conjecture. *Let $X_1, X_2, \ldots$ be a finite-state stationary ergodic process, and $q_1, q_2, \ldots, q_d$ a discrete distribution with finite entropy. Then*:

(i) *For any $\varepsilon > 0$, there exists an n such that for any simulation method of $m \geqslant n$ copies of $q_1, \ldots, q_d$ using the process $X_1, X_2, \ldots$, with a stopping time N, we will have $(1/m)E(N) \geqslant H(q_1, \ldots, q_d)/H(X) - \varepsilon$;*

(ii) *For any $\varepsilon > 0$, there exists an n such that it is possible to simulate n copies of $q_1, \ldots, q_d$ using the process $X_1, X_2, \ldots$ with a stopping time N, in such a way that $(1/n)E(N) \leqslant H(q_1, \ldots, q_d)/H(X) + \varepsilon$.*

Partial results in this direction, in particular regarding Markov processes, have been obtained by the author and will be published at a later date. We remark that the Unit Interval method has an immediate generalization

to simulation using any non-i.i.d. process. However, it remains to show that it satisfies a condition similar to that in Theorem 3 when $X_1, X_2, \ldots$ is a stationary ergodic process.

3. In the case of simulation using fair coins, it is possible to find the optimal simulation method. This extends in an obvious way to simulation methods using fair k-sided dice. Can the optimal simulation method be found in the general case of an i.i.d. process?

Acknowledgements

Thanks to Prof. David Gilat for his invaluable remarks and help; thanks also to the referee, who made some helpful suggestions.

References

Abramson, N., 1963. Information Theory and Coding. McGraw-Hill, New York.

Blum, M., 1986. Independent unbiased coin flips from a correlated biased source – a finite state Markov chain. Combinatorica 6(2) 97–108.

Elias, P., 1972. The efficient construction of an unbiased random sequence. Ann. Math. Statist. 43, 865–870.

Knuth, D.E., Yao, A.C., 1976. The complexity of nonuniform random number generation. In: Traub, J.F. (Ed.), Algorithms and Complexity, New Directions and Results. Academic Press, New York, pp. 357–428.

Stout, Q.F., Warren, B., 1984. Tree algorithms for unbiased coin tossing with a biased coin. Ann. Probab. 12, 212–222.