An Entropy Bound for Random Number Generation

Sung-il Pae,

Hongik University, Seoul, Korea

Summary
Many computer applications use random numbers as an important computational resource, and they often require random numbers of particular probability distributions. We consider a very general model of random number generation that converts a source that produces symbols according to a probability distribution into a random numbers of another probability distribution. In such a procedure, we are particularly interested in the average amount of source symbols to produce an output, which we call *efficiency* of the procedure. We discuss the entropy bound for the efficiency of random number generation, and as a main contribution, we give a new elementary proof for the entropy bound.

Key words:
Random number generation, Shannon entropy, information-theoretic bound, coin flip.

1. Introduction

Random numbers are essential in many computer applications including computer simulation, cryptography, randomized algorithms, Monte Carlo methods, etc., and they often require random numbers of particular probability distributions. In this paper, we consider a very general model of random number generation that converts a source that produces symbols according to a probability distribution into a random numbers of another probability distribution. The distribution of the source may be known beforehand, or may be unknown. For both cases, there are known methods using the model to produce a given probability distribution [1, 2].

In such a procedure, we are particularly interested in the average amount of source symbols to produce an output, which we call efficiency of the procedure. The efficiency is subject to an information-theoretic lower bound which is called the entropy bound. The entropy bound is written as the ratio of the Shannon entropies of the source distribution and the target distribution. We discuss the efficiency and its entropy bound in the context of the model of random number generation that we call tree functions, and as a main contribution, we give a new elementary proof for the entropy bound.

Section 2 summarizes previous works related to the random number generation that we discuss in this paper. Then we discuss a general model of random number generation called tree functions and the entropy bound in Section 3. The model is general enough to cover all the methods mentioned in the related works. In Section 4, we prove the main theorem, from which the entropy bound is derived and proved. Section 5 concludes the paper.

2. Related Works

Many previous papers addressed the problem of simulating a discrete probability distribution using another source of randomness. Von Neumann's method is probably the earliest known solution for this problem [3]. His trick converts a biased coin, where the bias may be unknown, to a unbiased coin. Although von Neumann's algorithm is more than fifty years old, it is still used, for example, in a modern computing device, an Intel chipset for random number generation [4].

The fact that von Neumann's algorithm works for unknown source bias is important because the source of randomness is not only usually biased, but also its distribution may be unknown. Diaconis et al. [5, 6] gave a dramatic demonstration of this fact: if we toss a coin with heads up, the probability that we will get heads as a result is more than 0.51, even if the coin is physically perfect.

Elias discussed an infinite sequence of (increasingly complicated) functions that converts a nonuniform discrete probability distribution into a uniform distribution, whose efficiencies approach arbitrarily close to the entropy bound [7]. Peres also devised procedures whose efficiencies approach the entropy bound [8]. Interestingly, Peres's procedures are defined recursively, and thus they are easily implementable. Dijkstra presented an elegant algorithm that simulates an m-faced (uniform) roulette from m flips of biased coins, where m is a prime number [9].

The above works are mostly concerned with generating uniform random numbers from a biased source. In a somewhat opposite direction, Knuth and Yao addressed the problem of simulating a given discrete probability distribution using an unbiased coin [10]. Their method results in optimal solutions in terms of efficiency in the usage of the source of randomness. Han and Hoshi studied a more general problem where the source bias is known but not necessarily 0.5 and proposed a method based on interval subdivision [11]. Their work addresses the problem of converting a probability distribution to another

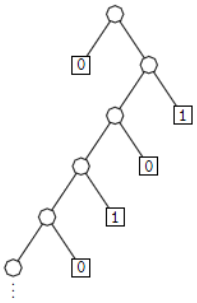
Manuscript received September 5, 2009
Manuscript revised September 20, 2009

distribution in a full generality to which the entropy bound applies. However, their method does not result in optimal solutions in general. Recently, Pae and Loui studied the previous methods for random number generation in a framework called randomizing functions and discussed their efficiency and computational complexity [1, 2].

3. Random Number Generation by Tree Functions

Suppose that a source produces symbols from $\Sigma = \{x_1, \ldots, x_n\}$ according to a probability distribution $\mathbf{p} = \langle p_1,\ldots,p_n\rangle$, and each produced symbol is independent of each other. Consider a function $f : T \to \{y_1,\ldots, y_m\}$, where T is an exhaustive prefix-free subset of Σ^*. Then f can be represented by an n-ary tree such that a string σ in T corresponds to a path from the root to a leaf, and the value associated with the leaf is $f(\sigma)$. Such a function can be used to convert a probability distribution to another distribution: the probability that f outputs y_j is $\Pr[\sigma \in f^{-1}(y_j)]$.

For example, let $\Sigma =$, and suppose that a source produces h with probability 1/3 and t with probability 2/3. Let $T =$ B_0, where $=\{h\}$ $\{$ $th^{2k+1}t \mid k = 0,1\}$ and $=\{$ $th^{2k}t \mid k = 0,1\}$, and outputs 0 and 1 on and , respectively. The function can be represented as the following infinite tree:



(1)

It is straightforward to verify that the probability that f outputs 0 is 1/2. So the output of f can be regarded as a fair coin flip, and f converts the probability distribution <1/3,2/3> to <1/2,1/2>. The average depth of the tree (1) is 15/7, which means that the procedure represented by the tree takes 15/7 coin flips on average to produce an output symbol.

Let us call such a function a *tree function*. If the set T is computable, then we can compute the corresponding function f. In this case, we may call such an algorithm tree algorithm. For example, in (1), f is computable using a finite automaton. In general, a prefix-free set may not be computable. Call the probability distribution of the output of a tree function the *target distribution*. When the probability distribution of the source and the target of f are $\mathbf{p} = \langle p_1,\ldots,p_n\rangle$ and $\mathbf{r} = \langle r_1,\ldots,r_m\rangle$, respectively, let us say f is a tree function for $(\mathbf{p}, \mathbf{r})$. We will call the average number of source symbols per target symbol of a tree function the *efficiency*.

Now we are ready to state the entropy bound:

Theorem 1. *For the source distribution* $\mathbf{p}=\langle p_1,\ldots,p_n\rangle$ *and the target distribution* $\mathbf{r}=\langle r_1,\ldots,r_m\rangle$, *the efficiency of a tree function for* $(\mathbf{p}, \mathbf{r})$ *is at least* $H(\mathbf{r})/H(\mathbf{p})$, *where H is the Shannon entropy function.*

The Shannon entropy of a two-valued distribution $\langle p, p\rangle$ is defined to be

$$H(p) = -p\log p - (1-p)\log p,$$

and more generally, for a probability distribution, $\mathbf{p} = \langle p_1,\ldots,p_n\rangle$, it is defined as follows [12]:

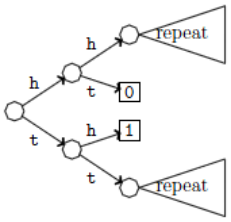
$$H(\mathbf{p}) = -\sum_{k=1}^{n} p_k \log p_k$$

In this paper, we assume that $\log x = \log_2 x$.

This theorem seems intuitively correct, because the Shannon's entropy is meant to be the minimum number of bits to represent the source of the given distribution.

As shown above, the efficiency of the tree function represented by (1) is 15/7, which is approximately 2.14. The entropy bound for the case, where source distribution is <1/3,2/3> and target distribution is $H(1/2)/H($ is approximately 1.09.

As another example of tree functions, consider the function $f_{VN} : \{h, t\}^2 \to \{0,1\}$ defined by f_{VN}(ht) = 0, f_{VN}(ht) = 1, f_{VN}(hh) = λ, and f_{VN}(tt) = λ, where λ is an empty string. By extending the function appropriately to $\{h, t\}^*$, we obtain a tree function, which is the famous von Neumann's method [3]: to obtain a fair coin using a biased coin, flip the coin twice; if the result is heads-tails, then regard it as a heads, if tails-heads, regard it as a tails, otherwise repeat the procedure. The infinite tree corresponding to von Neumann's method can be represented as follows:



Contrary to the procedure represented by the tree (1), which produces a different

distribution for source distribution other than <1/3,2/3>, von Neumann's method always produces the uniform distribution for any source distribution. Suppose that the bias of the source coin is p. Then, the efficiency of von Neumann's method, which is not hard to compute, is $1/2p(1-p)$, while the entropy bound is $H(1/2)/H(p) = 1/H(p)$. Figure 1 compares the efficiency of von Neumann's method and the entropy bound as the source bias p varies between 0 and 0.5. There are known methods, which are more complicated than von Neumann's, whose efficiencies approach to the entropy bound. (See, for example, [2].)

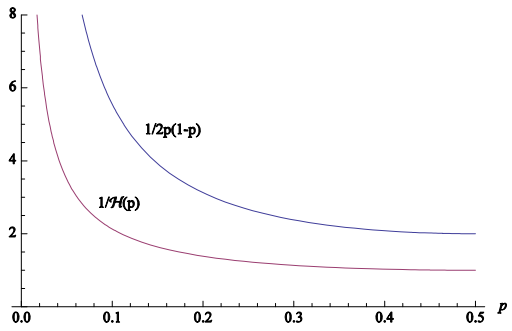


Fig. 1 The efficiency of von Neumann's method and the entropy bound.

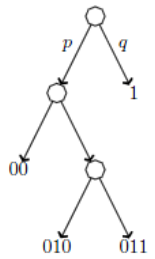
Several versions of this theorem appear in literature: Elias [7] and Peres [8] for the generation of a fair coin from a biased coin, Knuth and Yao [10], and Cover and Thomas [13, Section 5.12] for the generation of a general discrete distribution from a fair coin, and Roche [14] and Han and Hoshi [11] for the generation of a general discrete distribution from another general discrete distribution like our case.

We can see Theorem 1 as a corollary of Theorem 2 below. The main purpose of this paper is to give an elementary proof of Theorem 2. Cover and Thomas [13, Theorem 5.12.1] proves a special case of Theorem 2, in which the source distribution is two-valued and uniform. However, their proof does not generalize. Han and Hoshi [11] mentions Theorem 2 without a proof. Our proof is interesting because it is purely algebraic.

4. Entropy of an Induced Random Variable

Let X be a random variable that takes values over $\Sigma = \{x_1, \ldots, x_n\}$ such that $\Pr[X = x_i] = p_i$ for each $i = 1,\ldots,n$. Consider a random variable Y over an exhaustive prefix-free subset T of Σ^* such that for $\sigma = x_{i1}\ldots x_{ik}$ in T, the probability $\Pr[Y = \sigma] = p_{i1}\ldots p_{ik}$. Since T is exhaustive, $\sum_{\sigma\in T}\Pr[Y = \sigma] = 1$. We say that Y is induced from X via T.

An induced random variable can be represented as a complete n-ary tree. Conversely, a complete n-ary tree defines an induced random variable. As an example, let $\Sigma = \{0, 1\}$ with probabilities $<p, q>$. The following tree (2) represents an induced random variable over {1, 00, 010, 011}, whose probability distribution is $<q, p^2, p^2q, pq^2>$.



(2)

Now let $D = |Y|$. Then D is the random variable representing the length of a word in T. So $E(D)$, the expected value of D, is the average length of the words in T, or equivalently, the average depth of the tree corresponding to the induced random variable Y. We will show that the entropy of the induced random variable Y equals the entropy of the random variable X multiplied by $E(D)$.

Theorem 2. *Let Y be a random variable induced from X and let $D = |Y|$. Then $H(Y) = E(D)H(X)$.*

Proof. We first present a proof in the case $n = 2$. Because the probabilities for the words in T sum to 1, with a slight abuse of notation we can write

$$1 = \sum_T p^k q^l,$$

where k is the number of left edges taken in the path from the root to a terminal node of the tree (or the number of 0's in the corresponding word in T), and l is the number of right edges, hence the probability $p^k q^l$ for a terminal node. The average length of the words in T, or equivalently the average depth of T is

$$E(D) = \sum_T (k + l)p^k q^l,$$

which is the sum of probabilities of terminal nodes multiplied by their depths. Consequently,

$$-E(D)H(X) = \sum(k + l)p^k q^l p \log p + \sum(k + l)p^k q^l q \log q.$$

Now the entropy of Y is

$$\begin{aligned} H(Y) &= -\sum p^k q^l \log p^k q^l \\ &= -\sum kp^k q^l \log p - \sum lp^k q^l \log q \end{aligned}$$

It suffices to prove the following equalities to prove Theorem 2 for $n = 2$:

$$p\sum_T (k+l)p^k q^l = \sum_T kp^k q^l, \tag{3}$$

$$q\sum_T (k+1)p^k q^l = \sum_T lp^k q^l. \tag{4}$$

In the following, we will prove these two equalities. Consider the function

$$F(x) = \sum_T x^k (1-x)^l. \tag{5}$$

The function $F(x)$ is a polynomial in the case T is finite, and an infinite series of functions when T is infinite. By definition, F is identically 1 on [0, 1]. (In fact, F does not need to be restricted on the interval [0, 1].) Therefore, the first derivative of F is identically zero:

$$F'(x) = \sum_T [kx^{k-1}(1-x)^l - lx^k(1-x)^{l-1}] \equiv 0. \tag{6}$$

In case that T is infinite, hence the sum (5) is an infinite series of functions, the derivation (6) is justified because the series converges uniformly to 1. (See, for example, [15].)

Hence for every p in [0, 1], we obtain an identity

$$\sum_T kp^{k-1}q^l = \sum_T lp^k q^{l-1}. \tag{7}$$

The identity (7) is used in the following manipulation of equations, which proves the identity (3).

$$\begin{aligned} p\sum_T (k+l)p^k q^l &= pq\sum_T kp^k q^{l-1} + pq\sum_T lp^k q^{l-1} \\ &= pq\sum_T kp^k q^{l-1} + pq\sum_T kp^{k-1}q^l \\ &= \sum_T kp^k q^l(p+q) \\ &= \sum_T kp^k q^l \end{aligned}$$

The identity (4) is proved similarly.

The above proof generalizes to an n-valued distribution $\langle p_1,\ldots,p_n\rangle$. Consider the function of the form

$$F(x_1,\ldots,x_n) = \sum_T x_1^{k_1}\cdots x_n^{k_n}, \tag{8}$$

where T is a complete n-ary tree and the summation is over the words in T, and k_i is the number of the ith edges taken in the path from the root to the corresponding terminal node. The function $F(x_1,\ldots,x_n)$ is identically 1 on the hyperplane deifined by the equation $x_1+\cdots+x_n = 1$.

Then by taking partial derivatives at a point $(p_1,\ldots,p_n)$ on the same hyperplane, we have

$$\begin{aligned} \frac{\partial F}{\partial x_i}(p_1,\ldots,p_n) &= \sum k_i p_1^{k_1}\cdots p_i^{k_i-1}\cdots p_n^{k_n} \\ &\quad -\sum k_n p_1^{k_1}\cdots p_n^{k_n-1} \\ &= 0, \end{aligned}$$

for $i = 1,\ldots,n$. As a result, we obtain the following identity:

$$\sum k_1 p_1^{k_1-1}\cdots p_n^{k_n} = \cdots = \sum k_i p_1^{k_1}\cdots p_i^{k_i-1}\cdots p_n^{k_n} = \cdots = \sum k_n p_1^{k_1}\cdots p_n^{k_n-1}$$

With this identity, the proof follows for an n-valued distribution as in the two-valued case. □

Note that the summands of the functions (5) and (8) are very simple polynomials of the forms $x^k(1-x)^l$ and $x_1^{k_1}\cdots x_n^{k_n}$. Especially, in the two-valued case, if the corresponding tree T is a full tree of depth d, then the function is written as

$$f(x) = \sum_{i=0}^{d}\binom{d}{i}x^i(1-x)^{d-i}.$$

The summands $\binom{d}{i}x^i(1-x)^{d-i}$ are known as Bernstein polynomials, and they are also called Bernstein basis because they are linearly independent. They have nice numerical properties that are bases of the usefulness of Bézier curves. Given an exhaustive prefix-free set T, call the polynomials $x^k(1-x)^l$ (or $x_1^{k_1}\cdots x_n^{k_n}$ in n-dimensional case) corresponding to the words in T leaf polynomials. Pae and Loui discussed a criterion for the set of leaf polynomials to be linearly independent, and in that case, there is a method that assigns the output values to the leaves of T such that the corresponding random number generation, which generates the target distribution regardless of the source distribution, is the most efficient among all possible random number generations for the target and source [2].

Theorem 1 follows from Theorem 2. Let X be a random variable with alphabet $\Sigma = \{x_1, \ldots, x_n\}$ with distribution $\mathbf{p}$. If $f : T \to \{y_1,\ldots,y_m\}$ is a tree function for $(\mathbf{p}, \mathbf{r})$, then $f(Y)$ is a random variable over $\{y_1,\ldots,y_m\}$ with probability distribution $\mathbf{r}$, where Y is induced from X via T. By a well-known property of the Shannon entropy we have

$$H(f(Y)) \le H(Y).$$

Hence we have

$$H(f(Y)) \le E(D)H(X),$$

where $D = |Y|$. Note that the efficiency of a tree function is the average depth of the corresponding tree. Therefore, we have proved Theorem 1.

4. Conclusion

We discussed the problem of generating a random numbers of a particular probability distribution from another distribution. The problem can be studied using the tree functions that model the conversion processes naturally. The efficiency of the procedures, the average number of source symbols to generate an output symbol, is subject to an information-theoretical lower bound called the entropy bound, which is described in terms of the Shannon entropies of the source distribution and the target distribution.

We gave a new proof for the entropy bound for random number generation. The tree function model applies to most known methods for converting a probability distribution to another distribution, and the proof is based on elementary properties of polynomial functions and infinite series of functions that arise in the tree structure of the random number generation process.

Acknowledgment

This work was supported in part by a Hongik University research grant and National Research Foundation of Korea Grant funded by the Korean Goverment (2009-0077288).

References

[1] Sung-il Pae and Michael C. Loui. Optimal random number generation from a biased coin. In *Proceedings of the Sixteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1079-1088, January 2005.

[2] Sung-il Pae and Michael C. Loui. Randomizing functions: Simulation of discrete probability distribution using a source of unknown distribution. *IEEE Transactions on Information Theory*, 52(11):4965-4976, November 2006.

[3] John von Neumann. Various techniques for use in connection with random digits. Notes by G.E. Forsythe. In *Monte Carlo Method, Applied Mathematics Series*, volume 12, pages 36-38. U.S. National Bureau of Standards, Washington D.C., 1951. Reprinted in von Neumann's *Collected Works* 5 (Pergammon Press, 1963), 768-770.

[4] Benjamin Jun and Paul Kocher. The Intel random number generator. White paper prepared for Intel Corporation, 1999. Cryptography Research, Inc.

[5] P. Diaconis, S. Holmes, and R. Montgomery. Dynamical Bias in the Coin Toss. *SIAM Review*, 49(2):211, 2007.

[6] Persi Diaconis. The search for randomness. At American Association for the Advancement of Science annual meeting. Feb. 14, 2004. Seattle.

[7] Peter Elias. The efficient construction of an unbiased random sequence. *The Annals of Mathematical Statistics*, 43(3):865-870, 1972.

[8] Yuval Peres. Iterating von Neumann's procedure for extracting random bits. *Annals of Statistics*, 20(1):590-597, 1992.

[9] Edsger W. Dijkstra. Making a fair roulette from a possibly biased coin. *Information Processing Letters*, 36:193, 1990.

[10] Donald E. Knuth and Andrew C-C. Yao. The complexity of nonuniform random number generation. In Joseph F. Traub, editor, *Algorithms and Complexity: New Directions and Recent Results. Proceedings of a Symposium*, pages 357-428, New York, NY, 1976. Carnegie Mellon University, Computer Science Department, Academic Press. Reprinted in Knuth's *Selected Papers on Analysis of Algorithms* (CSLI, 2000).

[11] Te Sun Han and Mamoru Hoshi. Interval algorithm for random number generation. *IEEE Transactions on Information Theory*, 43(2):599-611, 1997.

[12] Claude Elwood Shannon and Warren Weaver. *The Mathematical Theory of Communication*. The University of Illinois Press, Urbana, 1964.

[13] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley Series in Telecommunications. John Wiley & Sons, New York, NY, USA, 1991.

[14] J. R. Roche. Efficient generation of random variables from biased coins. Technical report, AT&T Lab., 1992. Bell Tech. Report File case 20878.

[15] Tom Apostol. *Mathematical Analysis*. Addison-Wesley, 1974.

Sung-il Pae received the B.S. degree in Mathematics from Seoul National University in 1993, M.S. degree in Mathematics from University of Illinois at Urban-Champaign in 1997, and Ph.D. degree in Computer Science from University of Illinois at Urban-Champaign in 2005. During 2005-2007, he stayed at Korea Institute for Advanced Study before he joined Computer Engineering department of Hongik University, Seoul, Korea. His research interest includes Algorithms and Theoretical Computer Science.