UNBIASED RANDOM SEQUENCES FROM MEASUREMENTS OF POISSON PROCESSES

Riccardo Bernardini, Roberto Rinaldo

University of Udine
Department of Electrical, Management and Mechanical Engineering
Via delle Scienze 206, Udine, Italy
(bernardini, rinaldo)@uniud.it

ABSTRACT

We consider the problem of true random bit generation from source vectors of independent geometric random variables, reduced modulo M for practical implementation. Independent geometric random variables result from measurements of discretized Poisson processes, which are good models for a number of physical sources. We propose a generalization of the classical approach by Elias, compute theoretical bounds, and evaluate the efficiency of the scheme by means of experiments. The proposed technique shows a significant advantage with respect to the classical approach.

Index Terms— Security, Cryptography, Random numbers.

1. INTRODUCTION

There are countless applications relating to the need to generate random data, such as simulation algorithms based on the Monte Carlo method, network coding applications, compressive sensing, encryption. In most of these applications, one expects to receive in input "truly random bits." In particular, the use of pseudo-random numbers generated by numerical algorithms, may not provide the level of security required by cryptographic applications [1]. Even in applications where pseudo-random numbers are acceptable, security ultimately depends on the choice of a "seed" that should be truly random. These considerations have motivated research on the generation or extraction of truly random bits from physical sources [1, 2, 3, 4]. Every True Random Number Generator (TRNG) uses internally a physical process from which the randomness used to generate the random bits is harvested. Several natural processes (e.g., radioactive decay, photons landing on a photodiode, shot noise in electronic circuits) are without memory and follow a Poisson law. Therefore, it is of practical interest to devise a simple procedure to extract randomness from these sources in an efficient way, where the word "efficient" means that it must be possible to have the rate of bit production as close as desired to the information content of the physical process. This paper is motivated by the problem of generating truly random bits from sources that can be described by a Poisson process [5].

2. RELATION TO PRIOR WORK

The problem of true random number generation dates back to von Neumann [6] who considered the problem of simulating an unbiased coin by using a biased coin with unknown probability. Denoting with T and H the tail and head outcomes, respectively, he observed that considering two consecutive independent coin tosses, the events TH and HT are exactly equiprobable. Thus, mapping $TH \to 0$, $HT \to 1$, while discarding the events TT, HH, generates a sequence of truly random bits even if the original coin is biased. More efficient algorithms for generating random bits from a biased coin were proposed by various authors [7, 8, 9, 10]. See [11] for a more comprehensive bibliography, where the problem to generate random bits from a correlated source is considered. Elias [8] was the first to devise an optimal procedure in terms of information efficiency, namely, the expected number of unbiased random bits generated per coin toss is asymptotically equal to the entropy of the biased coin. Starting from a source that produces bit vectors $\mathbf{n} = [N_1, ..., N_L]$, of binary independent random variables (rv) $N_i \in \{0, 1\}$, $\mathrm{P}[N_i = 0] = q$, the procedure partitions the range of $\mathbf{n}$ into Q classes $C_1, ..., C_Q$, where class C_i consists of all the permutations (with repetition) of the bit string with a given Hamming weight. Due to independence, the elements of each class C_i are therefore equiprobable. The procedure to generate a bit string corresponding to an instance vector $[n_1, ..., n_L]$, requires to identify the class C_i to which $\mathbf{n}$ belongs, evaluate its cardinality j and a number $b \in \{0, ..., j-1\}$ which unambiguously identifies $\mathbf{n}$ within the class. Then the *Elias mapping* can be used. In particular, define as $k_i, i = 0, ..., N_j$, the positions of the $N_j + 1$ bits equal to one in the binary representation of j, so that

$$j = \sum_{i=0}^{N_j} 2^{k_i}.$$

If j is an even number, one can construct a one-to-one correspondence between b and an appropriate bit string, in a way that the generated bit sequence consists of independent and

978-1-4799-2893-4/14/$31.00 ©2014 IEEE

equiprobable symbols. As a matter of fact, one can associate the values $b = 0, ..., 2^{k_0} - 1$, to the 2^{k_0} bit strings of length k_0, the values $b = 2^{k_0}, ..., 2^{k_0} + 2^{k_1} - 1$ to the 2^{k_1} bit strings of length k_1, and so on. In this way, since b are independent random variables uniformly distributed in $\{0, ..., j-1\}$, the corresponding bit strings will have independent and equiprobable symbols. If j is an odd number, one of the values of b, say $b = j - 1$, can be associated to the output of a null-string. Once j and b are known, the Elias mapping procedure is easy to implement.

In this paper, we extend the original approach of Elias by considering partitions of the range of $\mathbf{n}$ into generic classes of equiprobable vectors, not just permutations as in the original procedure. We derive lower and upper bounds for the efficiency of this generalized approach. We then consider the case of vectors generated from measurements of a Poisson process, and present a procedure that has significant advantages with respect to the original Elias' scheme.

3. MAIN RESULT

We will consider sources that generate vectors $\mathbf{n} = [N_1, ..., N_L]$, where $N_i \in A$ are independent rv, and A is a generic finite alphabet. A *conditioner* is a map $\mathcal{E}(\mathbf{n}) : A^L \to \{0,1\}^*$, mapping vector $\mathbf{n}$ to the (possibly empty) bit string s_n, with length $\ell(s_n) = \ell_n$. A conditioner is *admissible* if

$$\forall s \in \operatorname{Im} \mathcal{E}, \mathrm{P}[\mathcal{E}(\mathbf{n}) = s | \ell_n = |s|] = 2^{-|s|},$$

where $|s|$ denotes the length of the bit string. It is possible to show that this property guaranties that the bit sequences obtained by concatenating the strings $\mathcal{E}(\mathbf{n}_i)$ corresponding to successive source vectors $\mathbf{n}_0, \mathbf{n}_1, ..., \mathbf{n}_K$, are truly random and robust with respect to the attacks of an opponent who could observe the number K and the length of the corresponding bit string.

Let us define $\bar{L} = \mathrm{E}[\ell_n]$ and denote with $R = \bar{L}/L$ the *rate* of the conditioner. Let $C_1, C_2, ..., C_Q$, be a partition of A^L so that all the outcomes belonging to class C_i are equiprobable, and let $\mathcal{E}(\mathbf{n})$ be constructed according to the Elias mapping described above. It is possible to show that

$$H(N) - \frac{H(\ell_n)}{L} - \frac{\log_2(Q)}{L} \leq R \leq H(N) - \frac{H(\ell_n)}{L},$$

where $H(X)$ denotes the entropy of rv X [12]. Moreover, for $L \to +\infty$, we have $H(\ell_n)/L \to 0$. For the original Elias' scheme, where classes C_i are obtained with permutations, and for the classes we propose below for geometric rv, we also have $\log_2(Q)/L \to 0$, so the techniques are asymptotically optimal. The proofs of these properties are omitted here for space reasons.

4. APPLICATION TO POISSON PROCESSES

We take measurements of a physical process described by a homogeneous Poisson process with intensity λ [5, 13]. We discretize the time axis into intervals of equal size Δ, and check if one or more arrivals occur in each interval. The observations are therefore a discrete time Bernoulli process, where the probability of 0 is equal to $p = e^{-\lambda\Delta}$. The inter-arrival times N_i of the discretized process follow a geometric distribution with probability mass function

$$p(k) = (1-p)\, p^k, k = 0, 1, \ldots . \tag{1}$$

Our objective is to devise a procedure to generate a sequence of truly random bits from vectors $\mathbf{n} = [N_1, ..., N_L]$, where N_i are independent geometric random variables.

The original scheme proposed by Elias for the construction of classes of equiprobable vectors is based on the fact that, since N_i are independent random variables, the permutations (with repetition) of the values of the vector all have the same probability. For example, the results

$$[6442], [6424], [6244], [4642], [4624], [4462],$$

$$[4426], [4264], [4246], [2644], [2464], [2644]$$

are obviously equiprobable. Note that the cardinality of the class of length-L vectors with j different symbols, each appearing k_i times, $k_1 + ... + k_j = L$, is given by the multinomial coefficient

$$\left(\begin{array}{c} L \\ k_1, k_2, \cdots k_j \end{array} \right) = \frac{L!}{k_1! k_2! \cdots k_j!}.$$

For example, for $L = 4$, $k_1 = 1, k_2 = 1, k_3 = 2$, there are 12 permutations as in the example above. We will call the classes constructed using this procedure the *Elias classes*.

The key idea proposed in these notes, for the special case of geometric random variables, is based on the following reasoning. A sequence $[n_1, n_2, ..., n_L]$ of natural numbers such that $n_1 + n_2 + \cdots + n_L = K$ is called a weak composition of K with L parts. In case N_i have a geometric distribution, due to the fact that $p(\mathbf{n})$ is the product of L probability mass functions of type (1), all the weak compositions with sum K will have the same probability, and they can be enumerated to generate bit strings as in the Elias mapping. It is well known that the number of weak compositions of K with L parts is given by

$$\left(\begin{array}{c} L + K - 1 \\ K \end{array} \right).$$

For example, with $L = 3$ and $K = 3$, we have the following 10 possibilities

$$[003], [030], [300], [012], [021],$$

$$[102], [120], [210], [201], [111].$$

Note that, in the original scheme of Elias, this set would be split into three classes, with 3, 6, and 1 element, respectively. We expect therefore that the proposed method can attain a higher efficiency since a more numerous class can be associated with the generation of longer bit strings.

As a matter of fact, in a practical implementation, it is unfeasible to deal directly with instances n_i of geometric random variables, since they can assume, although with vanishing probability, unlimited values not representable in a finite precision system. Alternatively, one can easily deal with N_i modulo some predefined value M. It is easy to see that the resulting random variable has a probability mass function

$$p_n(k) = \frac{(1-p)}{1-p^M} p^k, k = 0, ..., M-1,$$

$$p_n(k) = 0, \text{elsewhere}, \quad (2)$$

so that vectors $[n_1, n_2, ..., n_L]$ with the same sum $n_1 + n_2 + \cdots + n_L = K$ will still be equiprobable. The cardinality of the set of vectors with sum equal to K, constrained by the fact that $0 \leq n_i < M$, can be computed according to the following reasoning.

Consider the polynomial

$$\begin{aligned} p(x) &= (1 + x + x^2 + ... + x^{M-1})^2 \\ &= p_0 + p_1 x + p_2 x^2 + ... + p_{2M-2} x^{2M-2} \end{aligned}$$

The coefficient p_k of x^k in $p(x)$, $0 \leq k < 2M-1$, will count all the possible products $x^r x^q$, $0 \leq r, q < M$, such that $r + q = k$. Therefore, the coefficient p_k represents exactly the cardinality of the set of pairs of naturals $[n_1, n_2]$ with sum equal to k, and constrained by the fact that $0 \leq n_i < M$. In general, the coefficient p_k of

$$p(x) = (1 + x + x^2 + ... + x^{M-1})^L$$

will count the cardinality of the set of length-L vectors $[n_1, n_2, ..., n_L]$ with sum equal to k and $0 \leq n_i < M$. For instance, when $L = 3$, $M = 3$ and $k = 3$, we have the following 7 possibilities

$$[012], [021], [102], [120], [210], [201], [111].$$

5. VECTOR ENUMERATION

Let us now turn to the the problem of the enumeration of the elements of a class of equiprobable vectors. This problem can be solved via a look-up table, but the approach is readily unfeasible even for small L.

A general enumeration algorithm which allows to uniquely assign a certain number $0 \leq b < j$ to a particular vector $[n_1, n_2, ..., n_L]$ belonging to a class of cardinality j, can be obtained by partitioning the vectors in the class recursively, starting with the value of the first vector component n_1. We will exemplify the procedure considering the case of length-L vectors with the same sum K and constrained by $0 \leq n_i < M$, but the same reasoning can be applied for the enumeration of the Elias classes or of weak compositions. Let us denote with $N_M(l, k)$ a function that returns the cardinality of the class of constrained l-length vectors with sum k. Function $N_M(l, k)$ can be easily computed according to the results presented above. If $n_1 = 0$, then $N_M(L-1, K)$ vectors are possible, corresponding to all the admissible values of $[n_2, ..., n_L]$. Therefore, if $n_1 = 0$, we restrict b to $0, ..., N_M(L-1, K) - 1$. If $n_1 = 1$, then $N(L-1, K-1)$ vectors are possible, to which we reserve indices $N_M(L-1, K) \leq b < N_M(L-1, K) + N_M(L-1, K-1)$. We proceed by partitioning the set of indices for all possible values of n_1 so that a particular value of n_1 identifies one set of the partition. Then, for each value of n_2, we further partition the subset identified by n_1. In particular, the first $N_M(L-2, K-n_1)$ indices of the subset are reserved to the vectors with $n_2 = 0$, and so on, as before. We proceed with all the vector components till the last one, which originates partitions with one single element. The ®Matlab function of Fig. 1 is a code for the procedure. In the code, `a` is a matrix where `a(l,k+1)` contains the value $N_M(l, k)$. In the procedure, `b` keeps track of the smallest index in the current partition, until the partition contains one single element.

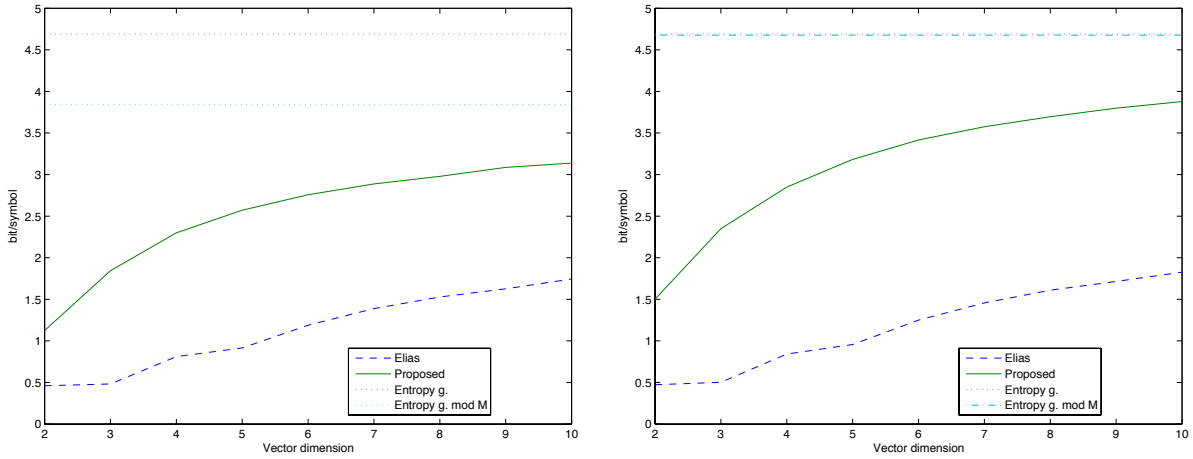
```
-------------------------------------
function y=enumerateM(n,a)

k=sum(n);
b=0;
i=1;
l=length(n);
while ((k>0) && (l>=2)),
    if n(i)>=1,
        for j=1:n(i)-1,
            b=b+a(l-1,k-j+1);
        end;
        b=b+a(l-1,k+1);
    end;
    k=k-n(i);
    l=l-1;
    i=i+1;
end;

y=b;
-------------------------------------
```

Fig. 1. ®Matlab function for the enumeration of constrained vectors with the same sum.

Note that a matrix containing all the values $N_M(l, k)$, $l = 1, ..., L,$, $k = 0, ..., L(M-1)$ has $L(1 + (M-1)(L+1)/2)$



(a) (b)

Fig. 2. Comparison between the Elias mapping and the proposed one. (a) $M = 16$, (b) $M = 64$.

non zero-elements. For instance, when $L = 4$ and $M = 64$, there are 634 non-zero elements.

6. EXPERIMENTS

In this section, we compare the performance of the proposed scheme with the one obtained using the original Elias classes. We assume that $\mathbf{n} = [N_1, ..., N_L]$ is a vector of independent geometric random variables, represented modulo M, each with probability mass function given by (2). It is well known that the entropy of the geometric random variable with probability mass function (1), is

$$H_g(p) = -\log_2(1-p) - \frac{p}{1-p}\log_2 p.$$

The entropy of the geometric random variable reduced modulo M is

$$\begin{aligned} H_{g,M}(p) &= -\log_2 \frac{1-p}{1-p^M} \\ &- \frac{p(Mp^M - Mp^{M-1} + 1 - p^M)}{(1-p^M)(1-p)} \log_2 p. \end{aligned}$$

As explained above, the Elias mapping $s_n = \mathcal{E}(\mathbf{n})$ generates a bit string s_n of length $\ell(s_n)$. For a given input vector $\mathbf{n}$, s_n will depend on the method used to form classes (e.g., the Elias classes or the proposed ones).

It is therefore easy to compute the efficiency of the two schemes on the basis of the average output bit string length

$$\bar{L} = \sum_{\mathbf{n}} \ell(s_n) p(\mathbf{n}), \quad \mathbf{n} = [n_1, ..., n_L],$$

where $p(\mathbf{n})$ is the product of L probability mass functions of type (2). Table 1 shows $R = \bar{L}/L$ for the two methods when the variables are represented modulo $M = 16$. We set $p = 0.9$. Note that in this case the entropy of the source is $H_g(p) = 4.6900$ and $H_{g,M}(p) = 3.8411$.

For larger values of L and M we simulated $T = 15000$ realizations of $\mathbf{n}$, concatenate the output binary strings into one string s_T and plot in Fig. 2 the values $\ell(s_T)/(LT)$ for $M = 16$ and $M = 64$, $L = 2, ..., 10$. Note that for $M = 64$, we have $H_{g,M}(p) = 4.6768$, due to the fact that the modulo operation has less influence for larger M. Simulations were performed in ®Matlab using the default uniform random number generator *Mersenne Twister*. Although both methods approach the entropy of the source as L increases, the table and the figures clearly show the advantage of the proposed method.

Table 1. Average length, in bit/symbol for the Elias classes and the proposed ones.

L	2	3	4	5
Elias classes	0.4617	0.4820	0.8104	0.9164
Proposed	1.1272	1.8488	2.2985	2.5738

7. CONCLUSIONS

In this paper we considered the problem of true random bit generation from source vectors of reduced independent geometric random variables, originating from measurements of a discretized Poisson process. We proposed a generalization of the classical approach by Elias, and derived theoretical results about the efficiency of the proposed approach. The advantages of the proposed solution are confirmed by experiments. The procedure can be practically used, for example, in a scheme where radioactive decay is the physical source [14, 15]. The bits generated by the proposed simple procedure can be used to generate truly random bits for key generation in cryptography applications.

8. REFERENCES

[1] G. Taylor and G. Cox, "Behind Intel's new random-number generator," *IEEE Spectrum*, pp. 1020–1022, 2011.

[2] Y. Akizawa, T. Yamazaki, A. Uchida, T. Harayama, S. Sunada, K. Arai, and K. Yoshimura, "Fast random number generation with bandwidth-enhanced chaotic semiconductor lasers at 8×50 gb/s," *IEEE Photonics Technology Letters*, vol. 24, no. 12, pp. 1042–1044, 2012.

[3] X. Li, A. B. Cohen, T. E. Murphy, and R. Roy, "Scalable parallel physical random number generator based on a superluminescent led," *Optics Letters*, vol. 36, no. 6, pp. 1020–1022, 2011.

[4] J. Walker, "Hotbits: Genuine random numbers, generated by radioactive decay," http://www.fourmilab.ch/hotbits/.

[5] F. Cannizzaro, G. Greco, S. Rizzo, and E. Sinagra, "Results of the measurements carried out in order to verify the validity of the poisson-exponential distribution in radioactive decay events," *The International Journal of Applied Radiation and Isotopes*, vol. 9, no. 11, p. 649652, 11 1978.

[6] J. von Neumann, "Various techniques used in connection with random digits," *Appl. Math. Ser., Notes by G. E. Forstyle, Nat. Bur. Stand.*, vol. 12, pp. 36–38, 1951.

[7] W. Hoeffding and G. Simon, "Unbiased coin tossing with a biased coin," *Ann. Math. Statist.*, vol. 41, pp. 341–352, 1970.

[8] P. Elias, "The efficient construction of an unbiased random sequence," *Ann. Math. Statist.*, vol. 43, pp. 865–870, 1972.

[9] Q. Stout and B. Warren, "Unbiased coin tossing with a biased coin," *Ann. Probab.*, vol. 12, pp. 212–222, 1984.

[10] Y. Peres, "Unbiased coin tossing with a biased coin," *Ann. Statist.*, vol. 20, pp. 590–597, 1992.

[11] H. Zhou and J. Bruck, "Efficient generation of random bits from finite state markov chains," *Information Theory, IEEE Transactions on*, vol. 58, no. 4, pp. 2490–2506, 2012.

[12] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York: Wiley, 2006.

[13] S. M. Ross, *Stochastic Processes*. New York: John Wiley, 1995, iSBN 978-0-471-12062-9.

[14] A. Figotin, et al., "Random number generator based on the spontaneous alpha-decay," US Patent US6 745 217 B2, 6 1, 2004.

[15] N. Tsuyuzaki, "Random pulse generation source, and semiconductor device, method and program for generating random number and/or probability using the source," US Patent US8 001 168 B2, 8 16, 2011.