Exact Output Rate of Generalized Peres Algorithm for Generating Random Bits from Loaded Dice

Sung-il Pae
Department of Computer Engineering
Hongik University
Seoul, Korea
Email: pae@hongik.ac.kr

***Abstract*—We report a computation of the exact output rate of recently-discovered generalization of Peres algorithm for generating random bits from loaded dice. Instead of resorting to brute-force computation for all possible inputs, which becomes quickly impractical as the input size increases, we compute the total output length on equiprobable sets of inputs by dynamic programming using a recursive formula.**

***Keywords*—Random number generation, Peres algorithm, exact output rate, random bits, loaded dice.**

I. Introduction

Peres algorithm recursively produces unbiased coin flips from biased coin flips, with von Neumann's method as its base [1]. Because it is defined by a simple recursion, Peres algorithm is easy to implement and yet runs fast.

The output *rate* of a procedure that converts a biased Bernoulli source with n-valued distribution $\mathbf{p} = (p_1, \ldots, p_n)$ to unbiased random bits is the average number of output bits per input, and it is known to bounded by Shannon entropy $H(\mathbf{p}) = -(\log_2 p_1 + \cdots + \log_2 p_n)$ [2], [3], [4]. Since Peres algorithm is such a procedure, consequently, its rate is bounded by the entropy bound, $h(p) = -(p\log_2 p + (1-p)\log_2(1-p))$. Interestingly, the rates of Peres algorithm approaches to the entropy bound as the input length tends to infinity [1], and we call such algorithms *asymptotically optimal*.

The exact output rate of Peres algorithm was reported [5] and compared with another asymptotically optimal method by Elias [2]. Recently, a generalization of Peres algorithm was found for generating unbiased random bits from loaded dice, that is, many-valued Bernoulli source [6]. We report, here, a computation of the exact output rate of the three-face case, thus the simplest, among the generalizations of Peres algorithm given in [6].

A. 3-Face Peres Function

Assume our die has three faces with values 0, 1, and 2 with probabilities p, q, and r, respectively, so that $p + q + r = 1$. A sequence in $\{0,1,2\}^N$ is considered to be taken from a source of Bernoulli(p, q, r). Denote by $S_{(n_0,n_1,n_2)}$ the subset of $\{0,1,2\}^N$ that consists of strings with n_0 0's, n_1 1's, and n_2 2's. Then

$$\{0,1,2\}^N = \bigcup_{n_0+n_1+n_2=N} S_{(n_0,n_1,n_2)},$$

and each $S_{(n_0,n_1,n_2)}$ is an equiprobable subset of elements whose probability of occurrence is $p^{n_0}q^{n_1}r^{n_2}$.

Consider the functions on $\{0,1,2\}^2$ defined as follows:

x	$\Pr(x)$	$\Psi_1(x)$	$u(x)$	$v(x)$	$w(x)$
00	p^2	λ	0	0	λ
01	pq	0	1	λ	1
02	pr	0	1	λ	2
10	pq	1	1	λ	1
11	q^2	λ	0	1	λ
12	qr	0	1	λ	0
20	pr	1	1	λ	2
21	qr	1	1	λ	0
22	r^2	λ	0	2	λ

TABLE I. Functions for three-face Peres method

The second column of the table shows the probabilities $\Pr(x)$ for $x \in \{0,1,2\}^2$. Note that

$$\Pr(\Psi_1(x) = 0) = pq + qr + rp = \Pr(\Psi_1(x) = 1).$$

Therefore, the output of Ψ_1 can be regarded as a fair coin flip. Extend the three functions Ψ_1, u, and v to $\{0,1,2\}^*$: for an empty string,

$$\Psi_1(\lambda) = u(\lambda) = v(\lambda) = \lambda,$$

for a nonempty even-length input, define (and the same for u and v)

$$\Psi_1(x_1x_2\ldots x_{2n}) = \Psi_1(x_1x_2) * \cdots * \Psi_1(x_{2n-1}x_{2n}),$$

where $*$ is concatenation, and for an odd-length input, drop the last bit and take the remaining even-length bits.

Define

$$\Psi(x) = \Psi_1(x) * \Psi(u(x)) * \Psi(v(x)) * \Psi(w(x)).$$

This function Ψ, recursively defined, with Ψ_1 as its base, is shown to produce unbiased coin flips and is also asymptotically optimal [6].

In order to compute the exact output rate, consider a decomposition of equiprobable set of inputs $S_{(n_0,n_1,n_2)}$: Fix (n_0, n_1, n_2) such that $N = 2n = n_0 + n_1 + n_2$. Let $C(l_1, l_2, l_3; m_0, m_1, m_2)$ be the subset of $S_{(n_0,n_1,n_2)}$ whose elements are the strings that are combination of l_1 01's, l_2 02's, l_3 12's, m_0 00's, m_1 11's, m_2 22's, where each pair is allowed to be transposed. For example,

$$x = 01|00|10|12|00|20|00|10|02|01|00|20|02|10|20|20|11$$

is in $C(5,6,1;4,1,0)$, and

$$\begin{aligned}\Psi_1(x) &= 010110010111,\\ u(x) &= 10110101110111110,\\ v(x) &= 00001,\\ w(x) &= 110212122122.\end{aligned}$$

So,

$$\begin{aligned}\Psi(x) &= 010110010111 * \Psi(10110101110111110)\\ &\quad * \Psi(00001) * \Psi(110212122122)\\ &= 010110010111\\ &\quad * (1000 * \Psi(10110100) * \Psi(1111) * \Psi(1111))\\ &\quad * (\lambda * \Psi(00) * \Psi(00) * \Psi(\lambda))\\ &\quad * (0001 * \Psi(011110) * \Psi(12) * \Psi(2000))\\ &= 010110010111 * (1000 * 10111 * \lambda * \lambda)\\ &\quad * \lambda * (0001 * 011 * 0 * 11)\\ &= 010110010111100010111000101 1011.\end{aligned}$$

Let $l = l_1 + l_2 + l_3$ and $m = m_0 + m_1 + m_2$ so that $n = m + l$. Then we have a decomposition

$$S_{(n_0,n_1,n_2)} = \bigcup_{\substack{n_0=l_1+l_2+2m_0\\ n_1=l_1+l_3+2m_1\\ n_2=l_2+l_3+2m_2}} C(l_1,l_2,l_3;m_0,m_1,m_2). \quad (1)$$

Call the set $C(l_1,l_2,l_3;m_0,m_1,m_2)$ the (n_0,n_1,n_2)-class of type $(l_1,l_2,l_3;m_0,m_1,m_2)$. If $C = C(l_1,l_2,l_3;m_0,m_1,m_2)$, then

$$\begin{aligned}\Psi_1(C) &= \{0,1\}^l\\ u(C) &= S_{(m,l,0)}\\ v(C) &= S_{(m_0,m_1,m_2)}\\ w(C) &= S_{(l_1,l_2,l_3)}.\end{aligned}$$

Lemma 1 (Structure of (n_0,n_1,n_2)-class [6])**.** *In equiprobable set $S_{(n_0,n_1,n_2)}$, the mapping*

$$x \mapsto \Phi(x) = (\Psi_1(x), u(x), v(x), w(x))$$

is one-to-one correspondence between the (n_0,n_1,n_2)-class of type $(l_1,l_2,l_3;m_0,m_1,m_2)$ and $\{0,1\}^l \times S_{(m,l,0)} \times S_{(m_0,m_1,m_2)} \times S_{(l_1,l_2,l_3)}$.

B. Asymptotic Optimality

Now, consider the truncated versions of Peres function, whose recursion depth is bounded by ν, defined as follows:

$$\Psi_\nu(x) = \Psi_1(x) * \Psi_{\nu-1}(u(x)) * \Psi_{\nu-1}(v(x)) * \Psi_{\nu-1}(w(x)),$$

where $\Psi_0(x) = \lambda$. Since x is from Bernoulli(p,q,r), $u(x)$, $v(x)$ and $w(x)$ are of distributions

$$\begin{aligned}U(p,q,r) &= \left(p^2+q^2+r^2, 2(pq+qr+rp), 0\right),\\ V(p,q,r) &= \left(\frac{p^2}{p^2+q^2+r^2}, \frac{q^2}{p^2+q^2+r^2}, \frac{r^2}{p^2+q^2+r^2}\right),\\ W(p,q,r) &= \left(\frac{qr}{pq+qr+rp}, \frac{pq}{pq+qr+rp}, \frac{rp}{pq+qr+rp}\right),\end{aligned}$$

respectively. The average output length per input of $u(x)$, $v(x)$, and $w(x)$ are $\frac{1}{2}$, $\frac{1}{2}(p^2+q^2+r^2)$, and $(pq+qr+rp)$, respectively. So, the rate ρ_ν of Ψ_ν is

$$\begin{aligned}\rho_\nu(p,q,r) =& (pq+qr+rp) + \frac{1}{2}\rho_{\nu-1}(U(p,q,r))\\ &+ \frac{1}{2}(p^2+q^2+r^2)\rho_{\nu-1}(V(p,q,r))\\ &+ (pq+qr+rp)\rho_{\nu-1}(W(p,q,r)),\end{aligned} \quad (2)$$

and, of course, $\rho_1(p,q,r) = pq+qr+rp$ and $\rho_0(p,q,r) = 0$.

Expanding this formula, we obtain, for example,

$$\begin{aligned}\rho_2(p,q,r) =& (pq+qr+rp) + (pq+qr+rp)(p^2+q^2+r^2)\\ &+ \frac{pqr(p+q+r)}{pq+qr+rp} + \frac{p^2q^2+q^2r^2+r^2p^2}{2(p^2+q^2+r^2)},\end{aligned}$$

and the formula for ρ_ν becomes complicated very fast as ν increases, as we can expect from (2).

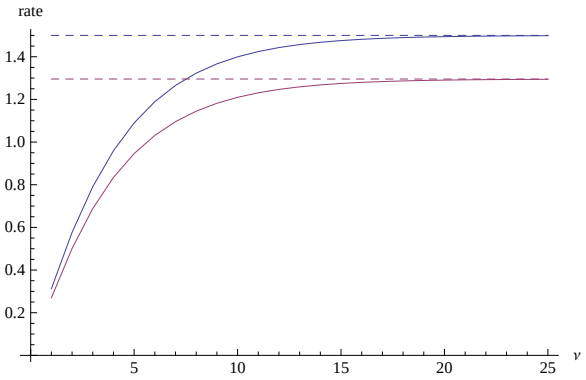

Fig. 1. Rates $\rho_\nu(p,q,r)$ for $\nu = 1,\ldots,25$ and $(p,q,r) = (0.25, 0.25, 0.5)$ and $(p,q,r) = (0.1, 0.3, 0.6)$. The dashed lines indicate the entropy bounds for each value of (p,q,r).

The original (2-face) Peres function [1] was defined as a truncated version and its rate is equal to $\rho_\nu(p,q,0)$. Hence, the (truncated) rate function ρ_ν also generalizes the 2-face case. Since the 2-face Peres function is asymptotically optimal, the corresponding truncated version converges to the Shannon entropy $H(p) = -(p\log_2 p + q\log_2 q)$. We can expect the rate ρ_ν of our 3-face truncated Peres function also converges to $H(p,q,r) = -(p\log_2 p + q\log_2 q + r\log_2 r)$. Fig.1 shows the plot of $\rho_\nu(p,q,r)$ for $\nu = 1,\ldots,25$ and $(p,q,r) = (0.25, 0.25, 0.5)$ and $(p,q,r) = (0.1, 0.3, 0.6)$. Indeed, the rates seem to converge to the corresponding entropy bounds, $H(0.25, 0.25, 0.5) = 1.5$ and $H(0.1, 0.3, 0.6) \approx 1.295$, respectively.

The following theorem, whose proof is given in [6], implies that Ψ is asymptotically optimal.

Theorem 2 ([6])**.**

$$\lim_{\nu\to\infty} \rho_\nu(p,q,r) = H(p,q,r).$$

II. Exact Output Rate

A. Total Output Length on Equiprobable Set $S_{(n_0,n_1,n_2)}$

Define $P(n_0, n_1, n_2)$ to be the total number of output bits over $S_{(n_0,n_1,n_2)}$, that is,

$$P(n_0, n_1, n_2) = \sum_{x \in S_{(n_0,n_1,n_2)}} |\Psi(x)|.$$

Then the rate of Ψ is

$$\begin{aligned}\rho(N) &= \frac{1}{N} \sum_{x \in \{0,1,2\}^N} |\Psi(x)| \Pr(x) \\ &= \frac{1}{N} \sum_{N=n_0+n_1+n_2} P(n_0, n_1, n_2) p^{n_0} q^{n_1} r^{n_2}.\end{aligned}$$

Note that $P(n_0, n_1, n_2)$ is independent on the probability distribution (p, q, r). So, once we compute an appropriate table of values of $P(n_0, n_1, n_2)$, which is computationally the most demanding part, the exact rate $\rho(p, q, r)$ can be easily computed for each (p, q, r). In the following, we give a recursive formula for $P(n_0, n_1, n_2)$ so that its values can be computed, for example, by dynamic programming.

With a bit of abuse of notation, for a class $C(l_1, l_2, l_3; m_0, m_1, m_2)$, use the same symbol P and let

$$P(l_1, l_2, l_3; m_0, m_1, m_2) = \sum_{x \in C(l_1,l_2,l_3;m_0,m_1,m_2)} |\Psi(x)|.$$

Then, by the decomposition (1) we have

$$P(n_0, n_1, n_2) = \sum_{\substack{n_0=l_1+l_2+2m_0 \\ n_1=l_1+l_3+2m_1 \\ n_2=l_2+l_3+2m_2}} P(l_1, l_2, l_3; m_0, m_1, m_2). \quad (3)$$

Now, by the structure lemma, for $C = C(l_1, l_2, l_3; m_0, m_1, m_2)$, the image by Ψ_1 over C is

$$\binom{n}{m, l} \binom{m}{m_0, m_1, m_2} \binom{l}{l_1, l_2, l_3}$$

copies of $\{0, 1\}^l$. So, we have

$$\sum_{x \in C} |\Psi_1(x)| = 2^l \binom{n}{m, l} \binom{m}{m_0, m_1, m_2} \binom{l}{l_1, l_2, l_3} \cdot l.$$

Similarly, we have

$$\begin{aligned}\sum_{x \in C} |\Psi(u(x))| &= 2^l \binom{m}{m_0, m_1, m_2} \binom{l}{l_1, l_2, l_3} P(m, l, 0), \\ \sum_{x \in C} |\Psi(v(x))| &= 2^l \binom{n}{m, l} \binom{l}{l_1, l_2, l_3} P(m_0, m_1, m_2), \\ \sum_{x \in C} |\Psi(w(x))| &= 2^l \binom{n}{m, l} \binom{m}{m_0, m_1, m_2} P(l_1, l_2, l_3).\end{aligned}$$

Since $|\Psi(x)| = |\Psi_1(x)| + |\Psi(u(x))| + |\Psi(v(x))| + |\Psi(w(x))|$, we have

$$\begin{aligned}P(l_1, l_2, l_3; m_0, m_1, m_2) = \\ 2^l \Bigg[&\binom{n}{m, l} \binom{m}{m_0, m_1, m_2} \binom{l}{l_1, l_2, l_3} \cdot l \\ &+ \binom{m}{m_0, m_1, m_2} \binom{l}{l_1, l_2, l_3} P(m, l, 0) \\ &+ \binom{n}{m, l} \binom{l}{l_1, l_2, l_3} P(m_0, m_1, m_2) \\ &+ \binom{n}{m, l} \binom{m}{m_0, m_1, m_2} P(l_1, l_2, l_3) \Bigg].\end{aligned} \quad (4)$$

1) Exploiting Symmetry: If (n_0', n_1', n_2') is a permutation of (n_0, n_1, n_2), then $P(n_0', n_1', n_2') = P(n_0, n_1, n_2)$. Therefore, we need to compute only $P(n_0, n_1, n_2)$ for $n_0 \geq n_1 \geq n_2$.

$$\begin{aligned}P(n_0', n_1', n_2') = P(n_0, n_1, n_2), \quad n_0 \geq n_1 \geq n_2, \\ (n_0', n_1', n_2') \text{ is a permutation of } (n_0, n_1, n_2)\end{aligned} \quad (5)$$

Symmetry in $P(l_1, l_2, l_3; m_0, m_1, m_2)$ is taken care of at this stage.

2) Odd-length Input: In the right-hand side of (4), a recursive call to $P(n_0, n_1, n_2)$ can be made for an odd value of $n_0+n_1+n_2$. In that case, we need to reduce it to even-length, for $n_i > 0$, $i = 0, 1, 2$,

$$\begin{aligned}P(n_0, n_1, n_2) = P(n_0 - 1, n_1, n_2) + P(n_0, n_1 - 1, n_2) \\ + P(n_0, n_1, n_2 - 1), \quad \text{if } n_0 + n_1 + n_2 \text{ is odd,}\end{aligned}$$

and for n_0 and n_1 are positive and $n_2 = 0$,

$$\begin{aligned}P(n_0, n_1, 0) = P(n_0 - 1, n_1, 0) + P(n_0, n_1 - 1, 0), \\ \text{if } n_0 + n_1 \text{ is odd.}\end{aligned}$$

3) Initial Conditions: Clearly, for $n \geq 0$,

$$P(n, 0, 0) = 0. \quad (6)$$

B. Linear Diophantine Equations

Two linear Diophantine equations are involved in the computation. First, given N, we need to find all the nonnegative solutions (n_0, n_1, n_2) such that $N = n_0 + n_1 + n_2$. This is a partition into 3 parts and the solutions can be efficiently generated by methods given in, for example, [7] or [8].

Now, as for the second equation, for a given (n_0, n_1, n_2), we need to generate all the nonnegative integer solutions $(l_1, l_2, l_3; m_0, m_1, m_2)$ of the equations

$$\begin{aligned}n_0 &= l_1 + l_2 + 2m_0, \\ n_1 &= l_1 + l_3 + 2m_1, \\ n_2 &= l_2 + l_3 + 2m_2.\end{aligned} \quad (7)$$

The solutions can be generated efficiently as follows: Since the coefficients for m_0, m_1, and m_2 are dominant, we first list all the possible candidates for (m_0, m_1, m_2) as

$$\begin{aligned}M = \{(m_0, m_1, m_2) \mid m_i \text{ is nonnegative integer s.t.} \\ 0 \leq m_i \leq n_i/2,\ i = 0, 1, 2\}.\end{aligned}$$

For example, given $(n_0, n_1, n_2) = (7, 5, 2)$, the corresponding M is

$$\{(0,0,0),(0,0,1),(0,1,0),(0,1,1),(0,2,0),(0,2,1),$$
$$(1,0,0),(1,0,1),(1,1,0),(1,1,1),(1,2,0),(1,2,1),$$
$$(2,0,0),(2,0,1),(2,1,0),(2,1,1),(2,2,0),(2,2,1),$$
$$(3,0,0),(3,0,1),(3,1,0),(3,1,1),(3,2,0),(3,2,1)\}.$$

Then, for each of these triples (m_0, m_1, m_2), we solve for nonnegative solutions of the equations

$$l_1 + l_2 = n_0 - 2m_0,$$
$$l_1 + l_3 = n_1 - 2m_1,$$
$$l_2 + l_3 = n_2 - 2m_2.$$

This system is non-singular and has a unique *real* solution (l_1, l_2, l_3), and if these l_i's are nonnegative integers, then we take $(l_1, l_2, l_3, m_0, m_1, m_2)$ as a solution. Therefore the number of solutions is bounded by $|M| = (\lfloor n_0/2 \rfloor + 1)(\lfloor n_1/2 \rfloor + 1)(\lfloor n_2/2 \rfloor + 1) \le (N/3 + 1)^3$.

For the case $(n_0, n_1, n_2) = (7, 5, 2)$ given above, the corresponding solutions are now

$$\{(5,2,0,0,0,0),(4,1,1,1,0,0),(5,0,0,1,0,1),$$
$$(3,2,0,1,1,0),(3,0,2,2,0,0),(2,1,1,2,1,0),$$
$$(3,0,0,2,1,1),(1,2,0,2,2,0),(1,0,2,3,1,0),$$
$$(0,1,1,3,2,0),(1,0,0,3,2,1)\}.$$

C. Maximum Output Rate

For any given procedure that converts a length-n input of biased Bernoulli source to unbiased random bits, the maximum average output rate can be obtained, like the rate of Peres algorithm discussed above, by computing the total output lengths on equiprobable sets [4]. In fact, the maximum output rate is obtained by Elias method. For example, for three-face case, let $E_n^3 : \{0,1,2\}^n \to \{0,1\}^*$ be the function corresponding to the Elias method. Then, for (n_0, n_1, n_2) such that $n = n_0 + n_1 + n_2$, the total output length over $S_{(n_0,n_1,n_2)}$

$$Q(n_0, n_1, n_2) = \sum_{x \in S_{(n_0,n_1,n_2)}} |E_n^3(x)|$$

can be computed, from the definition of Elias method, as follows [4]: let the standard binary expansion of $|S_{(n_0,n_1,n_2)}|$ be $\sum_i a_i 2^i$, where a_i is either zero or one. Then

$$Q(n_0, n_1, n_2) = \sum_i i \cdot a_i \cdot 2^i,$$

and the n-maximal output rate for input length n can be computed.

III. Computation Results

Using the recursive definitions for $P(n_0, n_1, n_2)$ and $P(l_1, l_2, l_3; m_0, m_1, m_2)$ given above, we can compute the values of them efficiently using dynamic programming. For example,

$$P(10, 20, 30) = 19\ 38905\ 30631\ 82778\ 17752\ 73600.$$

In comparison,

$$Q(10, 20, 30) = 28\ 46922\ 13778\ 64604\ 61389\ 79776.$$

Fig. 2 shows plots of the exact rates of three-face Peres algorithm $\rho(p, q, r)$, for $(p, q, r) = (0.25, 0.25, 0.5)$ and $(p, q, r) = (0.1, 0.3, 0.6)$, where the input lengths range from 2 to 160. Also shown is the maximal rates of Elias methods in comparison, for the same distributions (p, q, r) and for the same input lenghts.

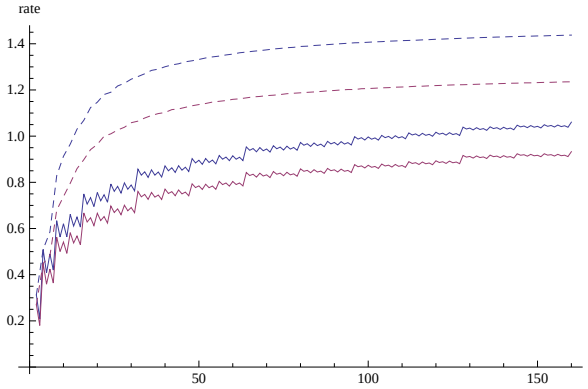

Fig. 2. Exact rates $\rho(p, q, r)$ for $(p, q, r) = (0.25, 0.25, 0.5)$ (shown in blue) and $(p, q, r) = (0.1, 0.3, 0.6)$ (in red), for input lengths $n = 2, \ldots, 160$. Dashed lines are the maximal rates (Elias) for the respective distbutions (again, shown in blue and red).

IV. Remarks

Although the method described here is much more efficient than brute-force calculation over all possible inputs, it still takes a considerable time. For example, it took several hours to obtain the data for Fig. 2 with a decently fast personal computer from the standard of the time when this paper was written.

Acknowledgement

This work was supported in part by the National Research Foundation of Korea (NRF) grant funded by Korean government (No. 2009-0077288).

References

[1] Y. Peres, "Iterating von Neumann's procedure for extracting random bits," *Annals of Statistics*, vol. 20, no. 1, pp. 590–597, 1992.

[2] P. Elias, "The efficient construction of an unbiased random sequence," *The Annals of Mathematical Statistics*, vol. 43, no. 3, pp. 865–870, 1972.

[3] S. Pae and M. C. Loui, "Optimal random number generation from a biased coin," in *Proceedings of the Sixteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, January 2005, pp. 1079–1088.

[4] ——, "Randomizing functions: Simulation of discrete probability distribution using a source of unknown distribution," *IEEE Transactions on Information Theory*, vol. 52, no. 11, pp. 4965–4976, November 2006.

[5] S. Pae, "Exact output rate of Peres's algorithm for random number generation," *Inf. Process. Lett.*, vol. 113, no. 5-6, pp. 160–164, 2013.

[6] ——, "A generalization of Peres's algorithm for generating random bits from loaded dice," 2013, submitted.

[7] D. E. Knuth, *The Art of Computer Programming, Combinatorial Algorithms, Part 1*. Addison-Wesley, 2011, vol. 4A.

[8] A. Nijenhuis and H. S. Wilf, *Combinatorial Algorithms: For Computers and Calculators*. Academic Press, 1978.