Fairing of Biased Coins in Bounded Time

Josh D. Cohen

Technical Report YALEU/DCS/TR-372
March 1985
(revised from manuscript of August 1983)

This work was supported in part by National Science Foundation Grants MCS–8116678 and MCS–8305382.

It is easy to generate random numbers with a uniform distribution when given a "fair coin", i.e. a coin that when flipped has an exactly 50% probability of giving each of *heads* and *tails* and for which the outcomes of successive flips are completely independent. But no coin (mechanical or electronic) is perfectly fair. A "fairing" algorithm is an algorithm which can use a biased coin, possibly many times, and produce a single **heads** or **tails** as its result. Such an algorithm may be desirable since the results it produces may have a smaller bias than the original coin which the algorithm employs. When only a bounded number of biased flips are to be used, perfect fairing is usually not possible. This report, however, will explore several methods of fairing biased coins and analyze some of the advantages, disadvantages, and error properties of each. Some methods of generating specific biases will also be given, and some problems associated with using a biased coin rather than a fair one will be examined. For a particular use, the threshold at which a coin's bias becomes large enough to warrant fairing will be examined.

Introduction

If you are given a biased coin and want to use it (possibly many times) to simulate a single flip of a fair coin, how would you go about it? One might ask first what the bias of the coin is. If the bias is not known a priori, it can be estimated with a statistically significant number of trial flips. If the coin is known to be perfectly 50-50, then you need do nothing special. If there is a fixed bias where, for instance, *tails* is expected with probability exactly $\frac{\sqrt{2}}{2}$, then you can simply flip the coin twice returning **tails** if and only if the coin yields *tails* on both flips.

In most cases, however, it will not be possible, within a fixed number of flips, to generate a perfectly fair coin — even under the assumption that the bias remains fixed and is independent of previous flips and other influences. A more modest goal will be to reduce the bias to within a reasonable limit.

In most of what follows, it shall be assumed that a biased coin is available for which successive flips *are* independent. p will be used to designate the probability that any given flip of the coin will yield *heads*, and $q = 1 - p$ will be used to designate the probability that a given flip will produce tails. Because of the independence assumption, p and q will not vary for a given coin. Finally, $\epsilon = |p - q|$ will be used to denote the *error* (or bias) of the coin.

A *fairing algorithm* is a deterministic procedure which has access to a (possibly) biased coin. The procedure may use the value yielded by the coin, perhaps many times, but no other source of randomness is permitted. A fairing algorithm is not required to terminate, but if it does, it is required to return either **heads** or **tails** as its result. P will be used to designate the probability that a given fairing algorithm returns **heads**, and Q will designate the probability that such an algorithm returns **tails**. P and Q may depend not only on the algorithm itself but also on the error of the coin which is used. $E = |P - Q|$ will denote the error of such an algorithm.

For example, if no fairing is used (i.e., the biased coin is used directly) then $P = p$, $Q = q$, and $E = P - Q = p - q$. The improvement in error that can be obtained will be shown for various techniques to be described.

This report will explore several fairing algorithms and analyze them with respect to how well they reduce E, how many biased coin flips they require, and when it might be desirable to use such an algorithm in preference to using the biased coin directly.

Some Techniques for Fairing

One approach might be called the THRESHOLD technique. The idea is simply to gauge how many *tails* are expected in a given number of flips. If at least that many *tails* are actually obtained, the result is **tails**. But if the number of *tails* obtained is less than expected, the result is **heads**. The THRESHOLD approach is simple but has many subtleties (which shall be explored in more depth later). There are also several disadvantages, including the need to know the bias of the coin ahead of time and the complexity of the calculations which must be performed to obtain the threshold criterion. Although the optimal threshold for a given coin using a preset number of flips can be calculated with a numerical sum, minimizing the error requires optimizing *BOTH* the number of flips and the threshold. It is easily seen that THRESHOLD will not produce a perfect 50-50 fairing except in certain very specialized circumstances (such as in the $\frac{\sqrt{2}}{2}$ case presented above).

Another approach is the XOR technique. The XOR function returns **heads** if the number of *tails* is even and returns **tails** if this number is odd. There are several advantages here over the first method since the bias need not be known a priori and since the resulting error (E) is significantly smaller than THRESHOLD in most cases. It will be seen, however, that with XOR the error is 0 *only* when the coin is fair to begin with.

An alternative approach, usually attributed to vonNeumann [vonN51], might best be called PAIRING. The PAIRING function, unlike the previous two, is asymmetric. It is calculated as follows:

Let $B()$ be the Biased coin function (i.e. a random variable which yields *heads* or *tails*, each with some fixed probability).

Function *PAIR*
  Repeat Until $X \neq Y$
    Let $X = B()$;
    Let $Y = B()$
  Return X

The fundamental notion here is that if the bias of B is fixed, then the (*heads*, *tails*) pair will occur with exactly the same probability as the (*tails*, *heads*) pair. In the other two cases, the pair is discarded and another pair is chosen. Note that each of **heads** and **tails** is returned with probability EXACTLY $\frac{1}{2}$. If the bias is close to 50-50, then approximately 50% of the pairs will yield an immediate result, so one can expect to complete the routine on average in about two iterations. Thus about two pairs are used, and approximately four calls to B (i.e. four flips of the biased coin) may be expected.

To be more precise, let p be the probability that B returns *heads* and let $q = (1 - p)$ be the probability that B returns *tails*. The probability that two calls to B return different values is then $2pq$. Thus, the expected number of pairs needed is $\frac{1}{2pq}$; and the expected number of biased flips is $2 \cdot \frac{1}{2pq} = \frac{1}{pq}$.

With this form of the pairing function, a "bad" sequence of biased flips is one in which each biased flip is the same as its partner. For instance, H H T T T T H H T T H H H H H H would be a bad sequence of flips since each pair chosen would be found to match and thus be discarded. A simple extension to this approach, however, can drastically reduce the number of bad sequences. The key observation is that (just as H T ... and T H ... occur with the same probability) H H T T ... and T T H H ... are equiprobable, and hence, can be used to select between **heads** and **tails**. Thus, the pairing function can be extended to distinguish between these cases.

With a stack of size $\log_2 n$ where n is the number of flips, a memory of all discarded flips can be retained. When two identical groups of flips are encountered, the sense of the groups (all *heads* or all *tails*) is placed on the stack for later comparison.

The Extended PAIRING function looks as follows:

Function *EPAIR*
  Initialize empty stack S
  $I = 0$
  Repeat Until $X \neq Y$
    Let $J = I = I + 1$;
    Let $X = Y = B()$;
    While J **is even and** $X = Y$
      Let $Y =$ **Pop** S;
      $J = J/2$
    Push X **onto** S
  Return X

EPAIR will (again by its symmetry) for any number of iterations yield each of **heads** and **tails** with the same probability. In particular after 2^n flips, for some positive integer n, the only bad sequences are all *heads* or all *tails*. Whereas the original pairing function was able to make 2^{n-1} "trial comparisons" on 2^n biased flips, the extended function makes $2^n - 1$ comparisons on the same number of flips. Thus, the expected number of biased flips needed is just over half the previous number — approximately $\frac{1}{2pq}$.

With this approach, if the sequence H H T is observed, then it is known that the next biased flip will "decide" the outcome. If the next flip is *heads*, then the result is **tails**; if, however, the next flip is *tails*, then the result is **heads**. In general, if the first 2^n biased flips yield the same result, and the first change is seen somewhere between flip 2^n and 2^{n+1}, then the outcome will be decided after at most 2^{n+1} flips, but it may be necessary to complete all 2^{n+1} flips to come to a decision.

Hoeffding and Simons [HoSi70] observed that additional symmetries could be exploited by allowing their algorithm more dynamic flexibility. Instead of H H T H yielding **tails** and H H H T yielding **heads**, for example, the results yielded by these two equiprobable sequences could be switched. This has the effect of causing both H H T H and H H T T to yield **heads**. Thus, when the sequence H H T is produced, the result **heads** can be returned with no additional flips.

Stout and Warren [StWa84] later discovered some further improvements that could be made in this direction. It is, however, a property of all such *even* algorithms (algorithms which balance probabilities by designating an equal number of equiprobable sequences as yielding each of **heads** and **tails**) that they will not terminate as long as all trial flips have so far yielded the same outcome. Since EPAIR will terminate within 2^{n+1} flips if the first change is observed between flip 2^n and flip 2^{n+1}, the expected number of flips used by the EPAIR algorithm is no worse than a factor of two greater than any algorithm from the class of even algorithms.

Analysis of Errors

For PAIR or EPAIR, $P = Q = \frac{1}{2}$, so $E = 0$, and the number of flips N is expected to be small. However, this is not really a fair comparison, since PAIR and EPAIR may require an indefinitely large number of biased flips before terminating, while THRESHOLD and XOR worked with a fixed number of flips.

A perhaps more just means of comparison (and perhaps more useful from both a theoretical and practical viewpoint) may be obtained by bounding the number of biased flips to be allowed.

This will be accomplished by introducing a bounded form of the EPAIR function (BEPAIR). BEPAIR is defined to be the result of executing EPAIR for at most a prescribed number (N) of iterations (flips), if EPAIR terminates in this time. For simplicity, BEPAIR will only accept bounds of the form $N = 2^n$ (n a non-negative integer). In these cases, EPAIR is at least as good as any even algorithm. If EPAIR does not terminate after N flips, then EPAIR must have been given a bad sequence (either all *heads* or all *tails*). There is, in some sense, no information in such a sequence since there is no evidence that the coin is not completely biased (will always yield *heads* or always yield *tails*) — a coin for which no useful fairing is possible. Thus, for lack of additional information, BEPAIR will be defined to be **heads** on a sequence of N *heads* and **tails** on a sequence of N *tails*. This is preferable to defining the same result on both bad sequences (since all other sequences are balanced against each other, and this allows the two bad sequences to offset each other somewhat). If p and q are (perchance) exactly $\frac{1}{2}$, then the two bad sequences occur with the same probability, so the balance is maintained (this gives the desirable property generally not attained by THRESHOLD that fairing will not hurt an already fair coin).

For BEPAIR, let P_G (P-Good) be the probability of BEPAIR returning **heads** due to normal termination of EPAIR, and let Q_G be the corresponding probability of **tails** due to a "good" termination of EPAIR. Let P_B and Q_B be the respective probabilities of **heads** and **tails** in termination due to a bad sequence. Thus,

$$P = P_G + P_B \quad \text{and} \quad Q = Q_G + Q_B \quad .$$

Observe also that $P_G = Q_G$ (by symmetry). Thus,

$$E = |P - Q| = |P_B - Q_B| = |p^N - q^N| \quad .$$

In the XOR technique with the number of flips fixed at N, P, the probability of returning **heads** is given by

$$P = \sum_{\substack{i=0 \\ i \text{ even}}}^{N} \binom{N}{i} p^{N-i} q^i \quad \text{and that of returning tails is} \quad Q = \sum_{\substack{i=0 \\ i \text{ odd}}}^{N} \binom{N}{i} p^{N-i} q^i \quad .$$

Assuming, without loss of generality, that $0 \le q \le p \le 1$; it is then easily seen that

$$(p-q)^N = \sum_{i=0}^{N} (-1)^i \binom{N}{i} p^{N-i} q^i = \sum_{\substack{i=0 \\ i \text{ even}}}^{N} \binom{N}{i} p^{N-i} q^i - \sum_{\substack{i=0 \\ i \text{ odd}}}^{N} \binom{N}{i} p^{N-i} q^i = P - Q \quad .$$

Thus, $E_{\text{XOR}} = (p-q)^N$.

This result may be somewhat surprising, since the following lemma will show that when allowed only a bounded number of biased flips, XOR is at least as good as (and is, in fact, usually *much* better than) PAIRING!

LEMMA

If $0 \le q \le p \le 1$ and $N \ge 1$, then $(p-q)^N \le p^N - q^N$.

PROOF

Since $0 \le q \le p \le 1$,

$$p^N - q^N \ge (p-q)p^{N-1} \ge (p-q)(p-q)^{N-1} = (p-q)^N . \qquad \blacksquare$$

It may be noted that, unless $p = q$, the above inequalities can be strengthened. In fact, for large N and $q < p$, $(p-q)^N \ll p^N - q^N$. If, for example, $p = 0.6$, $q = 0.4$, and $N = 64$, then $0.6^{64} \approx 6 \times 10^{-15}$, $0.4^{64} \approx 3 \times 10^{-26}$, and $0.2^{64} \approx 2 \times 10^{-45}$. Thus, the error of BEPAIR is $p^{64} - q^{64} \approx p^{64} \approx 6 \times 10^{-15}$. While the error of XOR is only $(p-q)^{64} \approx 2 \times 10^{-45}$.

This indicates that the error of XOR is much less than the error given by the two "bad" PAIRING sequences. This may seem somewhat counter-intuitive, but it suggests that although PAIRING may be better in some instances, if one places a bound on the number of times one is willing to flip the biased coin, then XOR is at least as good a method of fairing as is PAIRING.

Where do THRESHOLD techniques fit in? It was observed earlier that for at least one specific bias ($q = \frac{\sqrt{2}}{2}, \epsilon = \sqrt{2} - 1$), a threshold approach gives a perfect fairing, whereas the error formulas for XOR and BEPAIR show that the error with these techniques is zero if and only if the coin is fair to begin with. Thus, a threshold technique can be better than these other approaches.

THRESHOLD, however, can be much worse. For a fixed number of biased flips N and a fixed p and q, define the set of partial sums S_k by

$$S_k = \sum_{i=0}^{k-1} \binom{N}{i} p^{N-i} q^i \quad .$$

If k is chosen as the threshold (recall that this means that less than k *tails* yields **heads** and that k or more *tails* yields **tails**), then $Q = S_k$ and $P = 1 - S_k$, so $E = |1 - 2S_k|$. If for some integer k $(0 \le k \le N+1)$ $S_k = \frac{1}{2}$, then k is a perfect threshold. If not, then k should be chosen so as to minimize $|S_k - \frac{1}{2}|$. In the worst case, $\frac{1}{2}$ lies in the middle of some interval. In this case,

$$\frac{1}{2} - S_k = |\frac{1}{2} - S_{k+1}| = S_{k+1} - \frac{1}{2} = S_k + \binom{N}{k} p^{N-k} q^k - \frac{1}{2} \quad ;$$

so

$$\binom{N}{k} p^{N-k} q^k = 1 - 2S_k \quad ,$$

and

$$E = |P - Q| = 1 - 2S_k = \binom{N}{k} p^{N-k} q^k \quad .$$

Hence, the THRESHOLD error is bounded by

$$\max_i \binom{N}{i} p^{N-i} q^i \quad ,$$

and this bound is can be achieved. Thus, THRESHOLD can be **much** worse than XOR or PAIRING.

There is an interesting phenomenon observed here, and that is (unlike in XOR or PAIRING) increasing the number of biased flips may actually *increase* the error. A simple example is seen when the original coin is fair. With one flip, a threshold of 1 (less than 1 *tails* yields **heads**; at least 1 *tails* yields **tails**) gives a perfect fairing. However, no threshold will give a perfect fairing with two flips of this coin. In fact, $E = \frac{1}{2}$ is the best that can be achieved.

This brings up an interesting question of how to simultaneously optimize the number of flips and the threshold so as to minimize the error. As yet, we have no good approaches to this problem. However, the problem seems to have a flavor similar to rational approximation by continued fractions and a similar approach may be fruitful here.

THRESHOLDing may be generalized in several ways. First, keeping it symmetric, one may partition the integers in $0, \ldots, N$ into two sets. One of these sets is designated **heads** and the other

tails. The number (out of N) of biased flips which result in *tails* serves as the index to determine which of the two sets (and hence which result) is the case. XOR is just the special case of this method in which the partition splits **heads** and **tails** according to whether the integer is even or odd, respectively. Optimal choice of N and the partition seems even more difficult in this more general case than in the original.

The restricted problem of determining whether or not a a perfect 50-50 split is possible (much less finding an optimum split) is a special case of the PARTITIONING problem (can a set of integers be partitioned into two sets such that the sum of the integers in the two sets are equal). The general problem is NP-Complete; however, this special case could be tractible. As yet, we have no results on the complexity of the problem of finding an optimum split.

The technique may be generalized yet further if asymmetry is to be tolerated. This is the most general possibility, however, for it simply associates one of **heads** or **tails** with each sequence of N biased flips. Bounded pairing certainly falls under this umbrella as does any function which depends solely on a sequence of N biased flips (where N is fixed).

Even with this generality, however, most biased coins can not be faired perfectly in bounded time. If the probability of *heads* is p, then fairing it perfectly in a bounded number of flips would imply the existence a non-trivial algebraic equation over the rationals with p as a root. This is, of course, impossible if p is transcendental. Thus, if a bound is placed on the number of biased flips to be allowed, then perfect fairing is not possible in general.

The Independence Assumption

For any of the above methods to work, it is essential that each element of the sequence of biased flips used be independent of the others. In a natural setting, this may not always be the case. For instance, if every flip has 60% probability of being the same as the last flip and 40% probability of being different, then these methods as given will not work despite the fact the the coin is dependant on only one previous flip and that this dependance is of a simple form. However, in this case an independent sequence may be obtained by considering whether or not each flip was the same as the previous. This will give an independent sequence of ***same***s and ***different***s on which the above techniques may be performed. As long as some binary sequence may be derived from the original sequence in which each element has a fixed probability of taking on each of the two possible values, then these techniques are applicable. *heads* and *tails* need not occur with the same probability, it is only necessary that the probability of each of *heads* and *tails* does not vary.

Samuelson [Samu68] showed how to generate an independent sequence from any sequence that was generated by a fairly general class of Markov processes. The trick is to focus attention on just one state. Blum [Blum84] shows that an obvious generalization of this technique fails for a subtle reason and then presents a somewhat less intuitive generalization which allows all states to be considered simultaneously — giving a far more efficient methodology.

Biasing of Fair and Biased Coins

Occasionally it is desirable to have coins with a bias that is not 50-50. Many of the approaches given here can be easily extended to manage this possibility.

A simple THRESHOLD technique can adapt directly. The threshold must merely be chosen so as to approximate the desired probability rather than $\frac{1}{2}$.

generated changes. The error properties of a coin do not change while it is being used to generate a specific n bit number.

The following theorem gives the asymptotic character of $\mathcal{E}$ under the above conditions for all possible positive values of k (note that $k \leq 0$ implies a constant or increasing error which, in the limit, sends $\mathcal{E}$ to 1).

THEOREM

If n bit numbers are generated at "random" using a coin with yields 0 with probability p and 1 with probability q, and if the error $E = p - q = \frac{c}{n^k}$ (c and k arbitrary positive constants), then the worst case error of generating an element of any given set consisting of exactly $\frac{1}{2}$ of all n-bit numbers goes (as $n \to \infty$) to

$$\mathcal{E} = \begin{cases} 1, & \text{if } k < \frac{1}{2}; \\ \operatorname{erf}\left(\frac{c}{\sqrt{2}}\right), & \text{if } k = \frac{1}{2} \\ 0, & \text{if } k > \frac{1}{2}. \end{cases} \qquad (\text{for } c = 1 \text{ this is } 0.6827\ldots);$$

PROOF

By applying the Central Limit Theorem, one can calculate the number of standard deviations between the expected number of 0's and $\frac{n}{2}$. Since the worst case occurs precisely when less than $\frac{n}{2}$ 0's are obtained, this will be enough to find the probability of generating less than $\frac{n}{2}$ 0's and hence the error $\mathcal{E}$. Since the expected number of 0's is

$$np = n\left(\frac{1}{2} + \frac{c}{2n^k}\right),$$

and the size of a standard deviation is

$$\sqrt{npq} = \sqrt{n\left(\frac{1}{2} + \frac{c}{2n^k}\right)\left(\frac{1}{2} - \frac{c}{2n^k}\right)} = \sqrt{n\left(\frac{1}{4} - \frac{c^2}{4n^{2k}}\right)} \quad ;$$

the number of standard deviations is

$$\frac{n\left(\frac{1}{2} + \frac{c}{2n^k}\right) - \frac{n}{2}}{\sqrt{n\left(\frac{1}{4} - \frac{c^2}{4n^{2k}}\right)}} = \frac{\frac{c\sqrt{n}}{n^k}}{\sqrt{1 - \frac{c^2}{n^{2k}}}} = \frac{c}{\sqrt{n^{2k-1} - \frac{c^2}{n}}} \quad .$$

When the limit as n goes to infinity is passed, it is quickly seen that for $k > 1/2$, the number of standard deviations goes to 0; for $k < 1/2$, the number of standard deviations goes to infinity; and for $k = 1/2$, the number of standard deviations goes to c. In this latter case, the probability of landing within c standard deviations of the mean in a normal distribution is given by $\operatorname{erf}\left(\frac{c}{\sqrt{2}}\right)$. This gives the desired result.

∎

Thus, if $E < \frac{c}{\sqrt{n}}$ can be maintained, then elements of the desired set are obtained with some non-vanishing probability. If the original coin has error ϵ bounded by any function of the form $\frac{c}{n^k}$, it can be improved by use of the XOR algorithm with sufficiently many flips to produce a "new" coin with error bounded by $\frac{c'}{\sqrt{n}}$. Let ϵ be the error of the original coin. By applying XOR to m flips of the original coin, a new coin is produced with error ϵ^m. Since $\epsilon \leq \frac{c}{n^k}$, it is only required

XOR can be generalized to approximate the rational probability $\frac{a}{b}$ by flipping a large number of (fair or biased) coins and returning **heads** if and only if the number of *heads* obtained is less than a when taken modulo b.

PAIRING approaches may also be extended, but the probabilities efficiently attainable seem to be more limited. A $\frac{1}{3} - \frac{2}{3}$ split may be obtained by flipping a (fair or biased) coin three times rather than two in each iteration. If all flips are the same, the process is repeated. If not, the location of the "odd" flip (the one different from the other two) can be used to decide which third to return. This can in a straightforward way be generalized with a stack as before to force all *heads* or all *tails* to become the only bad sequences. If a bias of $\frac{a}{b}$ is desired, then one can exploit the fact that for fixed m and n, all sequences of m *heads* and n *tails* are equiprobable. If the number, $\binom{m+n}{m}$, of such sequences is greater than b, then any b such sequences can be designated as terminating. a of these are designated **heads** and the remaining $b-a$ are designated **tails**. The remaining (non-terminating) sequences require additional flipping. This approach, however, becomes less manageable as the desired probabilities become less regular.

All of the above techniques generalize in a natural way to allow the possibility of more than two distinct outcomes. The outcomes can be equiprobable or biased in any desired manner.

When is Fairing Worthwhile?

One question which arises immediately is "How much can a biased coin hurt me?" A common context is the following. Let D_n be the domain of binary integers of n digits. Let C_n be a subset of D_n containing exactly half the elements of D_n. By flipping coins, elements of D_n will be randomly chosen in hopes of finding members of C_n. *This situation occurs in primality testing as well as many other algorithms and is the basis of the Random Polynomial-Time complexity class RP* (see [Gill77]).

Let p be the probability that the coin yields 0 and q the probability that the coin yields 1. Assume that $p \geq q$ and $E = p - q$. The worst case, assuming for simplicity that n is odd, is where $C_n = \{x \in D_n : x \text{ has more 1's than 0's}\}$. If n is even, then exactly half of the sequences of D_n with the same number of 1's and 0's must be placed in C_n (those which begin with a 0, for instance). The error is comparable to the case of odd n, but the analysis of this case requires dragging around an extra term. It is simpler, therefore, to consider only the case when n is odd.

The probability that an integer x which is generated by n flips of this coin will land in C_n is

$$\sum_{i=0}^{\lfloor n/2 \rfloor} \binom{n}{i} p^i q^{n-i} = \sum_{i=0}^{\lfloor n/2 \rfloor} \binom{n}{i} \left(\frac{1}{2} + \frac{E}{2}\right)^i \left(\frac{1}{2} - \frac{E}{2}\right)^{n-i} .$$

So, the error $\mathcal{E} = (\text{prob. } x \notin C_n) - (\text{prob. } x \in C_n) = 1 - 2 \cdot (\text{prob. } x \in C_n)$ is given by

$$\mathcal{E} = 1 - 2 \sum_{i=0}^{\lfloor n/2 \rfloor} \binom{n}{i} \left(\frac{1}{2} + \frac{E}{2}\right)^i \left(\frac{1}{2} - \frac{E}{2}\right)^{n-i} = 1 - \left(\frac{1}{2}\right)^{n-1} \sum_{i=0}^{\lfloor n/2 \rfloor} \binom{n}{i} (1+E)^i (1-E)^{n-i} .$$

It is clear that for fixed $E > 0$, the error $\mathcal{E}$ goes to 1 as n goes to infinity . It is also clear that if the number of bits n is kept fixed, then $\mathcal{E}$ goes to 1 as E goes to 1. Thus, to find a useful asymptotic result, it is necessary to bind E and n in some way. A simple relationship is to let E be an inverse polynomial function of n given by $E = \frac{c}{n^k}$, for some positive c and k. *Note: It is important to recognize that when generating an n bit number, the error for each bit remains constant. The $E = \frac{c}{n^k}$ relationship describes how the error varies as the number of bits to be*

that $\epsilon^m < \left(\frac{c}{n^k}\right)^m = \frac{c'}{n^{mk}} < \frac{c'}{\sqrt{n}}$. This is satisfied when $mk > \frac{1}{2}$. So $m = \lceil \frac{1}{2k} \rceil$ gives a sufficient number of flips to be XORed to produce each bit.

Some Approximations

The above result gives the asymptotic character of the error function, but in a practical case, one may be using a coin with a certain known bias and want to decide whether or not the probability of generating elements of a particular set (with a given number of flips) is enhanced by fairing. To answer this question, one is again confronted with the equation

$$\mathcal{E} = 1 - \left(\frac{1}{2}\right)^{n-1} \sum_{i=0}^{\lfloor n/2 \rfloor} \binom{n}{i} (1+E)^i (1-E)^{n-i} \quad .$$

For $E \ll \frac{1}{n}$ $\left(n \ll \frac{1}{E}\right)$, the first few terms of the binomial expansion give the following approximation

$$\begin{aligned}
\mathcal{E} &\approx 1 - \left(\frac{1}{2}\right)^{n-1} \sum_{i=0}^{\lfloor n/2 \rfloor} \binom{n}{i} [1 + iE][1 - (n-i)E] \\
&\approx 1 - \left(\frac{1}{2}\right)^{n-1} \sum_{i=0}^{\lfloor n/2 \rfloor} \binom{n}{i} [1 + (2i-n)E] \\
&= -\left(\frac{1}{2}\right)^{n-1} \sum_{i=0}^{\lfloor n/2 \rfloor} \binom{n}{i} (2i-n)E \\
&= -\left(\frac{1}{2}\right)^{n-1} \left[2 \sum_{i=0}^{\lfloor n/2 \rfloor} \binom{n}{i} i - n \sum_{i=0}^{\lfloor n/2 \rfloor} \binom{n}{i}\right] E \\
&= -\left(\frac{1}{2}\right)^{n-1} \left[2n \sum_{i=0}^{\lfloor n/2 \rfloor} \binom{n-1}{i-1} - n2^{n-1}\right] E \\
&= -\left(\frac{1}{2}\right)^{n-1} \left[2n \sum_{i=0}^{\lfloor n/2 \rfloor - 1} \binom{n-1}{i} - n2^{n-1}\right] E \\
&= -\left(\frac{1}{2}\right)^{n-1} \left[2n2^{n-2} - n\binom{n-1}{\frac{n-1}{2}} - n2^{n-1}\right] E \qquad \text{(since for odd } n,\ \lfloor n/2 \rfloor = \tfrac{n-1}{2}\text{)} \\
&= -\left(\frac{1}{2}\right)^{n-1} \binom{n-1}{\frac{n-1}{2}} nE
\end{aligned}$$

Sterling's formula gives the relation

$$\binom{N}{\frac{N}{2}} \approx \frac{2^{N+1}}{\sqrt{2\pi N}} \quad .$$

So, this gives

$$\mathcal{E} \approx \left(\frac{1}{2}\right)^{n-1} \frac{2^n}{\sqrt{2\pi\,(n-1)}} nE = \frac{2nE}{\sqrt{2\pi\,(n-1)}} \approx \sqrt{\frac{2}{\pi}}\sqrt{n}E \quad .$$

Thus, if the condition $E \ll \frac{1}{n}$ is maintained, then for fixed E, $n \ll \frac{1}{E} \leq \frac{1}{E^2}$ (since $0 < E \leq 1$). So, $\mathcal{E}$ is bounded by a fixed constant. Similarly, when n is fixed, $E \ll \frac{1}{n} \leq \frac{1}{n^2}$ (since $n \geq 1$). So again, $\mathcal{E}$ is bounded by a constant. Therefore, $E \ll \frac{1}{n}$ implies that $\mathcal{E}$ is bounded above by $\sqrt{\frac{2}{\pi}} < 0.8$. This would imply that in excess of one in ten sequences of flips will produce an element of C_n, one fifth of the expectation if the coin were perfectly fair.

In particular, if $\mathcal{E} < \sqrt{2} - 1$ can be achieved, then at least one out of every two sequences of flips (on average) will yield an element of C_n. This is attained when (in addition to $E \ll \frac{1}{n}$) $E < (\sqrt{2}-1)\sqrt{\frac{\pi}{2n}}$, or alternately when $n < \frac{\pi}{2E^2}(3 - 2\sqrt{2})$. But since E is assumed to be *much* smaller then $\frac{1}{n}$, these conditions are already satisfied. Thus, if the original coin to be used satisfies $\epsilon \ll \frac{1}{n}$, then two sequences of flips ($2n$ total flips) will, with probability greater than $\frac{1}{2}$, yield an element of C_n. If it is presumed that any fairing method that might be used would require at least 2 "biased" flips for every "faired" flip returned, then $2n$ flips would yield at most one element of D_n. Even if the fairing were perfect, this element would only have a 50-50 chance of landing in C_n. Therefore, when $\epsilon \ll \frac{1}{n}$, the best results are obtained by using the coin directly rather than "spending flips" fairing the coin.

Another bound which is especially useful for coins with large errors may be obtained by the following chain of inequalities.

$$\begin{aligned}
&\sum_{i=0}^{\lfloor n/2 \rfloor} \binom{n}{i} (1+E)^i (1-E)^{n-i} \\
&\leq \sum_{i=0}^{\lfloor n/2 \rfloor} \binom{n}{i} (1+E)^{\frac{n}{2}} (1-E)^{\frac{n}{2}} \\
&= (1-E^2)^{\frac{n}{2}} \sum_{i=0}^{\lfloor n/2 \rfloor} \binom{n}{i} \\
&= 2^{n-1} (1-E^2)^{\frac{n}{2}} \\
&= 2^{n-1} (1-E^2)^{\frac{1}{E^2} \cdot \frac{nE^2}{2}} \\
&\leq 2^{n-1} e^{-\frac{nE^2}{2}} \qquad \text{(where } e \text{ is Euler's constant)}
\end{aligned}$$

So,

$$\mathcal{E} = 1 - \left(\frac{1}{2}\right)^{n-1} \sum_{i=0}^{\lfloor n/2 \rfloor} \binom{n}{i} (1+E)^i (1-E)^{n-i} \geq 1 - (1-E^2)^{\frac{n}{2}} \geq 1 - e^{-\frac{nE^2}{2}} \quad .$$

In particular, if $E < \frac{c}{n^k}$ (for $k < \frac{1}{2}$), an alternate proof of the first case of the previous theorem is seen.

Also,

$$\sum_{i=0}^{\lfloor n/2 \rfloor} \binom{n}{i} (1+E)^i (1-E)^{n-i}$$

$$= (1-E)^n \sum_{i=0}^{\lfloor n/2 \rfloor} \binom{n}{i} \frac{(1+E)^i}{(1-E)^i}$$

$$\geq (1-E)^n \sum_{i=0}^{\lfloor n/2 \rfloor} \binom{n}{i}$$

$$= 2^{n-1} (1-E)^n$$

So,

$$\mathcal{E} = 1 - \left(\frac{1}{2}\right)^{n-1} \sum_{i=0}^{\lfloor n/2 \rfloor} \binom{n}{i} (1+E)^i (1-E)^{n-i} \leq 1 - (1-E)^n \quad .$$

Thus,

$$1 - (1-E^2)^{\frac{n}{2}} \leq \mathcal{E} \leq 1 - (1-E)^n \quad .$$

An example may help to demonstrate how fairing can be beneficial here.

Suppose one is given an extremely biased coin with error $\varepsilon = \frac{1}{2}$, that is, the coin yields 0 with probability $\frac{3}{4}$ and 1 with probability $\frac{1}{4}$. Suppose further that one is trying to generate elements of the set $C_3 = \{3$ digit binary integers of which at least two digits are 1's$\}$.

Using the coin directly ($E = \varepsilon$), one finds that the probability of generating an element of C_3 is $\frac{5}{32}$. If, however, the coin is faired first by flipping it twice and XORing; then the previous analysis shows that $E = \varepsilon^2 = \frac{1}{4}$, that is, the faired result will be 0 with probability $\frac{5}{8}$ and 1 with probability $\frac{3}{8}$ (note that the coin remains biased towards 0 — we could have "cheated" by defining the algorithm so as to reverse the bias, but this would hardly have been fair). Using this coin, the probability of generating an element of C_3 is $\frac{81}{256}$. In this case, one sees with six flips of the original coin, the probability of finding a member of C_3 by fairing first is $\frac{81}{256}$; while without fairing the original coin, six flips will allow two chances to find a member of C_3, each with a success rate of $\frac{5}{32}$. This gives a probability of $\frac{295}{1024}$ of finding at least one member of C_3 (with fairing, the probability was $\frac{81}{256} = \frac{324}{1024}$). Thus, given the same number of physical coin flips (6 in this case), one gets better results by "using" some flips to improve the balance of the coin. Given 12 flips in the above example, the probability is in favor (about 53%) of finding a member of C_3 if fairing with XOR is used and is against (about 49%) finding a member of C_3 when no fairing is used.

Fairing is, therefore, advantageous in some instances. However, in other instances, it is not worth the cost. It seems that the $\frac{1}{\sqrt{n}}$ rule above works well: if the error (E) of the coin is less than the square root of the inverse of the number of bits in the random numbers being generated (n), then fairing is probably not worthwhile; if, however, $E > \frac{1}{\sqrt{n}}$, then fairing is likely to produce better results.

Acknowledgement

Many thanks go to Dana Angluin, Andrie Broder, Neil Immerman, David Lichtenstein, Lenny Pitt, Gregory Sullivan, David Wittenberg, and especially to my advisor Mike Fischer for their comments, criticisms, ideas, suggestions, and other help in making this work possible.

Bibliography

[Blum84] Blum, Manuel. "Independent Unbiased Coin Flips From a Correlated Biased Source: a Finite State Markov Chain." 25th *Symposium on Foundations of Computer Science* pp. 425-433.

[Gill77] Gill, John. "Computational complexity of probabilistic Turing machines." *SIAM Journal on Computing* Vol. 6, No. 4, pp. 675-695.

[HoSi70] Hoeffding, Wassily and Simmons, Gordon. "Unbiased Coin Tossing With a Biased Coin." *Annals of Mathematical Statistics.* Vol. 41, No. 2, pp. 341-352.

[Samu68] Samuelson, Paul A. "Constructing an Unbiased Random Sequence." *Journal of the American Statistical Association.* Vol. 63, No. 324, pp. 1526-1527.

[StWa84] Stout, Quentin F. and Warren, Bette "Tree Algorithms for Unbiased Coin Tossing With a Biased Coin." *Annals of Probability.* Vol. 12, No. 1, pp. 212-222.

[vonN51] von Neumann, John (written by George E. Forsythe). "Various Techniques Used in Connection With Random Digits." *J. Res. Nat. Bur. Stand. Appl. Math. Series.* Vol. 12, pp. 36-38.
(Also found in *Proceedings of the Symposium on "Monte Carlo Method."* held June-July 1949 in Los Angeles, CA. (Chapter 13); and in *John von Neumann: Collected Works.* Vol. 5, Pergamon Press (1963). pp. 768-770.)