ON THE POWER OF TWO-POINTS BASED SAMPLING

Benny Chor * Oded Goldreich **

MIT – Laboratory for Computer Science

Cambridge, Massachusetts 02139

Abstract — The purpose of this note is to present a new sampling technique and to demonstrate some of its properties. The new technique consists of picking two elements at random, and deterministically generating (from them) a long sequence of pairwise independent elements. The sequence is guarantees to intersect, with high probability, any set of non-negligible density.

1. Introduction

In recent years the role of randomness in computation has become more and more dominant. Randomness was used to speed up sequential computations (e.g. primality testing, testing polynomial identities etc.), but its effect on parallel and distributed computation is even more impressive. In either cases the solutions are typically presented such that they are guarateed to produce the desired result with some non-negligible probability. It is implicitly suggested that if a higher degree of confidence is required the algorithm should be run several times, each time using different coin tosses. Since the coin tosses for the distinct runs are independent random variables, the probability that no run yield the desired result goes exponentially down with the number of runs. This means that we can buy higher degree of confidence in the cost of more coin tosses.

Karp and Pippenger [14] have raised the problem of a *time-randomness tradeoff* in this setting. In particular, can one increase the degree of confidence without increasing the number of coin tosses. They demonstrated an affirmative answer to this question by using an explicitly constructed expander. In this note we present a simpler solution. Our solution uses a deterministic construction of an arbitrarily long sequence of pairwise independent sample points from two independent random sample points. Our solution is suitable for practical applications.

---

* Research supported in part by an IBM Graduate Fellowship and a Bantrell Postdoctoral Fellowship.

** Research supported in part by a Weizmann Postdoctoral Fellowship. On leave from the Computer Science Dept., Technion, Israel.

Both solutions (i.e. [14] and ours) are base on viewing a randomized algorithm as a deterministic algorithm with two inputs: the "true input" (denoted x) and a "random input" (denoted y). The length of the "random input" is polynomial in the length of x. The deterministic algorithm is almost identical to the random algorithm, the only difference is that instead of tossing a coin the deterministic algorithm uses the next bit of the "random input". Without loss of generality we may assume that for each x, all "random inputs" y given to the algorithm with x, are of the same length (denoted $lr(x)$). Let us now fix a "true input" x. We say that a "random input" y is *good* (for x) if the algorithm running on the inputs x and y produces the required result. We know that a non-negligible fraction of the strings of length $lr(x)$ are good for x. But we know very little about the structure of the subset of good strings. Nevertheless, all we need is to sample the set of $lr(x)$-bit strings in a search for a string which is good for x.

Thus, the question reduces to that of sampling a large population in order to find a good element. Note that we have a fast procedure (trivially induced by the random algorithm) to check whether an element is good. Our solution to the sampling problem consists of generating a long sequence of pairwise independent random elements which will be used as the sample points. The sequence is generated deterministically out of two independent randomly chosen elements. The probability that no element in the sequence is good goes down linearly with the length of the sequence.

2. Formal Framework

Consider a large universe U, containing a fixed subset S of substantial density $\rho = |S|/|U|$ (for example $\rho = \frac{1}{2}$). Suppose one wishes to find an element of S, while having no information about the structure of S (except for its density). An exhaustive search through U would do the job, but is too expensive if U is large. In fact, any deterministic algorithm could be defeated by certain choices of S. This calls for the use of random sampling.

By independently choosing k sampling elements out of U, an element of S can be found with very high probability (i.e. $1-(1-\rho)^k$). The underlying structure of this solution consists of two primitives. *Picking an element at random* out of the space U (with uniform probability distribution), and *checking whether a given element is in S*. We introduce a third primitive: *deterministic operations on elements of U*. The sampling technique presented in this note makes extensive use of the two deterministic primitives, allowing to use the randomizing primitive only twice.

For simplicity, we assume that $U = Z_p$, the set of residues modulo a prime p, and $S \subset Z_p$.

3. The New Technique: Two-Points Based Sampling

Construction:

Choose two random independent elements (x and y) in Z_p (with uniform probability).
Compute the residues $r_i \stackrel{\text{def}}{=} x + iy \bmod p$, for $1 \le i \le L$.

We now demonsrate a lower bound on the probability that at least one $r_i \in S$. This is done by first showing that the r_i's are pairwise-independent random variables and next by applying a stadard probabilistic argument.

Lemma: Let $2 \leq L < p$. Then the r_i's are pairwise-independent random variables, each uniformly distributed in Z_p.

proof:

Note that x and y are independent random variables with uniform probability distribution over Z_p . Thus, for every $a, b \in Z_p$, $Pr(x \equiv a) = \frac{1}{p}$, $Pr(y \equiv b) = \frac{1}{p}$ and $Pr(x \equiv a \wedge y \equiv b) = \frac{1}{p^2}$.

First we show that each r_i is a random variable with uniform probability distribution over Z_p. This is the case since

$$\begin{aligned} Pr\left(r_i \equiv c\right) &= Pr\left(x + iy \equiv c\right) \\ &= \sum_{b \in Z_p} Pr\left(y \equiv b\right) \cdot Pr\left(x + iy \equiv c \,|\, y \equiv b\right) \\ &= \sum_{b \in Z_p} Pr\left(y \equiv b\right) \cdot Pr\left(x \equiv c - ib\right) = p \cdot \frac{1}{p} \cdot \frac{1}{p} = \frac{1}{p} \end{aligned}$$

We next show that the two random variables r_i and r_j are statistically independent (for $1 \leq i \neq j \leq L$). For every $a, b \in Z_p$, the equations $x + iy \equiv a \pmod p$ and $x + jy \equiv b \pmod p$ have a unique solution in terms of $x, y \in Z_p$. In other words, the mapping

$$\begin{pmatrix} x \\ y \end{pmatrix} \rightarrow \begin{pmatrix} 1 & i \\ 1 & j \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} x + iy \\ x + jy \end{pmatrix} \equiv \begin{pmatrix} r_i \\ r_j \end{pmatrix}$$

is a bijection of $Z_p \times Z_p$ onto itself. Thus, for every $a, b \in Z_p$, $Pr(r_i \equiv a \wedge r_j \equiv b) = \frac{1}{p^2} = Pr(r_i \equiv a) \cdot Pr(r_j \equiv b)$. So r_i, r_j are independent. $\square$

Theorem: Let $2 \leq L < p$. Then with probability $1 - \frac{1-\rho}{\rho L}$, at least one of the r_i's is in S.

proof: For $1 \leq i \leq L$, let

$$\zeta_i = \begin{cases} 1 & \text{if } r_i \in S \\ 0 & \text{otherwise} \end{cases}$$

By the Lemma, each r_i is uniformly distributed in Z_p. Thus

$$Exp(\zeta_i) = Pr(\zeta_i = 1) = \rho$$

and

$$Var(\zeta_i) = Exp\left(\left(\zeta_i - Exp(\zeta_i)\right)^2\right) = \rho(1 - \rho) \; .$$

By the Lemma, for any $i \neq j$, r_i and r_j are independent random variables. Therefore ζ_i and ζ_j are also independent random variables, $i \neq j$. (Whenever the same function is applied to two independent random variables, the two results are independent random variables).

We calculate the probability that no r_i is in S.

$$Pr\left(\sum_{i=1}^{L}\zeta_i = 0\right) \leq Pr\left(\left|\frac{1}{L}\sum_{i=1}^{L}\zeta_i - Exp(\zeta_i)\right| \geq \rho\right)$$

Applying Chebyshev's inequality (see Feller [11, p. 233]), we get

$$Pr\left(\left|\frac{1}{L}\sum_{i=1}^{L}\zeta_i - Exp(\zeta_i)\right| \geq \rho\right) \leq \frac{Var(\frac{1}{L}\sum_{i=1}^{L}\zeta_i)}{\rho^2}$$

Let $\overline{\zeta_i} = \zeta_i - Exp(\zeta_i)$, then $Exp(\overline{\zeta_i}) = 0$. By pairwise independence $Exp(\overline{\zeta_i}\cdot\overline{\zeta_j}) = Exp(\overline{\zeta_i})\cdot Exp(\overline{\zeta_j})$. Hence,

$$\begin{aligned} Var\left(\frac{1}{L}\sum_{i=1}^{L}\zeta_i\right) &= \frac{1}{L^2}\sum_{i=1}^{L}\sum_{j=1}^{L} Exp\left(\overline{\zeta_i}\cdot\overline{\zeta_j}\right) \\ &= \frac{1}{L^2}\left(\sum_{i=1}^{L} Exp\left(\overline{\zeta_i}^2\right) + \sum_{1\leq i\neq j\leq L} Exp\left(\overline{\zeta_i}\right) Exp\left(\overline{\zeta_j}\right)\right) \\ &= \frac{1}{L^2}\cdot L\cdot Exp\left(\overline{\zeta_1}^2\right) = \frac{\rho(1-\rho)}{L} \end{aligned}$$

Thus, $Pr\left(\sum_{i=1}^{L}\zeta_i = 0\right) \leq \frac{1-\rho}{\rho L}$. The probability that $\sum_{i=1}^{L}\zeta_i = 0$ is exactly the probability that none of the r_i's is in S. □

Remark: The Theorem can be easily extended to deal with approximate counting, using a sequence of pairwise independent random sample points. Let ϵ be the desired degree of accuracy of the approximation, that is one would like to approximate the density of the subset (ρ) up to an *additive* error of ϵ. Then we get

$$Pr\left(\left|\frac{1}{L}\sum_{i=1}^{L}\zeta_i - Exp(\zeta_i)\right| \geq \epsilon\right) \leq \frac{\rho(1-\rho)}{\epsilon^2 L}.$$

4. Comparison with Independent Sampling

In order to compare our two-points based sampling technique to the "traditional" independent sampling, we consider two complexity measures: the number of random choices of elements in U, and the number of elements for which we test membership in S. The comparison is done with respect to the confidence parameter α, which is the probability of finding an element in the target set S. We consider only the case $\alpha > \rho$. (The case $\alpha \leq \rho$ is trivial.)

For the independent sampling the following holds: The number of independently chosen elements, k, equals the number of elements whose membership in S is tested. With k independent sampling, the probability of finding an element in S is $\alpha = 1-(1-\rho)^k$. Thus $k = \frac{\log 1-\alpha}{\log 1-\rho}$.*

* All logarithms are to base 2.

For the two-points based sampling the number of independently chosen elements is always two. Let L denote the number of elements whose membership in S is tested. The probability of finding an element in S is $\alpha = 1 - \frac{1-\rho}{\rho L}$. Thus $L = \frac{1-\rho}{\rho(1-\alpha)}$.

The effect of k independent samplings is achieved by a two-points based sampling on $L = \frac{1}{\rho(1-\rho)^{k-1}}$ elements. This is an exponential tradeoff between randomness (represented by k - the number of independent random choices) and deterministic computation (represented by L). Notice that the tradeoff does not depend on the desired degree of confidence (α), and is more favourable for small values of ρ (sparse sets S).

5. Comparison with an expander based technique

Karp and Pippenger [14] have previously suggested an alternative method for trading-off randomness and computation. Using an (explicitly constructed) expander, they cover the sample space U by $|U|$ subsets, each of size L. Every $\sigma|U|$ subsets cover at least $(1-\rho)|U|$ elements. Once this expander is constructed, k subsets are chosen at random, and all their members are tested. The probability that one of these $k \cdot L$ elements is in S, is at least $1 - \sigma^k$. Currently known expanders yield $\sigma = O\left(\frac{\log^2 L}{\rho L^\beta}\right)$, where $\beta = \frac{\log 3}{\log(1+2\sqrt{3}+\sqrt{2})} = 0.6202\ldots$

6. Extensions and Generalizations

So far we have shown how to generate a large sequence of pairwise independent elements in the field Z_p. This method can be extended to any finite field, and to rings which satisfy certain conditions. Another generalization will be generating sequences of k-wise independent elements, starting with k independently chosen elements.

In its most general form, our construction proceeds as follows. Let U be an arbitrary universe, and k a fixed integer. Let $\{f_i\}_{i=1}^{L}$ be a sequence of functions, $f_i : U^k \mapsto U$, such that the mapping

$$(x_1, \ldots, x_k)^T \mapsto (f_{i_1}(x_1, \ldots, x_k), \ldots, f_{i_k}(x_1, \ldots, x_k))^T$$

is a bijection of U^k onto itself (for all distinct $1 \leq i_1, \ldots, i_k \leq L$). Choosing the elements $x_1, \ldots, x_k \in U$ independently (with uniform probability distribution), the sequence

$$\{f_1(x_1, \ldots, x_k), \ldots, f_L(x_1, \ldots, x_k)\}$$

consists of L k-wise independent sampling elements, each uniformly distributed in U.

Using the generalized Chebyshev's inequality [11, p. 242], one can show that the probability of having at least one S element in this sequence exceeds $1 - \left(1 + \frac{1}{L}\right)\left(\frac{1-\rho}{\rho L}\right)^{\lfloor k/2 \rfloor}$ (for $L \geq \frac{1}{\rho}$). Notice that the performence guarantee of the L long sequence with k-wise independent elements is about the same as that of $k/2$ independent sequences, each consisting of L pairwise independent elements. However, k-wise independence may be useful for other purposes.

We conclude this section by presenting an implementation (suggested to us by Noga Alon [3]) of the general construction. (The same implemetation was discovered independently by Anderson [5] and Beame [6].) First note that U can be embedded in a finite field whose size is not significantly larger than $|U|$. Thus, without loss of generality, U is a finite field. Let $\{a_1, \ldots, a_L\}$ be a set of distinct elements in U. Define

$$f_i(x_1, \ldots, x_k) = \sum_{j=1}^{k} a_i^{j-1} x_j \qquad (\text{for } 1 \le i \le L) ,$$

where the arithmetic operations are in the field U. The desired properties of this family of functions follows from the non-singularity of the Vandermond matrix

$$\begin{pmatrix} 1 & a_{i_1} & \cdots & a_{i_1}^{k-1} \\ 1 & a_{i_2} & \cdots & a_{i_2}^{k-1} \\ \cdot & \cdot & \cdots & \cdot \\ \cdot & \cdot & \cdots & \cdot \\ 1 & a_{i_k} & \cdots & a_{i_k}^{k-1} \end{pmatrix}$$

In fact, this construction can be carried out if U is any commutative ring, provided that none of the $a_i - a_j$ $(1 \le i \ne j \le L)$ is a zero divisor.

7. Concluding Remarks

Our method can be viewed as "expanding randomness" for *sampling purposes*, without using any unproven assumptions. A much more general method for "expanding randomness" was presented by Blum and Micali [7] and Yao [22]. Under the assumption that 1-1 one-way functions exist it is possible to expand n truly random coin tosses into $poly(n)$ pseudo-random coin tosses that are good with respect to *any* polynomial-time algorithm.

Another possible perspective, is to view our construction as method for efficiently generating a long sequence of random k-wise independent events. This generation is efficient in the sense that only the first k elements in the sequence has to be randomly selected. The latter elements are deterministically computed given the first k. This property has already been demonstrated a useful tool in many applications (e.g. [1,4,5,15,17,20]). Especially inspiring is Luby's methodology of dispancing with randomness in special cases [17]. In Luby's setting the universe size is small but one needs many random elements in it. In case that pairwise independent random elements suffice, one can deterministically generate a small set of sequences such that using one of these sequences instead of random coin tosses yields the desired result. This technique is extendable to any fixed k (see [4,5]), but does not extend to the case that k grows with the instance size (see [9]).

An interesting property of our construction of pairwise independent integers modulo p, is that the sequence of bits obtained by taking the least significant bit of each of the integers is much less random then one may expect. Recall that $r_i \equiv x + iy \pmod p$ and let b_i denote the least significant bit of r_i, for $1 \le i \le L$. Clearly there cannot be more than $\min\{2^L, p^2\}$ possible b_i

sequences. (The 2^L upper bound follows from the length of the bit sequence, while the p^2 upper bound follows from the number of x, y pairs.) Suprisingly, as pointed out by A. Shamir, only L^3 bit sequences are possible. (The effect is best examplified by considering $L = 2\log p$. In this case there are only $O(\log^3 p)$ sequences, while the obvious upper bound is p^2.) Furthermore, as pointed out in [2], the sequence of b_i's can be predicted with high probability ($\geq 1 - L^{-\epsilon}$), when knowing $(1 + 2\epsilon) \cdot \log L$ bits of x, y. (One needs only to know the least significant bits of both x and y, the $\epsilon \cdot \log L$ most significant bits of x, and the $(1 + \epsilon) \cdot \log L$ most significant bits of y.) The fact that the b_i's can be predicted with very high probability plays a central role in [2].

Chronological Remark

The construction presented in section 3 was first discoved by us in April 1984 [10]. About half a year later, we found out that the construction (but not the application) had been presented by Joffe [12] in 1971 to a Probability Theory conference*. Constructions of pairwise (or k-wise) independent random variables have been used implicitly before in Computer Science works (e.g. [8,18,19,21,16]).

Acknowledgments

We would like to thank Noga Alon, Richard Karp, Nicholas Pippenger, and Adi Shamir for helpful discussions.

References

[1] Ajtai, M., and A. Wigderson, "Deterministic Simulation of Probabilistic Constant Depth Circuits", *Proc. of the 26th IEEE Symp. on Foundation of Computer Science*, (1985), pp. 11-19.

[2] Alexi,W., Chor,B., Goldreich,O., and Schnorr,C.P., "RSA and Rabin Functions: Certain Parts Are As Hard As The Whole", To appear in *SIAM Jour. on Computing.* Preliminary version in *Proc. of the 25th IEEE Symp. on Foundation of Computer Science*, (1984), pp. 449-457.

[3] Alon, N., private communication, 1985.

[4] Alon, N., Babai, L., and Itai, A., "A Fast and Simple Randomized Parallel Algorithm for the Maximal Independent Set Problem", to appear in *Jour. of Algorithms.*

[5] Anderson, R., "Set Splitting", manuscript, 1985.

[6] Beame, P., private communication, 1985.

[7] Blum,M., and Micali, S., "How to Generate Cryptographically Strong Sequences of Pseudo-Random Bits", *SIAM Jour. on Computing*, Vol. 13, No. 4, pp. 850-864, (Nov. 1984).

[8] Carter, J., and M. Wegman, "Universal Classes of Hash Functions", *Jour. of Comp. and Sys. Sc.*, Vol. 18, 1979, pp. 143-154.

* For a more extensive account of the events concerning restricted independence events, consult Luby [17]. We only note here that also the Vandermond construction (section 6) has been discovered before by Joffe [13].

[9] Chor, B., Friedman, G., Goldreich, O., Hastad, J., Rudich, S., and Smolanski, R., "The Bit Extraction Problem or t-Resilient Functions", *Proc. of the 26th IEEE Symp. on Foundation of Computer Science*, (1985), pp. 396-407.

[10]Chor, B., and Goldreich, O., "RSA/Rabin Least Significant Bits are $\frac{1}{2} + \frac{1}{poly(\log N)}$ Secure", MIT/LCS/TM-260, May 1984.

[11]Feller,W., *An Introduction to Probability Theory and its Applications*, John Wiley & Sons Inc., Vol. I, (third edition, 1968).

[12]Joffe, A., "On a Sequence of Almost Deterministic Pairwise Independent Random Variables", *Proc. Amer. Math. Soc.*, 1971, pp. 381-382.

[13]Joffe, A., "On a Set of Almost Deterministic k-Independent Random Variables", *Annals of Probability*, Vol. 2, No. 1, 1974, pp. 161-162.

[14]Karp,R.M, and Pippenger,N., "A Time-Randomness Tradeoff", presented at the *AMS conference on probabilistic computational complexity,* Durham, New Hampshire, (1982).

[15]Karp, R.M., E. Upfal and A. Wigderson, "The Complexity of Parallel Computation on Matroids", *Proc. of the 26th IEEE Symp. on Foundation of Computer Science*, (1985), pp. 541-550.

[16]Karp, R.M., A. Wigderson, "A Fast Parallel Algorithm for the Maximal Independent Set Problem", *Proc. of 16th ACM Symp. of Theory of Computing*, 1984, pp. 266-272.

[17]Luby, M., "A Simple Parallel Algorithm for the Maximal Independent Set Problem", *Proc. 17th ACM Symp. of Theory of Computing*, May 1985, pp. 1-10.

[18]Shamir, A., "How to Share a Secret", *Comm. of ACM*, Vol. 22, No. 11, Nov. 1979, pp. 612-513.

[19]Sipser, M., "A Complexity Theoretic Approach to Randomness", *Proc. of 15th ACM Symp. of Theory of Computing*, 1983, pp. 330-335.

[20]Spencer, T., "Provably Good Pattern Generators for Random Pattern Test", manuscript, 1985.

[21]Stockmeyer, L., "The Complexity of Approximate Counting", *Proc. of 15th ACM Symp. of Theory of Computing*, 1983, pp. 118-126.

[22]Yao,A.C., "Theory and Applications of Trapdoor Functions", *Proc. of the 23rd IEEE Symp. on Foundation of Computer Science*, pp. 80-91, (1982).