INFORMATION AND CONTROL 64, 43–51 (1985)

Pseudorandom Number Generation and Space Complexity*

MERRICK FURST

Department of Computer Science, Carnegie–Mellon University, Pittsburgh, Pennsylvania 15213

RICHARD LIPTON

Department of Electrical Engineering and Computer Science, Princeton University, Princeton, New Jersey 08544

AND

LARRY STOCKMEYER

IBM Research Lab, San Jose, California 95193

1. INTRODUCTION

Recently, Blum and Micali (1982) described a pseudorandum number generator that transforms m-bit seeds to m^k-bit pseudorandom numbers, for any integer k. Under the assumption that the discrete logarithm problem cannot be solved by polynomial-size combinational logic circuits, they show that the pseudorandom numbers generated are good in the sense that no polynomial-size circuit can determine the tth bit given the 1st through $(t-1)$th bits, with better than 50% accuracy. Yao (1982) has shown under the same assumption about the nonpolynomial complexity of the discrete logarithm problem, that these pseudorandom numbers can be used in place of truly random numbers by any polynomial-time probabilistic Turing machine. Thus, given a time n^k probabilistic Turing machine M and given any $\varepsilon > 0$, a deterministic Turing machine can simulate M by cycling through all seeds of length n^ε, giving a deterministic simulation in time 2^{n^ε}, an improvement over the time 2^{n^k} taken by the obvious simulation. Yao also shows that other problems, for example, integer factorization, can be used instead of the discrete logarithm in the intractability assumption.

* This paper is a revised and expanded version of a paper presented at the International Conference on "Foundations of Computation Theory" held in Borgholm, Sweden, August 21–27, 1983.

0019-9958/85 $3.00

The purpose of this paper is to observe that these intractability assumptions have implications toward space complexity. Two implications are that random time $T(n)$ is contained in deterministic space $T(n)/\log T(n)$, and checking whether a bipartite graph has a perfect matching can be done in space n^{ε}, for any $\varepsilon > 0$.

2. Definitions

We assume that the reader is familiar with time and space complexity for Turing machines (see, e.g., Aho, Hopcroft, and Ullman, 1974; Hopcroft and Ullman, 1979), combinational circuit complexity (see, e.g., Savage, 1976), and probabilistic Turing machines (see, e.g., Gill, 1977; Adelman, 1978). We consider only probabilistic Turing machines with one-sided error; that is, for rejected inputs the machine rejects with probability 1, and for accepted inputs the machine accepts with probability bounded away from zero.

Definition 1. Let $L \subseteq \{0, 1\}^*$ and let $\psi(x, y)$ be a predicate on two binary strings x and y. We say that ψ *defines* L *in random time* $T(n)$ if there is a constant $\eta > 0$ such that for all n and all x of length n:

(1) if $x \notin L$ then there is no y such that $\psi(x, y)$;

(2) if $x \in L$ then $|\{y \in \{0, 1\}^{T(n)} : \psi(x, y)\}|/2^{T(n)} \geqslant \eta$.

Furthermore, there is a deterministic Turing machine that, when given an x of length n and a y of length at least $T(n)$ on two separate input tapes, computes $\psi(x, y)$ within $T(n)$ steps.

Define RTIME($T(n)$) to be the class of languages $L \subseteq \{0, 1\}^*$ that are defined by some ψ in random time $T(n)$. Let R denote the union of RTIME($T(n)$) over all polynomials $T(n)$. Let DSPACE($S(n)$) (resp. DTIME($T(n)$)) denote the class of binary languages accepted by deterministic Turing machines in space $S(n)$ (resp. in time $T(n)$). Let P denote the union of DTIME($T(n)$) over all polynomials $T(n)$.

For our purposes we use the following definition of pseudorandom number generator which differs from the definitions in Blum and Micali (1982) and Yao (1982).

Definition 2. The function ρ is an $E(m)$-*expanding pseudorandom number generator* (*PNG*) if:

(1) $\rho : \{0, 1\}^m \to \{0, 1\}^{E(m)}$ for all m;

(2) ρ is computable by a deterministic Turing machine in space m and time polynomial in m;

(3) if ψ defines L in random time $T(n)$, then there is an n_0 such that if $x \in L$, $|x| \geq n_0$ and $E(m) \geq T(|x|)$, then there is an $s \in \{0, 1\}^m$ such that $\psi(x, \rho(s))$ is true.

We say that *polynomial-expanding PNGs exist* if an $E(m)$-expanding PNG exists for any polynomial $E(m)$.

Condition (3) means that to determine if $x \in L$, it is sufficient to check whether $\psi(x, \rho(s))$ is true for any $s \in \{0, 1\}^m$, where $E(m) \geq T(|x|)$. In condition (2) we assume that the Turing machine produces the output on a write-only output tape which s not bounded by m.

It is not known whether polynomial-expanding PNGs exist. However, under any of the following three assumptions, Yao (1982) proves that they do exist. We say that a function $F(m)$ is *superpolynomial* if for any polynomial $Q(m)$, $F(m) \geq Q(m)$, for almost all m.

ASSUMPTION A1. *Let p be an m-bit prime and let g be a generator of the multiplicative group $\{1, 2, \ldots, p-1\} \bmod p$. Let $L_{p,g}$ be the minimum size of a circuit that when given input y finds the x such that $g^x = y \pmod p$. Let $L(m)$ be the maximum of $L_{p,g}$ over all such p and g. Then $L(m)$ is superpolynomial.*

ASSUMPTION A2. *Let $F(m)$ be the minimum size of any circuit that can factor at least $\frac{4}{5}$ of the $2m$-bit composite numbers N with two m-bit prime factors. Then $F(m)$ is superpolynomial.*

ASSUMPTION A3. *There is a 1–1, onto function $f: \{0, 1\}^* \to \{0, 1\}^*$ such that f is computable in polynomial time and linear space, $|f(x)| = |x|$ for all x, and if $B_f(m)$ is the size of the smallest circuit which computes $f^{-1}(y)$ for at least $\frac{1}{2}$ of the y's of length m then $B_f(m)$ is superpolynomial.*

THEOREM 2.1 (Yao). *If Assumption* A1, A2 or A3 *holds, then polynomial-expanding PNGs exist.*

Proof. Assume A1, A2, or A3. Let $E(n) = n^k$, for some k, and let $\rho: \Sigma^n \to \Sigma^{E(n)}$ be a random number generator whose bits are hard to predict in the sense of Blum and Micali (1982), and Yao (1982). Such a random number generator has the property that without knowing the "seed" $s \in \Sigma^n$, no polynomial-size circuits can correctly predict the ith bit of $\rho(s)$ from the first $i-1$ bits with probability much greater than $\frac{1}{2}$. Such a ρ clarly satisfies parts (1) and (2) of the definition of PNGs. We show that it also satisfies part (3).

Let L be a set in random polynomial time. By definition, there exists a polynomial-time predicate $\psi(x, y)$, and a polynomial $E(n) = n^k$ such that

(1) $x \in L$ implies that at least half of the strings y of length $E(n)$ satisfy $\psi(x, y)$, and

(2) $x \notin L$ implies no y of length $E(n)$ satisfies $\psi(x, y)$.

For each x, let $n = |x|$, and let $W_x \subseteq \Sigma^{E(n)}$ be the set of "witnesses" to the fact that x is in L, i.e., let

$$W_x = \{ y \in \Sigma^{E(n)} \mid \psi(x, y) \}.$$

We show that, for all but a finite number of $x \in L$, some "random" string $y \in \rho(\Sigma^n)$ is a "witness" for x, i.e., $\rho(\Sigma^n) \cap W_x \neq \varnothing$.

We actually prove a stronger result, namely

$$\alpha_x = \left| \frac{|\rho(\Sigma^n) \cap W_x|}{|\Sigma^n|} - \frac{|W_x|}{|\Sigma^{E(n)}|} \right|$$

is not bounded below by $1/p(n)$, for any polynomial p. In other words, not only do the pseudorandom strings always include a "witness," but the proportion of pseudorandom strings that are "witnesses" is roughly the same as the proportion of truly random strings that are.

The proof is by contradiction. Assume that the difference in proportions, α_x is greater than $1/p(n)$. Let s be a "seed" in Σ^n. Consider the "random" string $\rho(s) = b_1 \cdots b_i$ to be the first i bits of $\rho(s)$. Define $f_i(s)$ to be the probability that a random completon of $\rho_i(s)$ is a "witness" for x, i.e.,

$$f_i(s) = \frac{|\{\rho_i(s) \cdot \Sigma^{E(n)-i}\} \cap W_x|}{2^{E(n)-i}}.$$

Define $r_{i+1}(s)$ to be the probability that a random completion of the $i+1$ bit string $b_1 \cdots b_i \bar{b}_{i+1}$ is a "witness" for x, i.e.,

$$r_{i+1}(s) = \frac{|\{\rho_i(s)\, \bar{b}_{i+1} \cdot \Sigma^{E(n)-i-1}\} \cap W_x|}{2^{E(n)-i-1}}.$$

Let f_i and r_i be the averages of $f_i(s)$ and $r_i(s)$ over all seeds. By definition,

$$\begin{aligned} f_0 &= \frac{1}{|\Sigma^n|} \cdot \sum_{s \in \Sigma^n} f_0(s) \\ &= \frac{1}{|\Sigma^n|} \cdot \sum_{s \in \Sigma^n} \frac{|\{\rho_0(s) \cdot \Sigma^{E(n)-0}\} \cap W_x|}{2^{E(n)-0}} \\ &= \frac{|\Sigma^{E(n)} \cap W_x|}{|\Sigma^{E(n)}|} \\ &= \frac{|W_x|}{|\Sigma^{E(n)}|} \end{aligned}$$

and

$$f_{E(n)} = \frac{1}{|\Sigma^n|} \cdot \sum_{s \in \Sigma^n} \frac{|\{\rho(s) \cdot \Sigma^0 \cap W_x|}{1}$$
$$= \frac{|\rho(E^n) \cap W_x|}{|\Sigma^n|}.$$

By assumption, $\alpha_x = |f_{E(n)} - f_0| > 1/\rho(n)$, therefore, for some $0 \leqslant i < E(n)$, $|f_{i+1} - f_i| > 1/p(n)\,E(n)$. Consider the following probabilistic, polynomial-time algorithm for predicting the $(i+1)$th bit of strings in $\rho(\Sigma^n)$. (Without loss of generality, assume that more pseudorandom strings are witnesses than are not witnesses; otherwise, reverse the outputs in what follows.) On input $b_1 b_2 \cdots b_i$, generate $E(n) - i$ random bits $t_{i+1}, \ldots, t_{E(n)}$. If the string $T = b_1 \cdots b_i t_{i+1} \cdots t_{E(n)}$ is a "witness," i.e., if $T \in W_x$, then output t_{i+1}, otherwise output $\neg t_{i+1}$.

The claim is that this algorithm works with probability bounded away from $\frac{1}{2}$. For a particular seed s, analyze the algorithm's behavior as follows. After the t_j have been generated at random there are four possible situations:

$$\{t_{i+1} \text{ is correct}, t_{i+1} \text{ is not correct}\} \times \{T \text{ is accepted}, T \text{ is rejected}\}.$$

(By t_{i+1} being correct we mean that $t_{i+1} = b_{i+1}$.) The probability that t_{i+1} is correct and T is accepted is exactly $f_{i+1}/2$. The probability that t_{i+1} is not correct and T is accepted is exactly $r_{i+1}/2$. Since half the time the random bit t_{i+1} is incorrect, the probability that t_{i+1} is incorrect and T is rejected is $\frac{1}{2} - r_{i+1}/2$.

Averaged over all seeds, the probability that T is accepted is $f_{i+1}/2 + r_{i+1}/2$, which is also f_i. Therefore, $1/p(n)\,E(n) < |f_i - f_{i+1}| = |f_{i+1}/2 - r_{i+1}/2|$.

Now, the probability that the algorithm correctly predicts the $(i+1)$th bit is

$$\Pr(t_{i+1} \text{ is correct and } T \text{ is accepted})$$
$$+ \Pr(t_{i+1} \text{ is incorrect and } T \text{ is rejected}),$$

which is $\frac{1}{2} + f_{i+1}/2 - r_{i+1}/2$. Therefore, our algorithm works with probability bounded away from $\frac{1}{2}$ by more than $1/p(n)\,E(n)$. This contradicts the "randomness" of ρ. Thus, $|\alpha_x| > 1/p(n)$ and, for sufficiently large x, $\rho(\Sigma^n)$ must contain a witness. ▮

The new observation here is that the pseudorandom strings can be computed in space m where m is the length of the seed. For Assumption A1

(resp. Assumption A2) the necessary computations are arithmetic operations modulo p (resp. modulo N) and only a fixed constant number of m-bit results need be remembered at any point during the computation. For Assumption A3, the computation requires computing several binary strings of the form $\alpha_i = f(s_i) f^2(s_i) f^3(s_i) \cdots$, where the s_i are taken from disjoint parts of the seed s. The output string is then the exclusive or the α_i. Thus, to compute a particular bit of the output, compute the appropriate bit of each α_i (which can be done in linear space since f is computable in linear space) and exclusive- or these bits as they are computed.

3. Results

Hopcroft, Paul, and Valiant (1977) have shown that a deterministic $T(n)$ time-bounded Turing machine can be simulated by a deterministic Turing machine in space $T(n)/\log T(n)$. Under any of the assumptions A1, A2, or A3 above, this result can be extended to random time bounded computations.

THEOREM 3.1. *If an $E(m)$-expanding PNG ρ exists for $E(m) \geq 2m \log m$, and if the function $T(n)/\log(T(n))$ is space constructible (Hopcroft and Ullman, 1979), then*

$$\mathrm{RTIME}(T(n)) \subseteq \mathrm{DSPACE}(T(n)/\log T(n)).$$

Proof. Let $L \in \mathrm{RTIME}(T(n))$ and let ψ define L in random time $T(n)$. Given an input x of length n, let the seed length $m = T(n)/\log T(n)$. For almost all n, $E(m) \geq T(n)$. Evaluate $\psi(x, p(s))$ for all $s \in \{0, 1\}^m$ and accept if the result is ever *true*. The computation of $\rho(s)$ can be done in space $m = T(n)/\log T(n)$ by condition (2) of Definition 2. Since ψ is computable in deterministic space $T(n)/\log T(n)$ by the result of Hopcroft, Paul, and Valiant (1977) mentioned above. By standard methods (see, e.g., Jones, 1975; or Stockmeyer and Meyer, 1973) if two functions are both computable in deterministic space $S(n) \geq \log n$ then their composition is computable in deterministic space $S(n)$.

Remark. Since Theorem 3.1 needs only a modest expansion of $O(m \log m)$ and since the predicate ψ is computable in time $O(m \log m)$, there is a fixed constant d such that the assumption of suerpolynomial complexity in A1, A2, or A3 can be relaxed to the assumption that $L(m)$, $F(m)$, or $B_f(m)$ is greater than m^d (Yao, 1984).

In a similar vein, a good space bound for any problem in P would imply a good space bound for any problem in R.

THEOREM 3.2. *If polynomial-expanding PNGs exist, then*

$$(\forall \varepsilon > 0)[P \subseteq \mathrm{DSPACE}(n^{\varepsilon})] \text{ iff } (\forall \varepsilon > 0)\ [R \subseteq \mathrm{DSPACE}(n^{\varepsilon})].$$

Proof. The "if" direction is immediate since $P \subseteq R$. For "only if", let $L \in \mathrm{RTIME}(n^k)$, and let $\varepsilon > 0$ be arbitrary. Choose an integer $b \geq k/\varepsilon$, and let ρ be an m^b-expanding PNG. To accept L for inputs of length n, let $m = n^{\varepsilon}$ and proceed as in the proof of Theorem 3.1. The computation of ρ can be done in space $m = n^{\varepsilon}$, and since $\psi \in P$ the computation of ψ can be done in space n^{ε} by assumption. (Even though the input (x, y) to ψ has length $n + n^k$, we can choose a constant $\delta > 0$ so small hat $(n + n^k)^{\delta} \leq n^{\varepsilon}$ for almost all n.)

THEOREM 3.3. *Let ψ define L in random polynomial time where ψ is computable in deterministic space n^{ε} for any $\varepsilon > 0$. If polynomial-expanding PNGs exist, then*

$$L \in \mathrm{DSPACE}(n^{\varepsilon}) \qquad \textit{for any } \varepsilon > 0.$$

Proof. The proof is similar to the two preceding proofs. Both ρ and ψ can be computed in space n^{ε}.

Remark. In Theorem 3.3, the algorithm that computes ψ in polynomial time and the algorithm that computes ψ in space n^{ε} do not have to be the same algorithm.

We now give an application of Theorem 3.3. Let **PER** denote the set of square adjacency matrices of bipartite graphs that have a perfect matching. Equivalently, **PER** is the set of square 0–1 matrices with nonzero permannt.

COROLLARY 3.4. *If polynomial-expanding PNGs exist, then*

$$\mathbf{PER} \in \mathrm{DSPACE}(n^{\varepsilon}) \qquad \textit{for any } \varepsilon > 0.$$

Proof. The following probabilistic algorithm for accepting **PER** is due to Lovasz. Let A be the given $n \times n$ 0–1 matrix. Replace each 1 in A by an integer chosen from the uniform distribution on $\{1, 2, \ldots, n\}$; let A' be the resulting matrix. Compute the determinant of A' and accept iff $\det(A') \neq 0$. If $\mathrm{perm}(A) = 0$ then $\det(A)' = 0$. For some constant $\eta > 0$, if $\mathrm{perm}(A) \neq 0$, then $\det(A') \neq 0$ with probability $> \eta$. Thus, the predicate $\psi(A, y)$ is a determinant computation where the string y specifies the integers to be substituted for the 1's in A. It is implicit in Czanky (1976) that the determinant computation can be done in space $(\log n)^d$ for some constant d which is $o(n^{\varepsilon})$ for any $\varepsilon > 0$.

Remark. Corollary 3.4 suggest an interesting distinction between problems defined by predicates $\psi(x, y)$, where the string y is read by a two-way head and those where y is read by a one-way head. For time-bonded probabilistic computation there is clearly no difference, but for space-bounded probabilistic computation where the space bound is much less than the length of y, it could matter. The above algorithm for **PER** needs a two-way head on y since the polylog-space algorithm for the determinant needs to read the entries of the matrix many times. Of course, this is no problem for our deterministic space n^ε algorithm. Since the seed s is short enough to be stored, the bits of $\rho(s)$ can be recomputed whenever they are needed in the computation of $\psi(A, \rho(s))$.

Let RTISP$(T(n), S(n))$ be the class of languages accepted by prohabilistic Turing machines that are simultaneously $T(n)$ time-bounded and $S(n)$ space-bounded. By the proof of Savitch's theorem (Savitch, 1970), it is known that RTISP$(T(n), S(n)) \subseteq$ DSPACE$(S(n) \log T(n))$. The proof of the following is similar to proofs above.

THEOREM 3.5. *If $S(n) \geqslant (T(n))^\varepsilon$, for some $\varepsilon > 0$, if $S(n)$ is space constructible, and if polynomial-expanding PNGs exist, then*

$$\text{RTISP}(T(n), S(n)) \subseteq \text{DSPACE}(S(n)).$$

ACKNOWLEDGMENT

We thank Andy Yao, Tom Leighton, and Silvio Micali for very helpful discussions about the results of Blum and Micali (1982) and Yao (1982).

REFERENCES

ADLEMAN, L. (1978), Two theorems on random polynomial time, *in* "Proc. 19th IEEE Sympos. on Foundations of Computer Science," pp. 75–83.

AHO, A. V., HOPCROFT, J. E., AND ULLMAN, J. D. (1974), "The Design and Analysis of Computer Algorithms," Addison–Wesley, Reading, Mass.

BLUM, M., AND MICALI, S. (1982), How to generate cryptographically strong sequences of pseudo random bits, *in* "Proc. 23rd IEEE Sympos. on Foundations of Computer Science," pp. 112–117.

CSANKY, L. (1976), Fast parallel matrix inversion algorithms, *SIAM J. Comput.* **5**, 618–623.

GILL, J. (1977), Computational complexity of probabilistic Turing machines, *SIAM J. Comput.* **6**, 675–695.

HOPCROFT, J., PAUL, W., AND VALIANT, L. (1977), On time versus space, *J. Assoc. Comput. Mach.* **24**, 332–337.

HOPCROFT, J. E., AND ULLMAN, J. D. (1979), "Introduction to Automata Theory, Languages and Computation," Addison–Wesley, Reading, Mass.

JONES, N. (1975), Space-bounded reducibility among combinatorial problems, *J. Comput. System Sci.* **11**, 68–85.

SAVAGE, J. E. (1976), "The Complexity of Computing," Wiley, New York.

SAVITCH, W. J. (1970), Relationships between nondeterministic and deterministic tape complexities, *J. Comput. System Sci.* **4**, 177–192.

STOCKMEYER, L. J., AND MEYER, A. R. (1973), Word problems requiring exponential time, *in* "Proc. 5th ACM Sympos. on Theory of Computing," pp. 1–9.

YAO, A. (1982), Theory and applications of trapdoor functions, *in* "Proc. 23rd IEEE Sympos. on Foundations of Computer Science,' pp. 80–91.

YAO, A. (1984), personal communication.