THEORETICAL ASPECTS OF THE SECURITY OF PUBLIC KEY CRYPTOGRAPHY

By

Evangelos Kranakis[1]

Department of Computer Science

Yale University

New Haven CT, 06520

Technical Report [illegible], September 1984
331

[1] The preparation of this monograph was supported in part by the National Science Foundation under grant MCS-8305382.

PROLOGUE

The need for secure transmission of information among many users through the microelectronic media, has made inevitable the departure of cryptography from the old notion of absolute security to embrace the new notion of relative security. Thus, in the first case the designer bases the security of the cryptosystem on absolute criteria (e.g. Shannon's Information Theory), while in the second case he proves that the system he designed is secure assuming that a certain problem (usually in Number Theory) is difficult to solve. This new idea has made possible the construction of cryptosystems, called public key cryptosystems.

The purpose of the present monograph, which is an outgrowth of twelve lectures at Yale University Computer Science Department in the Fall of 1984, is to isolate and explain the most important mathematical notions which arise from the recent literature on public key cryptosystems. And in doing this, I have made every posssible effort within the bounds of reason to make this monograph as self-contained as possible.

The work is divided into three main parts. The first part consisting of two sections develops the necessary number theory and probability theory (sections 1 and 2 respectively.) The second part consists of the discussion of certain pseudo random generators (section 3) and public key cryptosystems (section 4); their development tends to emphasize those concepts which can be generalized to develop a general theory. Finally, the third part (section 5) develops the general theory of pseudo random generators and public key cryptosystems.

The reader should be aware of the many different viewpoints given in the papers cited in the bibliography, not all of which could naturally be included in the present study. The exercises given at the end of most subsections are of three types: those that give a different proof of a result proved in the main text, those that give additional results, and those which remind the reader that he must complete the details of the proof of a result given in the text. In any case, none of them is difficult and they should all be attempted by the reader.

I have made every possible effort to attribute the theorems presented in the text to their original inventor. If sometimes I failed to do that it is due to ignorance rather than intent. At the same time I accept full responsibility for whatever flaws or errors the monograph may contain. and I would be greateful to receive any comments and suggestions that will improve the presentation.

In addition, I am particularly thankful to the insightful comments of the seminar participants during the above mentioned lectures. These included: Dana Angluin, Mike Fischer, Dan Gusfield, Neil Immerman, Susan Landau, and the students: Josh Cohen, Ming Kao, Phillip Laird, Jerry Leichter, Lenny Pitt and David Wittenberg. I would also like to express my deepest appreciation to Mike Fischer for his undiminishing support and encourangement as well as for the the numerous penetrating discussions that helped me improve the presentation of section 5.

FREQUENTLY USED NOTATION

- $|A|$; the cardinal of the set A.
- •; end of proof symbol.
- $\emptyset$; the empty set.
- $A \cup B, A \cap B, A - B$; the union, intersection and difference of the sets A, B.
- $f : A \longrightarrow B$; a mapping of a set A into a set B.
- $x \longrightarrow y$; the mapping carries the point x to the point y.
- $\exists, \forall, \Rightarrow, \Leftrightarrow$; there exists, for all, implies, if and only if.
- $x \equiv y \mod n$; x congruent to y modulo n.
- $(x|y)$; the Jacobi symbol of x with respect to y.
- $Z_n^* = \{x < n : \gcd(x, n) = 1\}$.
- $Z_n^*(+1) = \{x \in Z_n^* : (x|n) = +1\}$.
- $Z_n^*(-1) = \{x \in Z_n^* : (x|n) = -1\}$.
- $\varphi(n) = |Z_n^*|$; the Euler function.
- λ; the Carmichael function.
- $\pi(n)$; the number of primes $\leq n$.
- $\text{index}_{p,g}(n)$; the index of x with respect to $g \in Z_p^*$.
- $\lceil x \rceil, \lfloor x \rfloor, [x]$; ceiling of x, floor of x, integral part of x.
- $Pr[E]$; probability of the event E.
- $Pr_A[E] = Pr[E|A]$; conditional probability of the event E with respect to the event A.
- $E[X], Var[X], D[X]$; expectation, variance, divergence of the random variable X.
- $B_n(E)$; the number of occurrences of the event E in n indepedent trials.
- $F_n(E) = B_n(E)/n$.
- RSA; the Rivest, Shamir, Adleman cryptosystem.
- QRA; the Quadratic Residuosity Assumption.
- DLA; the Discrete Logarithm Assumption.

1 NUMBER THEORY

1.1 INTRODUCTION

The purpose of the present section is twofold: on the one hand, to introduce the reader to the basic concepts of number theory, and on the other hand, to provide proofs of some efficient procedures arising in the development of algorithms for the solution of some number theory problems, both of which will be essential to the discussion of pseudo-random generators and public-key cryptosystems.

The concepts introduced in this section include: Fibonacci Numbers, the Euler function, primitive roots, the Carmichael function, Langrange-Jacobi symbol, indices and continued fractions. In addition, complete proofs of the following theorems are given: Gauss theorem on the characterization of those m for which the multiplicative group Z_m^* is cyclic, in theorem 1.9, the Law of Quadratic Reciprocity, in theorem 1.13, Chebyshev's proof of a weaker version of the prime number theorem, in theorem 1.20, and a theorem on Diophantine approximations, in theorem 1.23. Theorem 1.7 provides an application of the Chinese Remainder Theorem to threshold schemes.

The algorithms described include: the method of exponentiation by repeated squarings and multiplications, in theorem 1.8, the method of Adelman, Manders and Miller for computing square roots modulo a prime, in theorem 1.15, and the method of Pohlig and Hellman for computing indices, in theorem 1.18.

It is true, that the details of the proofs of some of the theorems presented in this section (e.g. theorems 1.9, 1.13 and 1.20) are not necessary for understanding the concepts included in the sections of pseudo-random generators and public-key cryptosystems. However, a thorough study of the proofs and the exercises that follow the individual subsections will undoubtedly enhance the reader's proficiency with the number theory concepts involved.

1.2 THE HOMOMORPHISM THEOREM

Let G, H be two abelian groups such that H is a subgroup of G. For $a \in G$ consider the coset $H + a = \{h + a : h \in H\}$, where $+$ is the group operation on G. G/H is the **quotient** group of G modulo H. It consists of all cosets $H + a$, where a ranges over G. The group operation $\oplus$ on G/H is defined by $(H + a) \oplus (H + b) = H + (a + b)$. It is not hard to show that G/H with this operation is also an abelian group. It is clear that the family $\{H + a : a \in G\}$ of cosets is a partition of G into sets each of which has size exactly $|H|$. It follows that $|H|$ divides $|G|$.

Let f be an epimorphism from G onto another group H. The **kernel** $K = Ker(f)$ of f, is the set of all elements $a \in G$ such that $f(a) =$ the identity element of H. It is not hard to show that the group G/K is an abelian group which is isomorphic to H. The required isomorphism is given by the mapping

$F(K+a) = f(a)$. Hence the proof of the following theorem has been outlined:

Theorem 1.1 *(***Lagrange***)*

(i) If H is a subgroup of G then $|H|$ divides $|G|$.

(ii) If f is an epimorphism of the abelian group G onto the abelian group H and K is the kernel of f then the group G/K is isomorphic to the group H. Moreover, $|G| = |H| \cdot |K|$, and for all $a \in G$, $|f^{-1}\{a\}| = |K|$. •

EXERCISES

1: Let G be a finite abelian group. Show that all equations of the form $x^2 = a$, where $a \in G$, have exactly the same number of solutions in G. **Hint:** Consider the abelian group $H = \{a^2 : a \in G\}$ and let f be the epimorphism $f(x) = x^2$. Then use theorem 1.1.

2: Extend exercise 1 to equations of the form $x^n = a$, where $a \in G, n \geq 1$.

3: Show that the definition of the operation $\oplus$ is independent of the coset representation.

In the next two exercises H is a subgroup of the abelian group G. Complete the details of the proof of theorem 1.1 by showing that:

4: for all $a \in G$, $|H + a| = |H|$.

5: $\{H + a : a \in G\}$ forms a partition of G.

1.3 FIBONACCI NUMBERS

The sequence $f_0, f_1, \ldots, f_n, \ldots$ of **Fibonacci numbers** is defined by induction on $n \geq 0$ as follows:

$$f_n = \begin{cases} 0 & \text{if } n = 0 \\ 1 & \text{if } n = 1 \\ f_{n-1} + f_{n-2} & \text{if } n \geq 2 \end{cases}$$

It will be useful to know the order of magnitude of the n-th Fibonacci number. This is easily determined as follows. The quadratic equation $x^2 = x + 1$ has the two square roots $(1+\sqrt{5})/2$ and $(1-\sqrt{5})/2$. The positive square root $(1+\sqrt{5})/2$ is called the **golden ratio**, and is abbreviated with R. It is now easy to check by induction on n, that for all $n > 1, f_n \geq R^{n-2}$. Indeed, assume that $f_n \geq R^{n-2}$. Then, $f_{n+1} = f_n + f_{n-1} \geq R^{n-2} + R^{n-3} = R^{n-3}(R+1) = R^{n-3}R^2 = R^{n-1}$.

The Fibonacci numbers arise very naturally in the study of the number of steps needed to evaluate the greatest common divisor of two integers. Indeed, assume that $a > b > 0$ are two given integers. Use the Euclidean algorithm to define sequences $0 < r_n < r_{n-1} < \ldots < r_1 < r_0 = b < r_{-1} = a, d_1, d_2, \ldots, d_n, d_{n+1}$ such that $r_{i-2} = d_i r_{i-1} + r_i, i = 1, \ldots, n$ and $r_{n-1} = d_{n+1} r_n$. It is clear that $r_n = \gcd(a, b)$ (see exercise 1 below.) It follows by reverse induction on $i = n, n-1, \ldots, 0, -1$ that $r_i \geq f_{n+1-i}$. In particular,

$a \geq f_{n+2}$ and $b \geq f_{n+1}$. However, it is clear that the number of division steps needed to compute $\gcd(f_{n+2}, f_{n+1})$ is exactly $n+2$, which is also the number of division steps needed to compute $\gcd(a,b)$. Since, $a \geq f_{n+2} \geq R^n$, it follows that $\log_R a \geq n$. Therefore the following theorem has been proved.

Theorem 1.2 *(G. Lamé) If N is an integer > 0, then for any pair a, b of positive integers $\leq N$, the number of division steps required to compute* $\gcd(a,b)$ *is at most* $-2 + \lfloor \log_R N \rfloor$. •

EXERCISES

1: Show that $r_n = \gcd(a,b)$.

2: Show that the Euclidean algorithm leads to an efficient algorithm which given any integers a, b will compute integers α, β such that $\gcd(a,b) = \alpha a + \beta b$. Generalize this to the greatest common divisor of n integers.

3: Prove a similar theorem for the greatest common divisor of n integers.

4: Prove a similar theorem for the least common multiple of n integers. **Hint:** Use the identity $\mathrm{lcm}(a_1, \ldots, a_n) = (a_1 \cdots a_n)/\gcd(a_1, \ldots, a_n)$.

5: Show that the length of the side of the canonical decagon inscribed in the unit circle is equal to R, where R is the golden mean.

1.4 CONGRUENCES

Let a, b be integers. The symbol $a|b$ means that a divides b i.e. $b = ka$, for some integer k. The integers a, b are called **congruent** modulo the integer m, and this will be abbreviated $a \equiv b \bmod m$, if $m|(a-b)$, otherwise a and b will be called **incongruent** modulo m, and this will be abbreviated by $a \not\equiv b \bmod m$. It is clear that for each fixed m, the relation $\equiv \bmod m$ is **reflexive, symmetric,** and **transitive**, and hence it is an **equivalence** relation on the set Z of all integers. For each integer a let $\mathbf{a}$ denote the equivalence class of a i.e. the set of all integers x such that $x \equiv a \bmod m$. For each m there exist exactly m equivalence classes modulo m, namely $\mathbf{0}, \mathbf{1}, \cdots, \mathbf{m-1}$. $Z_m = \{\mathbf{0}, \mathbf{1}, \cdots, \mathbf{m-1}\}$ is the set of all equivalence classes modulo m, and $Z_m^* = \{\mathbf{a} : \gcd(a, m) = 1\}$.

One can define two operations, addition $(+)$ and multiplication $(\cdot)$ on the set Z_m as follows: $\mathbf{a} + \mathbf{b}$ (respectively $\mathbf{a} \cdot \mathbf{b}$) $=$ the equivalence class of $a+b$ (respectively $a \cdot b$). The set Z_m endowed with these two operations forms a commutative ring with unit. In fact both $< Z_m, + >$ and $< Z_m^*, \cdot >$ are abelian groups.

Example 1.1 *Figure 1 gives the multiplication table of Z_{11}^*.*

If $\gcd(a, m) = 1$ then there exist integers b, c such that $ab + cm = 1$. Hence, $\mathbf{a} \cdot \mathbf{b} = \mathbf{1}$ i.e. $\mathbf{a}$ is invertible in Z_m^*. The order of the group Z_m^* is denoted by $\varphi(m)$, and φ is called the **Euler totient function.**

.	1	2	3	4	5	6	7	8	9	10
1	1	2	3	4	5	6	7	8	9	10
2	2	4	6	8	10	1	3	5	7	9
3	3	6	9	1	4	7	10	2	5	8
4	4	8	1	5	9	2	6	10	3	7
5	5	10	4	9	3	8	2	7	1	6
6	6	1	7	2	8	3	9	4	10	5
7	7	3	10	6	2	9	5	1	8	4
8	8	5	2	10	7	4	1	9	6	3
9	9	7	5	3	1	10	8	6	4	2
10	10	9	8	7	6	5	4	3	2	1

Figure 1: Multiplication table of Z_{11}^*

An important corollary of the above considerations is the following

Theorem 1.3 (**Euler-Fermat**)*For all* $a \in Z_m^*, a^{\varphi(m)} \equiv 1$ mod$m$.

Proof: Let a be as above, and let $u_1, \dots, u_{\varphi(m)}$ be an enumeration of all the elements of Z_m^*. It is clear that $a \cdot u_1, \dots, a \cdot u_{\varphi(m)}$ is also an enumeration of all the elements of Z_m^*. Consequently, $a \cdot u_1 \cdots a \cdot u_{\varphi(m)} = u_1 \cdots u_{\varphi(m)}$ and hence $a^{\varphi(m)} \cdot u_1 \cdots u_{\varphi(m)} = u_1 \cdots u_{\varphi(m)}$. But it follows from the above observations that the element $u_1 \cdots u_{\varphi(m)}$ is invertible in Z_m. Consequently, $a^{\varphi(m)} \equiv 1$ mod$m$.•

In order to avoid unnecessary notational complications, from now on the same symbol will be used for an integer a and its equivalence class a modulo a certain integer m. This will cause no confusion because it will always be clear from the context which of the two notions is meant.

Theorem 1.4 (**Euler**) $\sum_{d|m} \varphi(d) = m$.

Proof: Let $\varphi_d(m)$ = the number of integers $x \in Z_m$ such that $\gcd(x, m) = d$. It is then clear that $\sum_{d|m} \varphi_d(m) = m$. However, $\varphi_d(m) = \varphi(m/d)$, provided that d divides m. It follows that

$$m = \sum_{d|m} \varphi_d(m) = \sum_{d|m} \varphi(d/m) = \sum_{d|m} \varphi(d),$$

which completes the proof of the theorem. •

Any equation of the form $f(x) \equiv 0$ mod$m$, where $f(x)$ is a polynomial expression in the variable x with coefficients in Z_m is called **congruence modulo** m. Such a congruence is called **solvable** if there is an $x \in Z_m$ such that

$f(x) \equiv 0 \bmod m$; the set of all x's in Z_m which satisfy this congruence is called the set of **solutions** of that congruence. One of the most important questions in Number Theory is to develop methods to solve equations of the form $f(x) \equiv 0 \bmod m$, where $f(x)$ is a polynomial expression in the variable x with coefficients in Z_m. For linear congruences with one unknown this question is answered in the theorem below:

Theorem 1.5 (**Solving Linear Congruences**) *The linear congruence $ax \equiv b \bmod m$ is solvable if and only if $g = \gcd(a, m)$ divides b. In fact if x_0 is any solution of $ax \equiv b \bmod m$ then the list*

$$x_i = x_0 + \frac{im}{g}, \text{ where } i = 0, \dots, g-1,$$

forms the complete set of its distinct solutions modulo m.

Proof: If the congruence $ax \equiv b \bmod m$ is solvable then m must divide $ax - b$. Since $g|m$ and $g|a$ it is clear that $g|b$. Conversely, assume that $g|b$. It follows that $b = kg$, for some integer k. It is well known however, using basic properties of the greatest common divisor, that there exist integers λ, μ such that

$$g = \lambda a + \mu m.$$

It follows that

$$b = kg = k\lambda a + k\mu m = (k\lambda)a + (k\mu)m,$$

and hence $k\lambda$ is a solution of the congruence $ax \equiv b \bmod m$.

It is not hard to see that if x_0 is any solution of the above congruence so is any of the x_i's defined above, moreover the solutions x_i are distinct modulo m. It remains to show that any arbitrary solution c of $ax \equiv b \bmod m$, is equal to some x_i. Indeed, since $ac \equiv ax_0 \equiv b \bmod m$, it follows that $m|a(c - x_0)$. But $g = \gcd(a, m)$, and hence $(m/g)|(c - x_0)$, which completes the proof of the theorem.•

EXERCISES

1: Show that for all $n > 1$, n is prime $\Leftrightarrow \varphi(n) = n - 1$.

2: Show that for all $t \geq 1$ and all prime p, $\varphi(p^t) = (p-1)p^{t-1}$. Use this to compute $\varphi(n)$ for all integers $n > 0$.

1.5 THE CHINESE REMAINDER THEOREM

Systems of linear congruences may not necessarily have solutions although each of the congruences of the system do.

Example 1.2 *Both congruences:*

$$x \equiv 0 \bmod 3, \quad and \quad x \equiv 1 \bmod 6$$

have solutions, but the system does not.

However, if the moduli are pairwise relatively prime, the system has a solution as this is shown in the theorem below.

Theorem 1.6 *(***Chinese Remainder Theorem***)*
The system $a_i x \equiv b_i \bmod m_i$, $i = 1, \ldots, k$ has exactly one solution modulo $m = m_1 \cdots m_k$, provided that $m_1, \ldots, m_k$ are relatively prime and $\gcd(a_1, m_1) = \cdots = gcd(a_k, m_k) = 1$.

Proof: The uniqueness of the solution follows easily from the fact that the integers $m_1, \ldots, m_k$ are relatively prime. Next, find integers c_i such that $a_i c_i \equiv 1 \bmod m_i$, where $i = 1, \ldots, k$. If $m = m_1 \cdots m_k$ and $n_i = m/m_i$ then it is clear that $\gcd(n_1, \ldots, n_k) = 1$. Hence, it follows from the basic properties of the greatest common divisor, there exist integers $t_1, \ldots, t_k$ such that

$$t_1 n_1 + \cdots + t_k n_k = 1.$$

Put $e_i = t_i n_i$. Then one can easily verify that

$$e_i \equiv \delta_{i,j} \bmod m_j,$$

where $\delta_{i,j} = 1$ if $i = j$ and $\delta_{i,j} = 0$ if $i \neq j$. Now, choose $c = e_1 c_1 b_1 + \cdots + e_k c_k b_k$. It remains to show that c is a solution of the above system of congruences. Indeed, for each i,

$$a_i c \equiv a_i b_1 e_1 c_1 + \cdots + a_i b_k e_k c_k \equiv a_i b_i e_i c_i \equiv b_i \bmod m_i,$$

and the proof is complete •

An interesting application of the Chinese remainder theorem, which is also relative to the security of message transmission, is to the construction of (k, n) threshold schemes. A (k, n) **threshold scheme** consists of n people $P_1, \ldots, P_n$ sharing a secret S in such a way that the following properties hold

1. Each P_i has some information I_i.
2. Knowledge of any k of the $\{I_1, \ldots, I_n\}$ enables one to find S easily.
3. Knowledge of less than k of the $\{I_1, \ldots, I_n\}$ does not enable one to find S easily.

Theorem 1.7 *For all $2 \leq k \leq n$ there exists a (k, n) threshold scheme.*

Proof: (Mignotte) The construction of (k, n) threshold schemes is based on the construction of (k, n) threshold sequences. A (k, n) **threshold sequence** is an increasing sequence $m_1 < \cdots < m_n$ of relatively prime positive integers such that

$$m_1 \cdot m_2 \cdots m_k > m_n \cdot m_{n-1} \cdots m_{n-k+2}. \tag{1}$$

Assume that a threshold sequence $m_1 < \cdots < m_n$ has been constructed and let $M = m_1 \cdot m_2 \cdots m_k$, $N = m_n \cdot m_{n-1} \cdots m_{n-k+2}$. Let the secret S be any integer such that $N \leq S \leq M$ and let the information I_i be defined by

$$I_i \equiv S \bmod m_i,\ i = 1, \ldots, n.$$

It will be shown that the above defined secret S and informations $\{I_1, \ldots, I_n\}$ form a (k, n) threshold scheme. Indeed, let $\{I_{i_1}, \ldots, I_{i_k}\}$ be given. By the Chinese remainder theorem the system

$$x \equiv I_i \bmod m_i,\ i \in \{i_1, \ldots, i_k\},$$

has exactly one solution. The proof of theorem 1.6 shows that this solution is S and is given by

$$S \equiv e_{i_1} \cdot I_{i_1} + \cdots + e_{i_k} \cdot I_{i_k} \bmod (m_{i_1} \cdots m_{i_k}),$$

where $e_i \equiv \delta_{i,j} \bmod m_j$. It follows from (1) that in fact

$$S = e_{i_1} \cdot I_{i_1} + \cdots + e_{i_k} \cdot I_{i_k}$$

On the other hand if only $\{I_{i_1}, \ldots, I_{i_{k-1}}\}$ are given, then it follows again from the Chinese remainder theorem that

$$S \equiv e_{i_1} \cdot I_{i_1} + \cdots + e_{i_{k-1}} \cdot I_{i_{k-1}} \bmod (m_{i_1} \cdots m_{i_{k-1}}). \tag{2}$$

Clearly, (2) is the only congruence available in order to compute the value of S. It follows that there are at least $\frac{M-N}{N}$ values for S satisfying (2). To conclude the proof of the theorem it remains to construct (k, n) threshold sequences such that the quantity $\frac{M-N}{N}$ is big. This will make it difficult to compute S if less than k of the $\{I_1, \ldots, I_n\}$ are known. This is done using inequality (21) in exercise 3 of subsection 1.15. Indeed, find t such that inequality (21) holds. It follows that there exist at least n distinct primes in the interval $(p_t^{(k^2-1)/k^2}, p_t]$. Let $m_1, \ldots, m_n$ be the last n primes in the last interval i.e. $m_i = p_{t-n+i}$, where $i = 1, \ldots, n$. It remains to show that this is a (k, n) threshold sequence. Indeed,

$$M = m_1 \cdot m_2 \cdots m_k \geq p_t^{\frac{k^2-1}{k}} > p_t^{k-1} \geq m_n \cdot m_{n-1} \cdots m_{n-k+2} = N.$$

This completes the proof of the theorem •

EXERCISES

1: If r is the number of prime factors of $m > 1$ then $x^2 \equiv x \bmod m$ has exactly 2^r distinct modulo m solutions. **Hint:** Use the Chinese Remainder Theorem.

1.6 MODULAR EXPONENTIATION

Given a fixed modulus m and an exponent e, the problem arises to compute x^e mod$m$, for any given x. The method to be described below, which solves this problem is called the method of **exponentiation by repeated squarings and multiplications.**

Theorem 1.8 *There is an efficient algorithm A such that given as inputs m, e, x it will output $A(m, e, x) = x^e$ mod$m$. The algorithm A requires at most $\lfloor \log_2 e \rfloor$ squarings, $2\lfloor \log_2 e \rfloor$ multiplications and $2\lfloor \log_2 e \rfloor$ divisions.*

Proof: Let e, m and x be integers as above. Consider e's representation in the binary system i.e. $e = 2^n e_n + 2^{n-1} e_{n-1} + \ldots + 2e_1 + e_0$, where $n = \lfloor \log_2 e \rfloor$. Then $x^e \equiv x^{2^n e_n + \ldots + 2e_1 + e_0}$ mod$m$. Define the sequences $x_0, \ldots, x_n$ and $y_1, \ldots, y_n$ by reverse induction as follows: $x_n = x^{e_n}, y_n \equiv x_n^2$ mod$m$ and $x_{n-i} \equiv y_{n-i+1} x^{e_{n-i}}$ mod$m, y_{n-i} \equiv x_{n-i}^2$ mod$m$. It follows easily by reverse induction that $x_0 \equiv x^e$ mod$m$.

The above recursive construction is also exhibited in the algorithm below:

```
Input: $e, m, x$
Step 1: Compute $n$, and bits $e_0, e_1, \ldots, e_n$ such that
            $e = 2^n e_n + 2^{n-1} e_{n-1} + \ldots + 2^1 e_1 + e_0$, where $e_n \neq 0$.
Step 2: Set $y := 1$.
Step 3: For $i = n, n-1, \ldots, 0$ repeat
                set : $y \equiv y^2 x^{e_i}$ mod$m$.
Output: $y$.•
```

Example 1.3 *Using the table in Example 1.1 and the above algorithm, the table in figure 2 shows that* $7^{13} \equiv 3$ mod11.

i	e_i	$y \equiv y^2 \cdot 7^{e_i}$ mod11	*Output*
3	1	$1^2 \cdot 7^{e_3}$	7
2	1	$7^2 \cdot 7^{e_2}$	2
1	0	$2^2 \cdot 7^{e_1}$	6
0	1	$6^2 \cdot 7^{e_0}$	3

Figure 2: Computation of 7^{13} mod11.

EXERCISES

1: Find a similar algorithm for modular multiplication.

1.7 PRIMITIVE ROOTS

Call an integer $g \in Z_m^*$ a **primitive root** modulo m if g generates the multiplicative group Z_m^* i.e. $Z_m^* = \{g, g^2 \bmod m, \ldots, g^{\varphi(m)} \bmod m\}$. If there is a primitive root modulo m then the group Z_m^* is cyclic, and vice versa. In the sequel, it will be useful to know for which m is the group Z_m^* cyclic. The following theorem gives the complete answer.

Theorem 1.9 (Gauss) *For all m, Z_m^* is cyclic if and only if m is equal to one of $1, 2, 4, p^k, 2p^k$, where p is an odd prime and k is a positive integer.*

Proof: $(\Leftarrow)$

If m is equal to either of 1, 2, 4 it is easy to see that Z_m^* is cyclic (see exercise 2 at the end of this subsection.) Next it will be shown that for each of the possible values of m the group Z_m^* is cyclic. Let p be an odd prime.

Z_p^* is cyclic:

The order of an element $x \in Z_p - \{0\}$ is the least exponent e such that $x^e \equiv 1 \bmod p$. For each divisor d of $p-1$, let $S_d = \{x \in Z_p$: the order of x is $d\}$. Howevever, for each $x \in S_d$ and each $c < d, x^c \in S_d \Leftrightarrow \gcd(c,d) = 1$. (Indeed, on the one hand $(\Leftarrow)$ if $x^{ci} \equiv 1 \bmod p$ then $d|ci$, and hence $d|i$. Thus, the order of $x^c \bmod p$ is d, and on the other hand $(\Rightarrow)$ if $k = \gcd(c,d)$ then $(x^c)^{d/k} \equiv (x^{c/k})^d \equiv 1 \bmod p$ which implies that $k = \gcd(c,d) = 1$.) Let a be an arbitrary element of S_d. Then it is clear that $a^d \equiv 1 \bmod p$. Since Z_p is a finite field, the equation $x^d \equiv 1 \bmod p$ can have at most d solutions, namely $a, a^2, \ldots, a^d$. Therefore, $S_d \subseteq \{a, a^2, \ldots, a^d\}$. It follows from the above characterization of S_d that if S_d is nonempty then $|S_d| = \varphi(d)$. But, the family $\{S_d : d|(p-1)\}$ forms a partition of Z_p^*. It follows from Euler's theorem that

$$p - 1 = \sum_{d|(p-1)} \varphi(d) = \sum_{d|(p-1)} |S_d|$$

Consequently, for all $d|(p-1), |S_d| = \varphi(d)$ and Z_p^* must be a cyclic group.

The following claim will be useful in the sequel:

Claim 1: There exists a primitive root g modulo p such that for all $k > 1$, $g^{\varphi(p^{k-1})} \not\equiv 1 \bmod p^k$.

Proof of claim 1: Let g be a primitive root modulo p. Then notice that $(g+p)^{p-1} \equiv g^{p-1} + (p-1)pg^{p-2} \equiv g^{p-1} - pg^{p-2} \bmod p^2$. Hence, at least one of the two primitive roots $g, g+p$, say g_0, must satisfy the congruence $x^{p-1} \not\equiv 1 \bmod p^2$. The rest of the proof of the claim is by induction on k. It will be shown that g_0 satisfies the requirements of the claim. The proof in case $k = 2$ has already been completed. Assume by induction that $g_0^{\varphi(p^{k-1})} \not\equiv 1 \bmod p^k$. By the theorem of Euler-Fermat there exists an integer t such that $g_0^{\varphi(p^{k-1})} = 1 + tp^{k-1}$. By the induction hypothesis p does not divide t. It follows that $g_0^{\varphi(p^k)} \equiv (1 + tp^{k-1})^p \equiv 1 + tp^k + (1/2)p(p-1)t^2p^{2k-2} = 1 + tp^k \not\equiv 1 \bmod p^{k+1}$. This completes the proof of claim 1.

$Z^*_{p^k}$ is cyclic:

Let g be a primitive root modulo p which satisfies the condition of claim 1. It will be shown that for all $k > 0$, g is a primitive root modulo p^k. Let $k > 1$ be fixed, and let $e =$ least exponent such that $g^e \equiv 1$ modp^k. Clearly $g^e \equiv 1$ mod$p$, and hcnce $(p-1)|e$. However, $e|\varphi(p^k) = (p-1)p^{k-1}$. It follows that $e = \varphi(p^t) = (p-1)p^{t-1}$, for some $t \leq k$. But it is clear from the choice of g that t must be equal to k i.e. $e = \varphi(p^k)$.

$Z^*_{2p^k}$ is cyclic:

Let g be a primitive root modulo p^k, where k is positive. Let g_0 be the odd number among the two integers $g, g+p^k$. It will be showm that g_0 is a primitive root modulo $2p^k$. Indeed, $\varphi(2p^k) = \varphi(p^k)$. If one defines $e =$ least exponent such that $g_0^e \equiv 1$ mod$(2p^k)$, then it follows from the Euler-Fermat theorem that $e|\varphi(2p^k)$, and hence $e \leq \varphi(2p^k)$. But g_0 is a primitive root modulo p^k, and hence $e \geq \varphi(p^k)$. Hence $e = \varphi(p^k)$. This completes the proof of $(\Leftarrow)$

$(\Rightarrow)$

Suppose that m is not of the form $1, 2, 4, p^k, 2p^k$, where p is an odd prime and $k > 0$. It will be shown that

Claim 2: For all $a \in Z^*_m, a^{\varphi(m)/2} \equiv 1$ mod$m$

Proof of claim 2: If $m = 2^k$, then $\varphi(m)/2 = 2^{k-2}$. The claim will be proved by induction on k. The initial step $k = 3$ is trivial. Assume $a^{2^{k-2}} \equiv 1$ mod2^k is true. Then $a^{2^{k-2}} = 1 + t2^k$, for some t. Hence, $a^{2^{k-1}} \equiv (1+t2^k)^2 \equiv 1 + t2^{k+1} + t^2 2^{2k} \equiv 1$ mod2^{k+1}.

If $m = 2^k p^n$, where $k > 1$ and $n > 0$, then $\varphi(m)/2 = 2^{k-2}p^{n-1}(p-1)$ is divisible by both $\varphi(2^k)$ and $\varphi(p^n)$. Hence the claim follows in this case easily, using the result in case $m = 2^k$ and the Euler-Fermat theorem.

If $m = 2^k p_1^{n_1} \cdots p_r^{n_r}$ where $k > 1, r > 0$ then it is clear that $\varphi(m)/2 = 2^{k-2}\varphi(p_1^{n_1}) \cdots \varphi(p_r^{n_r})$. It follows that $\varphi(m)/2$ is divisible by each of the integers $\varphi(2^k), \varphi(p_1^{n_1}), \ldots, \varphi(p_r^{n_r})$ and the rest of the proof can be completed exactly as before. This completes the proof of the claim, and hence of the theorem.●

Figure 3 displays a table of the first ten primes and their corresponding least primitive root:

p	3	5	7	11	13	17	19	23	29	31
g	2	2	3	2	2	3	2	5	2	3

Figure 3: Table of primitive roots.

EXERCISES

1: Let g be a primitive root modulo m. Then for all nonnegative integers t, g^t mod$m$ is a primitive root modulo m if and only if $\gcd(t, \varphi(m)) = 1$. In particular, there exist exactly $\varphi(\varphi(m))$ primitive roots modulo m, provided there exists at least one primitive root modulo m.

2: Show that the groups Z_1^*, Z_2^*, Z_4^* are cyclic.

3: If $m = p_1^{e_1} \cdots p_r^{e_r}$ is the prime factorization of m and $q_i = p_i^{e_i}$ for $i = 1, \ldots, r$ then the group Z_m^* and the product group $Z_{q_1}^* \times \cdots \times Z_{q_r}^*$ are isomorphic.

4: Show that for each odd a and each $r \geq 3, a^{2^{r-2}} = 1 \bmod 2^r$.

5: Use exercise 4 to show that if $r \geq 3$ then $Z_{2^r}^*$ is isomorphic to the product of a cyclic group of order 2 and a cyclic group of order 2^{r-2}. **Hint:** -1 generates the group of order 2, and 5 the group of order 2^{r-2}.

1.8 ARTIN'S CONJECTURE

Theorem 1.9 gives a complete characterization of those m, for which the multiplicative group Z_m^* is cyclic i.e. Z_m^* has a generator. Howevever, the following natural questions arise:

Question 1: Is there an efficient algorithm which when given as input a prime number p will output a primitive root modulo p?

Question 2: Given a specific integer g, determine the primes p such that g is a primitive root modulo p.

The second question is also mentioned by Gauss in [Ga] for the special case $g = 10$. The results of section 3 make apparent the importance of these questions for the construction of pseudo-random generators. Nevertheless, to this date both of the above questions are open. Some empirical evidence is provided below (see [Scha], pp. 80 - 83, for additional empirical data). Let $\nu_g(n) =$ the number of primes $p \leq n$ such that g is a primitive root modulo p, and let $\pi(n) =$ the number of primes $\leq n$.

g	$\nu_g(10^4)$	$\nu_g(10^4)/\pi(10^4)$
2	470	.382
3	476	.387
5	492	.400
6	470	.382
7	465	.378

Figure 4: Artin's Constant

Based on probalistic heuristic considerations, Artin has conjectured that:

Conjecture 1: (Artin) Every integer $g \neq -1, 1$ which is not a complete square is a primitive root of infinitely many primes.

More exactly it is conjectured that:

Conjecture 2: For every integer $g \neq -1, 1$ which is not a complete square,

$$\frac{\nu_g(n)}{\pi(n)} \approx F_g \cdot A,$$

where A is Artin's constant (approximately equal to .37395...), and which is independent of n, g and F_g is a rational given in [Hoo]; for many values of g (e.g. $g = 2, 3, 6$) $F_g = 1$.

It is significant to note that Hooley in [Hoo] has confirmed Conjecture 2 under the assumption that Riemann's hypothesis holds for certain Dedekind functions.

EXERCISES

1: (Pratt [Prat]) The following result can be useful in testing if a given g is a primitive root modulo n, assuming that the prime factors of $n-1$ are known. Show that: g is a primitive root modulo $n \Leftrightarrow$ for all prime factors p of $n-1$, $g^{(n-1)/p} \not\equiv 1 \mathrm{mod} n$.

1.9 THE CARMICHAEL FUNCTION

A useful generalization of Euler's criterion is given through the Carmichael function λ. For each integer $m, \lambda(m)$ is defined as follows:

$$\lambda(2^t) = \begin{cases} \varphi(2^t) & \text{if } t < 3 \\ \varphi(2^t)/2 & \text{if } t \geq 3 \end{cases}$$

for any given integer $m = 2^t n$, where n is odd, one defines

$$\lambda(m) = \mathrm{lcm}(\lambda(2^t), \varphi(n)).$$

The intended improvement of the Euler-Fermat theorem (see theorem 1.3) is given in the theorem below

Theorem 1.10 *(Carmichael) For all $a \in Z_m^*, a^{\lambda(m)} \equiv 1$ mod$m$.*

Proof: It has been shown in theorem 1.8 that for any integer m which is not of the form $1, 2, 4, p^k, 2p^k$, where p is an odd prime, $k > 0$, and for all a in Z_m^*,

$$a^{\varphi(m)/2} \equiv 1 \mathrm{mod} m. \tag{3}$$

Let $p_0, p_1, \ldots, p_r$ be the distinct prime divisors of m, and for each i let q_i be the largest power of p_i dividing m. Hence, $m = q_0 \cdot q_1 \cdots q_r$. By the theorem of Euler-Fermat and the observation in equation 3, it is true that for all $a \in Z_m^*$, and all $i = 0, \ldots, r, a^{\varphi(q_i)} \equiv 1$ modq_i, and if $p_o = 2$ then $a^{\lambda(q_0)} \equiv 1$ modq_0. But this is enough to complete the proof of the theorem.●

EXERCISES

1: Let m be odd. Show that $\lambda(m)$ is the least exponent e such that for all $a \in Z_m^*, a^e = 1$ mod$m$. **Hint:** Let $p_1, \ldots, p_r$ be the distinct prime divisors of m, and for each i let q_i be the largest power of p_i dividing m. For each i let g_i be a primitive root modulo q_i. Fix an $i = 1, \ldots, r$. Use the Chinese Remainder Theorem to find an $a_i in Z_m^*$ such that $a_i \equiv g_i$ modq_i and $a_i \equiv 1$ modq_j for all $j \neq i$. Let e be the least exponent e such that for all $a \in Z_m^*, a^e \equiv 1$ mod$m$. By assumption, $a_i^e \equiv 1$ mod$m$ and hence $g_i^e \equiv 1$ modq_i. But g_i is a primitive root modulo q_i. Thus, $\varphi(q_i)|e$.

1.10 THE LAGRANGE SYMBOL

Call an x in Z_m^* **quadratic residue** modulo m, if $x \equiv y^2$ mod$m$ for some $y \in Z_m^*$; otherwise x is called a **quadratic nonresidue** modulo m. Let QR_m (respectively QNR_m) be the set of all quadratic residues (respectively nonresidues) modulo m.

For each prime number p, and any $x \in Z_m^*$ let

$$(x|p) = \begin{cases} 1 & \text{if } x \in QR_p \\ -1 & \text{if } x \in QNR_p. \end{cases}$$

$(x|p)$ is called the **Lagrange symbol** of x modulo p.

Remark: The symbol $(\frac{x}{p})$ is also widely used in the literature as identical to the symbol $(x|p)$.

One of the most useful properties of the Langrange symbol is expressed in the following

Theorem 1.11 (**Euler's Criterion**) *For all primes $p > 2$, and all $x \in Z_p^*$,*

$$x^{(p-1)/2} = (x|p) \text{ mod} p$$

Proof: Let $x \in Z_p^*$. Then $x^{p-1} \equiv 1$ mod$p$, and hence either $x^{(p-1)/2} \equiv 1$ mod$p$ or $x^{(p-1)/2} \equiv -1$ mod$p$. The mapping $f : Z_p^* \longrightarrow \{-1, 1\}$ such that $f(x) \equiv x^{(p-1)/2}$ mod$p$, is a group homomorphism. Since for any primitive root g of $Z_p^*, g^{(p-1)/2} \equiv -1$ mod$p$, the mapping f is onto and consequently the kernel K of f is a proper subgroup of Z_p^* of size $(p-1)/2$ (see theorem 1.1).

If $(x|p) = 1$, then $x \in QR_p$. Thus, $x \equiv y^2$ mod$p$ for some $y \in Z_p^*$. It follows that $x^{(p-1)/2} \equiv y^{p-1} \equiv 1$ mod$p$, by the theorem of Euler-Fermat (see theorem 1.3.) Thus, $x \in K \Rightarrow x^{(p-1)/2} \equiv 1$ mod$p$.

However, $QR_p = \{1^2 \text{ mod} p, 2^2 \text{ mod} p, \ldots, (p-1)^2 \text{ mod} p\}$. Moreover, $(p - x)^2 \equiv p^2 - 2px + x^2 \equiv x^2$ mod$p$ for all $x \in Z_p^*$. Thus, exactly one half the numbers in Z_p^* are quadratic residues, and the other half quadratic nonresidues modulo p. Since, $K \supseteq QR_p$, and both K, QR_p have exactly the same size, $(p-1)/2$, it follows that $K = QR_p$. Thus, $K = QR_p$.

If $(x|p) = -1$, then $x \in QNR_p$. It follows from the above remarks $x \notin K$ and $x^{(p-1)/2} \equiv -1 \bmod p$. Thus, $x \notin K \Rightarrow x^{(p-1)/2} \equiv -1 \bmod p$. which completes the proof of the theorem.●

EXERCISES

1: The following is a generalization of Euler's criterion (see theorem 1.11) : for all primes $p > 2$, for all $k > 0$, and all $x \in Z^*_{p^k}, x^{\varphi(p^k)/2} \equiv 1 \bmod p \Leftrightarrow x \in QR_{p^k}$. **Hint:** Argue as in the above proof to show that the mapping $f : Z^*_{p^k} \longrightarrow \{-1, 1\}$ such that $f(x) = x^{\varphi(p^k)/2} \bmod p^k$, is a group epimomorphism, whose kernel K equals QR_{p^k}.

2: Use exercise 1 to show that for all integers $x \in Z^*_{p^k}, x \in QR_p \Leftrightarrow x \in QR_{p^k}$. **Hint:** ($\Rightarrow$) Let $x \in QR_p$ and put $a = x^{\varphi(p)/2}$. Then $p|(x^{\varphi(p)/2} - 1)$. Hence, $x^{\varphi(p^k)/2} - 1 = a^{p^{k-1}} - 1 = (a-1)(a^{p^{k-1}-1} + a^{p^{k-1}-2} + \ldots + a + 1)$. Notice that the second factor of the last product is divisable by p^{k-1}.

3: If both $x, y \in QNR_p$ then $xy \in QR_p$.

1.11 THE LANGRANGE-JACOBI SYMBOL

The definition of of the Langrange symbol can be extended to all m and all x in Z^*_m. Indeed, let $m = p_1 \cdots p_r$, where $p_1, \ldots, p_r$ are primes. Then the **Langrange-Jacobi symbol** is defined by $(x|m) = (x|p_1) \cdots (x|p_r)$. One also defines the sets $Z^*_m(+1) = \{x \in Z^*_m : (x|m) = 1\}$, and $Z^*_m(-1) = \{x \in Z^*_m : (x|m) = -1\}$.

In determining whether a given $x \in Z^*_p$ is a quadratic residue modulo a prime p one needs to compute $(x|p)$. This is in fact done using the next two theorems.

Theorem 1.12 (**Evaluating** $(x|m)$) *Let* $x, y \in Z^*_m$.

(i) If $x \equiv y \bmod m$ *then* $(x|m) = (y|m)$

(ii) $(x|m) \cdot (y|m) = (x \cdot y|m)$

(iii) $(-1|m) = (-1)^{(m-1)/2}$

(iv) $(2|m) = (-1)^{(m^2-1)/8}$, *where* m *is odd.*

Proof: The proofs of (i), (ii) are easy and are left as an exercise to the reader. As a first step, the theorem will be reduced to the case of the Lagrange symbol, i.e. both m and n are primes. This reduction is based on the following claim whose proof is straightforward

Claim: If s, t are odd then

$$\frac{s-1}{2} + \frac{t-1}{2} \equiv \frac{st-1}{2} \bmod 2, \text{ and}$$

$$\frac{s^2-1}{8}+\frac{t^2-1}{8}\equiv\frac{s^2t^2-1}{8}\ \mathrm{mod} 2$$

Using the definition of the Jacobi symbol, the reduction to the case of the Langrange symbol is an immediate consequence of (4-7) below. Let $m = p_1\cdots p_r, n = q_1\cdots q_t$ be the prime number factorizations of m, n respectively. Then the above claim implies that

$$\frac{p_1-1}{2}+\cdots+\frac{p_r-1}{2}\equiv\frac{m-1}{2}\ \mathrm{mod} 2, \tag{4}$$

$$\frac{q_1-1}{2}+\cdots+\frac{q_r-1}{2}\equiv\frac{n-1}{2}\ \mathrm{mod} 2, \tag{5}$$

$$\frac{p_1^2-1}{8}+\cdots+\frac{p_r^2-1}{8}\equiv\frac{m^2-1}{8}\ \mathrm{mod} 2, \tag{6}$$

$$\frac{q_1^2-1}{8}+\cdots+\frac{q_r^2-1}{8}\equiv\frac{n^2-1}{8}\ \mathrm{mod} 2. \tag{7}$$

From now on it will be assumed that $m = p, n = q$ are primes.

It is now obvious that (iii) is an immediate consequence of Euler's criterion. It only remains to give the proof of (iv)

Since $((p-1)/2)!$ is the product of numbers all of which are less than p, it is clear that p does not divide $((p-1)/2)!$. Also notice that

$$(-1)^k\cdot k\equiv\begin{cases} k & \text{if } k \text{ is even}\\ p-k & \text{if } k \text{ is odd}\end{cases}$$

It follows that on the one hand

$$(-1)^1 1(-1)^2 2\cdots(-1)^{(p-1)/2}\frac{p-1}{2} = \tag{8}$$

$$\left(\frac{p-1}{2}\right)!(-1)^{1+2+\cdots+(p-1)/2} = \left(\frac{p-1}{2}\right)!(-1)^{(p^2-1)/8}, \tag{9}$$

and on the other hand using the theorem of Euler-Fermat,

$$(-1)^1 1(-1)^2 2\cdots(-1)^{(p-1)/2}\frac{p-1}{2}\equiv 2\cdot 4\cdot 6\cdots(p-1)\equiv \tag{10}$$

$$2^{(p-1)/2}\left(\frac{p-1}{2}\right)!\equiv(2|p)\left(\frac{p-1}{2}\right)!\ \mathrm{mod} p. \tag{11}$$

The result now follows combining equations (8-11) and the fact that p does not divide $((p-1)/2)!$.●

Theorem 1.13 *(***Law of Quadratic Reciprocity, Gauss***) For all odd $m, n > 2$ which are relatively prime the following equation holds*

$$(n|m)\cdot(m|n) = (-1)^{(m-1)/2}\cdot(-1)^{(n-1)/2}$$

Proof: For any set $M \subseteq Z_m^*$ define the set $-M = \{-a : a \in M\}$. For the given primes p, q let $P = \{1, 2, \ldots, (p-1)/2\}$ and $Q = \{1, 2, \ldots, (q-1)/2\}$. It is clear that for any $c \in Q$ there exists $b \in Q$ such that either $pc \equiv b \bmod q$ or $pc \equiv -b \bmod q$ (a similar property holds for P). Define $n(p,q) =$ the number of times that pc is congruent modulo q to an integer in $-Q$, as c runs through Q. The proof is based on the following

Lemma 1.1 *(Gauss' Lemma)* $(p|q) = (-1)^{n(p,q)}$

Proof of the Lemma: For each $c \in Q$ one can find s_c, b_c

$$pc \equiv s_c b_c \bmod q, \tag{12}$$

where $b_c \in Q$ and $s_c = +1$ or -1. The mapping $c \longrightarrow b_c (c \in Q, b_c \in Q)$ is $1-1$ (and hence also onto.) Indeed, assume that $b_c = b_d$. Hence, either $pc \equiv pd \bmod q$ or $pc \equiv -pd \bmod q$. But p, q are relatively prime. It follows from the definition of Q that $c = d$. Hence, the mapping $c \longrightarrow b_c$ is a permutation of Q and

$$\left(\frac{q-1}{2}\right)! = b_1 b_2 \cdots b_{(q-1)/2}. \tag{13}$$

Multiplying the congruences (12) as c ranges over Q and using (13) one obtains that,

$$p^{(q-1)/2} \equiv (-1)^{n(p,q)} \bmod q. \bullet$$

The proof of the lemma can now be easily completed using the Euler's criterion.

Returning to the proof of the theorem let $\vartheta = \epsilon^{2\pi i/p}$ (respectively $\varrho = \epsilon^{2\pi i/q}$) be the primitive p-th (respectively q-th) root of unity. It follows from Gauss's Lemma and the fact that for all $a \in Q$ there exists $b \in Q \cup (-Q)$ such that $pa \equiv b \bmod q$, that

$$(p|q) = (-1)^{n(p,q)} = \prod_{a \in Q} \frac{\varrho^{pa} - \varrho^{-pa}}{\varrho^{a} - \varrho^{-a}}. \tag{14}$$

However, the following identity holds for all $x \neq 0$,

$$x^p - x^{-p} = \prod_{b \in Z_p} (x\vartheta^b - x^{-1}\vartheta^{-b}) \tag{15}$$

To see this, multiply both sides of (15) by x^p and use $\vartheta^p = 1$ to show that the resulting polynomials have the same leading coefficient 1 and the same zeroes: $\vartheta^b, -\vartheta^b$, where $b = 0, \ldots, p-1$. Combining (14) and (15) one easily obtains that

$$(p|q) = \left(\prod_{a \in Q} \prod_{b \in Z_p} (\varrho^a \vartheta^b - \varrho^{-a} \vartheta^{-b})\right) \Big/ \left(\prod_{a \in Q} (\varrho^a - \varrho^{-a})\right)$$

$$= \prod_{a \in Q} \prod_{b \in Z_p - \{0\}} (\varrho^a \vartheta^b - \varrho^{-a} \vartheta^{-b})$$

$$= \prod_{a \in Q} \prod_{b \in P} (\varrho^a \vartheta^b - \varrho^{-a} \vartheta^{-b}) \cdot (\varrho^{-a} \vartheta^b - \varrho^a \vartheta^{-b}).$$

Hence,

$$(p|q) = \prod_{a \in Q, b \in P} ((\vartheta^{2b} + \vartheta^{-2b}) - (\varrho^{2a} + \varrho^{-2a})).$$

Interchanging the roles of p and q one also obtains

$$(q|p) = \prod_{a \in Q, b \in P} ((\varrho^{2a} + \varrho^{-2a}) - (\vartheta^{2b} + \vartheta^{-2b})).$$

Since, each of the last two products has $(p-1)(q-1)/4$ factors, the proof of the Law of Quadratic Reciprocity is complete.●

It is not hard to see that computing the Jacobi symbol $(x|m)$ of two relatively prime integers x, m is similar to computing the greatest common divisor of x, m. This is illustrated in the example below.

Example 1.4 *Show that* $76 \notin QNR_{131}$. *Indeed,*
$(76|131) = (2|131) \cdot (2|131) \cdot (19|131) =$
$(19|131) = (131|19) \cdot (-1)^{(131-1)/2} \cdot (-1)^{(19-1)/2} =$
$(17|19) \cdot (-1) \cdot (-1) = (19|17) \cdot (-1)^{(19-1)/2} \cdot (-1)^{(17-1)/2} =$
$-(19|17) = -(2|17) = -(-1)^{(17^2-1)/8} = -1$

An analysis similar to that in the proof of Lamé's theorem, (see theorem 1.2) shows that

Theorem 1.14 *If* $N \geq a, b > 0$ *are integers, with* a, b *relatively prime then the number of steps required to compute* $(a|b)$ *is* $O(\log_R N)$.●

EXERCISES

1: Prove properties (i), (ii) of theorem 1.12.

2: Compute $(56|39)$.

3: Determine $(3|p)$, where p is a prime > 3.

4: For $k > 2$ and a odd, $a \in QR_{2^k} \Leftrightarrow a \equiv 1 \bmod 8$ **Hint:** $(\Rightarrow)$ see the proof of claim 2 in theorem 1.9;$(\Leftarrow)$ use exercise 5 of subsection 1.7.

From now on assume that p is an odd prime and k ¿ 1.

5: Show that $(1+p)^{p^{k-1}} \equiv \bmod p^k$ and $(1+p)^{p^{k-2}} \equiv (1+p^{k-1}) \not\equiv 1 \bmod p^k$. Hence, $(1+p)$ generates in $Z^*_{p^k}$ a cyclic subgroup H of $Z^*_{p^k}$ of order p^{k-1}.

6: (This is a continuation of exercise 5.) Let g be a primitive root modulo p. Then $g_0 = g^{p^{k-1}}$ is a primitive root modulo p and g_0 generates a cyclic subgroup G of $Z^*_{p^k}$ of order $p-1$. Moreover, $G \times H$ is isomorphic to $Z^*_{p^k}$. Every element $a \in Z^*_{p^k}$ can be written uniquely in the form $a \equiv g_0^t(1+p)^r$ modp^k, where $0 \leq t < p-1$ and $0 \leq r < p^{k-1}$. (exercise 6 gives a new proof of the cyclicity of $Z^*_{p^k}$, for $k > 1$.)

7: For all a, if a is relatively prime to p then $a \in QR_p \Leftrightarrow a \in QR_{p^k}$ **Hint:** ($\Rightarrow$) Write a in the form $a = g_0^t(1+p)^r$ modp^k, where $0 \leq t < p-1$ and $0 \leq r < p^{k-1}$. Notice that t is even. Find c such that $2c \equiv 1$ modp^{k-1} and let $b \equiv g_0^{t/2}(1+p)^{cr}$ modp^k. Show that $a \equiv b^2$ modp^k. (exercise 7 gives a new proof of exercise 2 in subsection 1.10.)

8: Give the proof of theorem 1.14.

1.12 COMPUTING SQUARE ROOTS

One of the most important problems in complexity theory is to find an efficient algorithm which given as input an $x \in QR_n$ and an integer n it will output a square root of x modulo n. It will be seen in the sequel that such an algorithm exists if n is prime. It will also be shown that for composite n, the above problem is equivalent to the problem of finding an efficient algorithm which given as input n it will output the factors of n.

If p is an odd prime number then such an efficient probabilistic procedure for computing square roots modulo p is given in the theorem below.

Theorem 1.15 (**Adleman-Manders-Miller**) *There exists an efficient probabilistic polynomial time algorithm which when given as inputs an odd prime p and an $a \in QR_p$ it will output a square root of a modulo p.*

Proof: Let p be a prime and $a \in QR_p$. Write $p-1$ in the form $p-1 = 2^eP$, where P is odd. Choose any random $b \in QNR_p$. Define a sequence $a_1, a_2, \ldots, a_n, \ldots$ of quadratic residues modulo p and a sequence of indices $e \geq k_1 > \ldots > k_n > \ldots$ as follows by induction on n:

$$a_1 = a,$$

$$k_{n-1} = \text{least k such that } a_{n-1}^{2^kP} \equiv 1 \text{ mod}p,$$

$$a_n \equiv a_{n-1}b^{2^{e-k_{n-1}}} \text{ mod}p.$$

However, it is true that

$$a_n^{2^{k_{n-1}-1}P} \equiv (a_{n-1}b^{2^{e-k_{n-1}}})^{2^{k_{n-1}-1}P} \equiv$$

$$a_{n-1}^{2^{k_{n-1}-1}P}b^{2^{e-1}P} \equiv (-1)(-1) \equiv 1 \text{ mod}p,$$

The proof of the last congruence uses the fact that $b^{2^{e-1}P} = b^{(p-1)/2} \equiv (b|p) = -1$. Using the minimality of k_{n-1}, the above congruences, and the Euler-Fermat theorem, it follows that for all integers n if $k_{n-1} > 0$ then $k_n < k_{n-1}$. Hence, there exists an $n \leq e$ such that $k_n = 0$, and for such an n, $a_n^{(P+1)/2}$ is a square root of a_n. Next one defines by reverse induction a sequence $r_1, \ldots, r_n$ such that for all $i, r_i^2 \equiv a_i \bmod p$. Indeed, let $r_n \equiv a_n^{(P+1)/2} \bmod p$. Assume that r_{i+1} has already been defined and let $r_i \equiv r_{i+1}(b^{2^{e-k_i-1}})^{-1} \bmod p$. It is straightforward to see that for all $i, a_i \equiv r_i^2 \bmod p$.

The above observations provide an efficient algorithm, described more explicitely below, to compute square roots modulo a prime number p. One merely chooses a random b such that $(b|p) = -1$ and then follows the above described procedure with input p, a. More explicitely,

Input: p (prime), $a \in QR_p$.

Step 1: Compute an odd P and e such that $p - 1 = 2^e P$.

Step 2: Choose random b such that $(b|p) = -1$.

Step 3: Set $y := a, r := a^{(P+1)/2} \bmod p$.

Step 4: Find the least k such that $y^{2^k P} \equiv 1 \bmod p$.

Step 5: **If** $k = 0$ then output r **else** set

$$y \equiv y b^{2^{e-k}} \bmod p, r \equiv r(b^{2^{e-k-1}})^{-1} \bmod p$$

and go to step 4.

Output: r.

The running time of the algorithm is a polynomial in the lengths of p and a, plus the time required to find an integer b such that $(b|p) = -1$.•

The case of composite n is studied in the theorem below.

Theorem 1.16 *For all $x \in Z_{pq}^*$, where p, q are distinct odd primes, $x \in QR_{pq} \Leftrightarrow x \in QR_p$ and $x \in QR_q$. Moreover there is an efficient algorithm which given as inputs x, u, v, p, q, where p and q are distinct odd primes and $x \equiv u^2 \bmod p$ and $x \equiv v^2 \bmod q$, will output a w such that $x \equiv w^2 \bmod (pq)$.*

Proof: Suppose that x, u, v, p, q are as in the hypothesis of the theorem. Since p, q are relatively prime one can compute efficiently integers a, b such that $1 = ap + bq$. Put $c = bq = 1 - ap$ and $d = ap = 1 - bq$. It is then clear that

$$c \equiv 0 \bmod q, c \equiv 1 \bmod p, d \equiv 0 \bmod p, d \equiv 1 \bmod q.$$

It will be shown that $w = cu + dv$ is a quadratic residue modulo n. It is enough to show that w is a quadratic residue both modulo p and modulo q. Indeed,

$$w^2 \equiv (cu + dv)^2 \equiv (c^2u^2 + d^2v^2 + 2cduv) \equiv u^2 \equiv x \bmod p.$$

A similar calculation shows that $w^2 \equiv x \bmod q$. This completes the proof of the theorem.•

EXERCISES

1: Extend theorem 1.16 to the case of products of relatively prime integers.

1.13 INDICES

Let p be a prime and let g be a primitive root modulo p. It is known that $Z_p^* = \{g^0, g^1, \ldots, g^{p-2}\}$, and hence for any $x \in Z_p^*$ one can define the **index** or **discrete logarithm** of x with respect to g, abbreviated $\text{index}_{p,g}(x)$, as the unique $m \leq p-2$ such that $x \equiv g^m \bmod p$.

The following theorem gives a very useful characterization of quadratic residues in terms of the above defined index.

Theorem 1.17 (**Characterization of Quadratic Residues**) *Let p be an odd prime, and let g be a primitive root modulo p. Then for any $x \in Z_p^*, x \in QR_p \Leftrightarrow$ $\text{index}_{p,g}(x)$ is even.*

Proof: The proof of $(\Leftarrow)$ is easy and is left to the reader.

$(\Rightarrow)$

Let x be a quadratic residue modulo p. There exists an integer u such that $x \equiv u^2 \bmod p$. Let $t = \text{index}_{p,g}(u) < p-1$. Then $u \equiv g^t \bmod p$ and $x \equiv g^{2t} \bmod p$. It follows that $\text{index}_{p,g}(x) \equiv 2t \bmod (p-1)$, and hence $\text{index}_{p,g}(x)$ is even.•

If $n = pq$ is the product of two distinct primes then the product mapping $< x, y > \longrightarrow xy$ is an isomorphism between the groups $Z_p^* \times Z_q^*$ and Z_n^*. Let g (respectively h) be a primitive root modulo p (respectively q). Then any element in Z_n^* can be written in a unique way in the form $z = g^r h^t$. As before, let the index of z with respect to g, h, abbreviated $\text{index}_{n,g,h}(z)$ be the pair $< r, t >$.

Using the multiplication table of Z_{11}^* one can compute the following table of indices:

x	1	2	3	4	5	6	7	8	9	10
$\text{index}_{11,2}(x)$	0	1	8	2	4	9	7	3	6	5

Figure 5: Table of values of $\text{index}_{11,2}(x)$.

EXERCISES

1: Prove $(\Leftarrow)$ in theorem 1.17.

2: Let $n = pq$ be the product of two distinct odd primes such that g (respectively h) is a primitive root modulo p (respectively q.) For any $z \in Z_n^*, z \in QR_n \Leftrightarrow$ both components of $\text{index}_{n,g,h}(z)$ are even.

3: Let g be a primitive root modulo the prime $p > 2$. Show that for all $a, b \in Z_p^*$, and all $n > 0$,

1. $\mathrm{index}_{p,g}(ab) \equiv \mathrm{index}_{p,g}(a) + \mathrm{index}_{p,g}(b) \bmod (p-1)$.
2. $\mathrm{index}_{p,g}(a^n) \equiv n \cdot \mathrm{index}_{p,g}(a) \bmod (p-1)$.
3. $\mathrm{index}_{p,g}(1) = 0$.
4. $\mathrm{index}_{p,g}(g) = 1$.
5. $\mathrm{index}_{p,g}(-1) = (p-1)/2$.

4: If g is a primitive root modulo p, then $g \notin QR_p$.

1.14 COMPUTING INDICES

One of the most significant problems in complexity theory is to find an efficient algorithm A such that for any prime p, any primitive root g modulo p and any $x \in Z_p^*, A(p, g, x) = \mathrm{index}_{p,g}(x)$. This problem is very significant for the construction of secure cryptographic protocols. In general, no such algorithm is known. However, the theorem below provides such an efficient algorithm in the case where the prime factorization of $p-1$ is known. For each integer n, $|n|$ denotes the binary length of n.

Theorem 1.18 (**Pohlig-Hellman**) *For any polynomial poly(.) there exists an efficient algorithm A such that if p is a prime such that the prime factors* $p_1, \ldots, p_r$ *of* $p-1$ *satisfy* $p_1, \ldots, p_r \leq poly(|p|)$, g *is a generator of* Z_p^*, *and* $y \in Z_p^*$ *then* $A(p, g, p_1, \ldots, p_r, y) = \mathrm{index}_{p,g}(y)$. *Moreover, A runs in time polunomial in the length* $|p|$ *of p.*

Proof: Let p be a prime number such that prime factors $p_1, \ldots, p_r$ of $p-1$ satisfy $p_1, \ldots, p_r \leq poly(|p|)$. Let g be a generator of Z_p^*, and let $y \in Z_p^*$. For each $j = 1, \ldots, r$, let $e_j =$ the largest exponent e such that $p_j^e | (p-1)$ and let $q_j = p_j^{e_j}$. For each $j = 1, \ldots, r$ define $y_j \equiv y^{(p-1)/q_j} \bmod p, g_j \equiv g^{(p-1)/q_j} \bmod p, x_j =$ the unique x such that $g_j^x \equiv y_j \bmod p$. Notice that $x_j < q_j$. Thus, x_j can be represented in the number base p_j as follows:

$$x_j = x_{j,0} + x_{j,1} p_j + x_{j,2} p_j^2 + \ldots + x_{j,e_j - 1} p_j^{e_j - 1}, \text{ where } x_{j,i} < p_j. \qquad (*)_j$$

The idea of the proof is the following: first, one gives an algorithm A_1 which on input p, p_j, g, y computes $x_{j,0}$ as above; second, one extends A_1 to give an algorithm A_2 that on input p, p_j, g, y computes x_j; and third, one uses the Chinese Remainder theorem to compute the $\mathrm{index}_{p,g}(y)$ from the previously computed x_j.

The algorithm A_1 is given below.

Input: p, p_j, g, y.

Step 1: Compute q_j, g_j, y_j as above.

Step 2: Compute $z_j \equiv y_j^{(p-1)/p_j}$ mod$p$.

Step 3: Compute $i_0 =$ the first $i < p_j$ such that $z_j \equiv g_j^{i(p-1)/p_j}$ mod$p$.

Output: i_0.

It is a consequence of the Euler-Fermat theorem that $i_0 = x_{j,0}$. Indeed, using $(*)_j$ one obtains $i_0 = x_{j,0}$ through the following congruences: $z_j \equiv g_j^{i_0(p-1)/p_j} \equiv y_j^{(p-1)/p_j} \equiv g_j^{x_j(p-1)/p_j} \equiv g_j^{x_{j,0}(p-1)/p_j}$ mod$p$.

An easy extension of the above algorithm gives a new algorithm to compute x_j. Indeed, consider the following algorithm A_2 defined by

Input: p, p_j, g, y.

Step 1: Compute q_j, e_j, g_j, y_j.

Step 2: Compute $A_1(p, p_j, g, y)$.

Step 3: Put $g_{j,0} = g_j, y_{j,0} = y_j, c_{j,0} = A_1(p, p_j, g, y)$.

Step 4: For $i = 0$ to e_{j-1} do: Compute the following

$$g_{j,i+1} \equiv g_{j,i}^{p_j} \text{ mod} p, y_{j,i+1} \equiv y_{j,i} g_{j,i}^{-c_{j,i}} \text{ mod} p.$$

Step 5: Compute $c_{j,i+1} = A_1(p, p_j, g_{j,i+1}, y_{j,i+1})$.

Output: $c_{j,0} + c_{j,1}p_j + c_{j,2}p_j^2 + \ldots + c_{j,e_j-1}p_j^{e_j-1}$

To prove the correctness of A_2 one shows by induction on $i \leq e_j$, that $c_{j,i} = x_{j,i}$ e.g. it has already been shown that $x_{j,0} = c_{j,0}$. Thus,

$$y_{j,1} \equiv y_{j,0} g_{j,0}^{-c_{j,0}} \equiv$$

$$g_j^{x_{j,1}p_j + \ldots + x_{j,e_j-1}p_j^{e_j-1}} \equiv (g_j^{p_j})^{x_{j,1} + \ldots + x_{j,e_j-1}p_j^{e_j-2}}.$$

Consequently,

$$x_{j,1} = A_1(p, p_j, g_{j,1}^{p_j}, y_j g_j^{-x_{j,0}}) = A_1(p, p_j, g_{j,1}, y_{j,1}) = c_{j,1}.$$

The proof of $c_{j,i} = x_{j,i}$ $(i > 1)$ is similar.

The rest of the proof is an application of the Chinese remainder theorem. Indeed, consider the following algorithm A:

Input: $p, g, p_1, \ldots, p_r, y$.

Step 1: Compute $x_j = A_2(p, p_j, g, y)$.

Output: The unique x such that for all $j = 1, \ldots, r$, $x \equiv x_j$ mod$(q_1 \ldots q_r)$.

To see that A works notice that for all $j = 1, \ldots, r, y^{(p-1)/q_j} \equiv g^{x_j(p-1)/q_j}$ mod$p$ and $x \equiv x_j$ modq_j. Since, gcd$((p-1)/q_1, \ldots, (p-1)/q_r) = 1$, there exist $t_1, \ldots, t_r$ such that $t_1(p-1)/q_1 + \ldots + t_r(p-1)/q_r = 1$. It follows from the Euler-Fermat theorem that

$$y \equiv y^1 \equiv y^{t_1(p-1)/q_1 + \ldots + t_r(p-1)/q_r} \equiv y^{t_1(p-1)/q_1} \ldots y^{t_r(p-1)/q_r} \equiv$$

$$g^{x_1 t_1(p-1)/q_1} \ldots g^{x_r t_r(p-1)/q_r} \equiv g^{x_1 t_1(p-1)/q_1 + \ldots + x_r t_r(p-1)/q_r} \equiv g^x \text{ mod} p,$$

which completes the proof of the theorem.●

1.15 THE PRIME NUMBER THEOREM

Let b, c be relatively prime positive integers, and let $\pi_{b,c}(x)$ = the number of primes $p \leq x$ such that p is of the form $p = bk + c$. If $b = 1, c = 0$ then $\pi(x) = \pi_{1,0}(x)$ = the number of primes $p \leq x$. The **prime number theorem** is the following statement (where the logarithm is taken with respect to the base e) :

Theorem 1.19 (**Dirichlet, Hadamard, de la Vallée Poussin**) *If* $\gcd(b, c) = 1$, *then*

$$\lim_{x \to \infty} \frac{\pi_{b,c}(x)}{x/\log x} = \frac{1}{\varphi(b)} \bullet$$

In particular, as a special case of theorem 1.19 one obtains that

$$\lim_{x \to \infty} \frac{\pi(x)}{x/\log x} = 1.$$

The following table gives some values of $\pi(n)$.

n	10^1	10^2	10^3	10^4	10^5	10^6	10^7
$\pi(n)$	4	25	168	1,229	9,592	78,498	664,579

Figure 6: Table of values of $\pi(n)$.

A proof of theorem 1.19 would lie outside the scope of the present section. However, the proof of the following weaker version of the prime number is very simple and elegant.

Theorem 1.20 (**Chebyshev**) *For all* $x > 1,200$,

$$\frac{2}{3} \frac{x}{\log x} < \pi(x) < \frac{17}{10} \frac{x}{\log x} \tag{16}$$

Proof: (Zagier) For simplicity throughout the present proof p will range over prime numbers. For each real r, let $\lfloor r \rfloor$ denote the integral part of r.

Proof of the lower bound: This is is based on the following

Claim: For all x, k the following holds:

$$\binom{x}{k} \leq x^{\pi(x)}. \tag{17}$$

Proof: Let x be fixed. It is clear that for any power p^t of the prime p the number of integers among $1, 2, \ldots, x-1, x$ divisable by p^t is exactly $\lfloor x/p^t \rfloor$. For each integer n let

$$e(n,p) = \text{the largest exponent } e \text{ such that } p^e | n,$$

Further, let

$$\Delta_e = \{1 \leq d \leq x : p^e | d\}.$$

It is then clear that

$$|\Delta_e| = \left[\frac{x}{p^e}\right].$$

Moreover,

$$\Delta_1 \supseteq \Delta_2 \supseteq \cdots \supseteq \Delta_e \supseteq \Delta_{e+1} \supseteq \cdots.$$

However, for any $1 \leq d \leq x$,

$$d \notin \Delta_{e(d,x)+1}, d \in \Delta_{e(d,x)} \subseteq \cdots \subseteq \Delta_1.$$

Hence, each $1 \leq d \leq x$ is counted in the sum $\sum_{e \geq 1} |\Delta_e|$ exactly $e(d,x)$ times. It follows that

$$e(x!, p) = \sum_{d=1}^{x} e(d,p) = \sum_{e \geq 1} |\Delta_e| = \sum_{e \geq 1} \left\lfloor \frac{x}{p^e} \right\rfloor.$$

It follows from the last equation and the definition of the binomial coefficient that

$$e\left(\binom{x}{k}, p\right) = e(x!, p) - e((x-k)!, p) - e(k!, p) =$$

$$\sum_{e \geq 1} \left(\left[\frac{x}{p^e}\right] - \left[\frac{x-k}{p^e}\right] - \left[\frac{k}{p^e}\right] \right). \tag{18}$$

Since, each of the summands in (18) is either 0 or 1, and all summands vanish if $e > \log x / \log p$, it is clear that $e\left(\binom{x}{k}, p\right) \leq \lfloor \log x / \log p \rfloor$ and hence, $p^{e\left(\binom{x}{k}, p\right)} \leq x$. Now, the claim follows from the fact that

$$\binom{x}{k} = \prod_{p \leq x} p^{e\left(\binom{x}{k}, p\right)} \leq x^{\pi(x)}.$$

This completes the proof of the claim. To complete the lower bound proof apply (17) to $k = 0, 1, \ldots, x$, and add the resulting inequalities to obtain:

$$2^x = \sum_{k=0}^{x} \binom{x}{k} \leq (x+1) \cdot x^{\pi(x)}.$$

Taking the logarithm of both sides of the above inequality one obtains that

$$\pi(x) \geq \frac{x \log 2}{\log x} - \frac{\log(x+1)}{\log x} > \frac{2}{3}\frac{x}{\log x}.$$

(Notice that the right side of the last inequality is valid for $x > 200$.)

Proof of the upper bound: It is clear from the definition of the binomial coefficient that

$$\prod_{x<p\leq 2x} p \text{ divides } \binom{2x}{x}.$$

But, the product to the left of the above equation has exactly $\pi(2x) - \pi(x)$ factors, all of them $\geq x$. Hence, using the binomial theorem one obtains that

$$x^{\pi(2x)-\pi(x)} \leq \prod_{x<p\leq 2x} p \leq \binom{2x}{x} < \sum_{i=0}^{2x} \binom{2x}{i} = 2^{2x},$$

and taking the logarithm (in base e) of both sides of the above inequality

$$\pi(2x) - \pi(x) < \frac{2x \log 2}{\log x} < 1.39\frac{x}{\log x}. \qquad (19)$$

Next, assume by induction that the right side of equation (16) is true for x. It follows from (19) and the induction hypothesis that

$$\pi(2x) < \pi(x) + 1.39\frac{x}{\log x} < 3.09\frac{x}{\log x} < 1.7\frac{2x}{\log(2x)}. \qquad (20)$$

Moreover, using (20) one obtains that

$$\pi(2x+1) \leq \pi(2x) + 1 < 3.09\frac{x}{\log x} + 1 < 1.7\frac{2x+1}{\log(2x+1)},$$

which completes the induction proof and hence the proof of the theorem. •

For more information the reader should consult [E] (pp. 23 - 25, and exercises 1.8 - 1.13 in pages 30 - 31).

EXERCISES

1: Use $\varphi(4) = 2$, to show that asymptotically for all x, half the primes $p \leq x$ satisfy $p \equiv 3 \bmod 4$.

2: How many primes of a given length k exist? **Hint:** Use theorem 1.19.

3: The result of the present exercise is used in the proof of theorem 1.7. Let p_n be the n-th prime and let $0 < \alpha < 1$ be a fixed real number. Let $\pi(n, \alpha)$

denote the number of primes in the interval $(p_n^\alpha, p_n]$. Use the prime number theorem to show that for all n, t,

$$\pi(n+m,\alpha) \approx (n+t)\cdot\left(1-\frac{1}{\alpha p_{n+t}^{1-\alpha}}\right).$$

In particular, for any $2 \leq k \leq n$ there exist arbitrarily large integers t such that

$$\pi\left(t, \frac{k^2-1}{k^2}\right) > n. \tag{21}$$

1.16 CONTINUED FRACTIONS

For any two positive real numbers α, β let $[\alpha, \beta] = \alpha + 1/\beta$. This notation is extended by induction to sequences $\alpha_1, \ldots, \alpha_n, \ldots$ of positive real numbers by the equation

$$[\alpha_1, \ldots, \alpha_{n+1}] = [\alpha_1, [\alpha_2, \ldots, \alpha_{n+1}]].$$

For any real number $\alpha > 1$ define the sequence $\alpha_1, \alpha_2, \ldots, \alpha_n, \ldots$ of reals and the sequence $a_1, a_2, \ldots, a_n, \ldots$ of nonnegative integers as follows: $\alpha_1 = \alpha, a_n = \lfloor \alpha_n \rfloor$ = the greatest integer $\leq \alpha_n$, and

$$\alpha_{n+1} = \frac{1}{\alpha_n - a_n}$$

i.e $\alpha_n = [a_n, \alpha_{n+1}]$. The sequence $[a_1], [a_1, a_2], \ldots, [a_1, \ldots, a_n], \ldots$ defined as above from the given real number α is called **the continued fraction expansion** of α.

Remark 1: Notice that if $\alpha_n = a_n$ then α_{n+i} is undefined for all $i > 0$ Moreover, for all n,

$$\alpha_n = [a_n, \alpha_{n+1}] = [a_n, a_{n+1}, \alpha_{n+2}] = \cdots.$$

In particular,

$$\alpha = [a_1, \alpha_2] = [a_1, a_2, \alpha_3] = \cdots.$$

The continued fraction expansion $[a_1], [a_1, a_2], \ldots, [a_1, \ldots, a_n], \ldots$ of the real number α **breaks-up** if for some $n, \alpha_n = a_n$.

The following observation which is an immediate consequence of the above definitions and the Euclidean algorithm, will be useful in the sequel: if d is the divisor and r is the remainder in the Euclidean division $x = yd + r$, where $x > y > r > 0$ and $\gcd(x, y) = 1$ then

$$d = \left\lfloor \frac{x}{y} \right\rfloor \text{ and } \left[d, \frac{y}{r}\right] = \frac{x}{y}.$$

Theorem 1.21 *A real number > 1 is rational if and only if its continued fraction expansion breaks-up.*

Proof: Let $\alpha > 1$ be a real number with continued fraction expansion $[a_1, a_2], \ldots, [a_1, \ldots, a_n], \ldots$. If for some $n, a_n = \alpha_n$ then an easy computation shows that $\alpha = [a_1, \ldots, a_{n-1}, \alpha_n] = [a_1, \ldots, a_{n-1}, a_n]$ is rational.

Conversely, assume that $\alpha = a/b$ is rational. Use the Euclidean algorithm to define sequences

$$0 < r_n < r_{n-1} < \ldots < r_1 < r_0 = b < r_{-1} = a, d_1, d_2, \ldots, d_n, d_{n+1}$$

such that

$$a = d_1 b + r_1, b = d_2 r_1 + r_2,$$

$$r_1 = d_3 r_2 + r_3, \ldots, r_{n-2} = d_n r_{n-1} + r_n, r_{n-1} = d_{n+1} r_n.$$

It follows by induction on i that

$$\alpha_i = \frac{r_{i-2}}{r_{i-1}}, \ a_i = d_i, \text{ for } i = 1, \ldots, n.$$

Moreover,

$$\alpha_{n+1} = \frac{r_{n-1}}{r_n} = d_{n+1} = a_{n+1} \bullet$$

Let $[a_1, a_2], \ldots, [a_1, \ldots, a_n], \ldots$ be the continued fraction expansion of the real number $\alpha > 1$; define the sequences

$$A_{-1}, A_0, A_1, \ldots, A_n, \ldots, B_{-1}, B_0, B_1, \ldots, B_n, \ldots,$$

as follows:

$$A_{-1} = 0, A_0 = 1, B_{-1} = 1, B_0 = 0 \text{ and}$$

$$A_n = a_n A_{n-1} + A_{n-2}, B_n = a_n B_{n-1} + B_{n-2}.$$

The fraction A_n/B_n is called the n-th **convergent** of α.

The basic properties of the convergents can be found in the theorem below.

Theorem 1.22 *Let $\alpha > 1$ be a real number with continued fraction expansion $[a_1, a_2], \ldots, [a_1, \ldots, a_n], \ldots$ and convergents A_n/B_n. For any integer $n \geq 0$ the following hold*

(i) $A_n B_{n-1} - A_{n-1} B_n = (-1)^n$.

(ii) $\gcd(A_n, B_n) = 1$.

(iii) $A_0 < A_1 < \cdots < A_n < \cdots, \quad B_0 < B_1 < \cdots < B_n < \cdots$.

(iv) $\alpha = (A_n \alpha_{n+1} + A_{n-1})/(B_n \alpha_{n+1} + B_{n-1}), n \geq 1$.

(v) $|\alpha - A_n/B_n| < 1/B_n^2$, *if* α_{n+1} *is defined.*

(vi) $A_n/B_n - A_{n-1}/B_{n-1} = (-1)^n/(B_n B_{n-1}), n > 1$.

(vii) $A_n/B_n - A_{n-2}/B_{n-2} = a_n(-1)^{n-1}/(B_n B_{n-2}), n > 2$.

(viii) $A_{2n-1}/B_{2n-1} < A_{2n+1}/B_{2n+1} < \alpha < A_{2n}/B_{2n} < A_{2n-2}/B_{2n-2}$.

(ix) $\lim_{n \to \infty} A_n/B_n = \alpha$.

Proof: The proof of the theorem, although tedious, it is straightforward by induction on n and is left as an exercise to the reader. Notice that (ix) follows from (v) and the fact that the sequence B_n has exponential growth. In fact, an easy induction on n, using the definitions of A_n, B_n will show that $A_n, B_n \geq f_n \geq R^{n-2}$, where R is the golden mean. Hence, $n \leq 2 + \log_R B_n, 2 + \log_R A_n$, where $n > 1$ and the number of steps needed to compute A_n (respectively $B_n)isequaltoO($number of steps needed to compute f_n).•

A rational A/B is called a **Diophantine approximation** of the real number $\alpha > 1$ if and only if $B > 0$ and $\gcd(A, B) = 1$ and for all integers C, D with $D \leq B$ and $C/D \neq A/B$ the inequality $|A - B\alpha| < |C - D\alpha|$ holds. It is easy to see that if A/B is a Diophantine approximation of $\alpha > 1$, then for all integers C, D,

$$D \leq B \text{ and } C/D \neq A/B \Rightarrow \left|\frac{A}{B} - \alpha\right| < \left|\frac{C}{D} - \alpha\right|.$$

The following theorem will be essential in the study of the $1/p$ pseudorandom generator.

Theorem 1.23 *Let A/B be a rational and $\alpha > 1$ a real number such that $B > 0$ and $\gcd(A, B) = 1$. Then the following statements hold*

(i) A/B is a Diophantine approximation of $\alpha \Rightarrow A/B$ is a convergent of α.

(ii) $|\alpha - A/B| < 1/(2B^2) \Rightarrow A/B$ is a Diophantine approximation of α.

Proof: (i) First notice that

$$\frac{A_2}{B_2} < \frac{A_4}{B_4} < \ldots < \alpha < \ldots < \frac{A_3}{B_3} < \frac{A_1}{B_1}.$$

At first it will be shown that A/B is either a convergent or else lies between two convergents of α. Indeed, assume on the contrary

$$\frac{A_1}{B_1} < \frac{A}{B}.$$

Recall that $a_1/1 = A_1/B_1$. It follows that

$$\left|\frac{a_1}{1} - \alpha\right| < \left|\frac{A}{B} - \alpha\right| = \frac{|A - \alpha B|}{B} \leq |A - \alpha B| < \left|\frac{a_1}{1} - \alpha\right|,$$

which is a contradiction. Hence, $A_1/B_1 \geq A/B$. Next, assume on the contrary that

$$\frac{A_2}{B_2} > \frac{A}{B}.$$

Recall that $B_2 = a_2$. It follows that

$$\left|\frac{A}{B} - \alpha\right| > \left|\frac{A}{B} - \frac{A_2}{B_2}\right| \geq \frac{1}{B_2 B}.$$

Thus,

$$|A - \alpha B| > \frac{1}{B_2} = \frac{1}{a_2} \geq \frac{1}{\alpha_2} = |a_1 - \alpha| = |A_1 - \alpha B_1|,$$

which contradicts the definition of Diophantine approximation. It follows that

$$\frac{A_2}{B_2} \leq \frac{A}{B} \leq \frac{A_1}{B_1}.$$

Now it can be shown that A/B is a convergent. Indeed, assume on the contrary that A/B lies strictly between two convergents i.e.

$$\frac{A_{n+1}}{B_{n+1}} < \frac{A}{B} < \frac{A_{n-1}}{B_{n-1}}.$$

A contradiction will be derived by distinguishing two cases.

Case 1: n is odd (see figure 7).

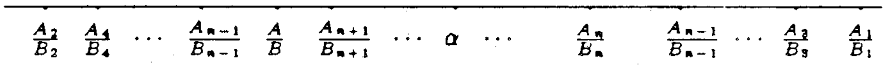


Figure 7: The Convergents of α

$$\left|\frac{A_n}{B_n} - \frac{A_{n-1}}{B_{n-1}}\right| = \frac{|(-1)^n|}{B_n B_{n-1}} =$$

$$\frac{1}{B_n B_{n-1}} > \left|\frac{A}{B} - \frac{A_{n-1}}{B_{n-1}}\right| \geq \frac{1}{B B_{n-1}}.$$

It follows that $B \geq B_n$. Moreover,

$$\left|\frac{A}{B} - \alpha\right| \geq \left|\frac{A}{B} - \frac{A_{n+1}}{B_{n+1}}\right| \geq \frac{1}{B B_{n+1}} \geq \frac{1}{(B_n \alpha_{n+1} + B_{n-1})B}$$

and

$$\left|\alpha - \frac{A_n}{B_n}\right| = \left|\frac{A_n \alpha_{n+1} + A_{n-1}}{B_n \alpha_{n+1} + B_{n-1}} - \frac{A_n}{B_n}\right| \leq \frac{1}{(B_n \alpha_{n+1} + B_{n-1})B_n}.$$

It follows that

$$|A_n - \alpha B_n| \leq \frac{1}{B_{n+1}} \leq |A - \alpha B| < |A_n - \alpha B_n|,$$

since $B_n \leq B$, and A/B is a Diophantine approximation of α, which is a contradiction.

Case 2: n is even.

This is omitted because it is similar to the proof of case 1. Hence the proof of (i) is complete.

(ii) Assume on the contrary that A/B is not a Diophantine approximation of α. This means that there exist integers C, D with $D \leq B$ and $C/D \neq A/B$ such that the following inequality holds:

$$|A - B\alpha| \geq |C - D\alpha| \tag{22}$$

In the proof below it will be assumed that $\alpha < A/B$. The case $\alpha > A/B$ is treated similarly. Notice that $|AD - CB| \geq 1$, and hence,

$$\left|\frac{A}{B} - \frac{C}{D}\right| \geq \frac{1}{BD}. \tag{23}$$

Case 1: $C/D < A/B < \alpha$

In this case one has

$$0 < \frac{A}{B} - \frac{C}{D} < \alpha - \frac{C}{D} = \frac{D\alpha - C}{D} \leq \frac{B\alpha - A}{D} =$$
$$\left(\alpha - \frac{A}{B}\right) \cdot \frac{B}{D} < \frac{B}{D} \cdot \frac{1}{2B^2} = \frac{1}{2BD},$$

which contradicts equation (23).

Case 2: $A/B < C/D < \alpha$

In this case one uses $D \leq B$ to obtain

$$0 < \frac{C}{D} - \frac{A}{B} < \alpha - \frac{A}{B} < \frac{1}{2B^2} \leq \frac{1}{2BD},$$

which contradicts equation (23).

Case 3: $\alpha < C/D$

$$0 < \frac{C}{D} - \alpha = \frac{C - D\alpha}{D} \leq \frac{A - B\alpha}{D} = \frac{B}{D}\left(\alpha - \frac{A}{B}\right).$$

It follows that $A/B \leq \alpha < C/D$. Consequently,

$$0 < \frac{C}{D} - \frac{A}{B} = \frac{C}{D} - \alpha + \alpha - \frac{A}{B} \leq \frac{B}{D}\left(\alpha - \frac{A}{B}\right) + \left(\alpha - \frac{A}{B}\right)$$
$$= \left(1 + \frac{B}{D}\right)\left(\alpha - \frac{A}{B}\right) = \frac{D+B}{D}\left(\alpha - \frac{A}{B}\right) < \frac{D+B}{D}\frac{1}{2B^2}$$
$$\leq \frac{2B}{D}\frac{1}{2B^2} = \frac{1}{BD},$$

which contradicts equation (23). This completes the proof of the theorem. •

EXERCISES

1: Give the details of the proof of Theorem 1.22.

2: What is the limit of the sequence $[1,1], [1,1,1], \ldots$?

1.17 BIBLIOGRAPHICAL REMARKS

The approach taken in this section is to provide a self-contained introduction to all the material on Number Theory, necessary to understand the results on the security of Pseudo-random Generators (section 3) and Public-key Cryptosystems (section 4). There are numerous nice introductory or advanced books in Number Theory. Such books include, [W] [Vi], [Lev], [NZ], [Kr], [Scha]. Since the present chapter is intended to empasize techniques useful to understanding the security of Public Key Cryptosystems, the material presented is combined with the study of the complexity of certain problems in Number Theory. A more exhaustive study of such Algorithms can be found in [Kn], as well as in [An].

For more information on Lamé's theorem (1.2) see [Kn] (page 313.) The result on (k, n) threshold schemes is originally due to [Sham2]. However, the proof of theorem 1.7 given here follows the presentation of [Mi]. The characterization, in theorem 1.9, of those m for which Z_m^* is cyclic, as well as the Law of Quadratic Reciprocity (theorem 1.13) was first proved by Gauss in [Ga]. The interested reader can find a lot of information on Artin's conjecture in [Scha] (pp. 80 - 83, 222 - 225) as well as in [Has] (pp. 74 - 75).

The Law of Quadratic Reciprocity is very useful in solving Diophantine equations. More than 150 proofs of this theorem have so far appeared in the literature, including 8 given by Gauss himself. The present simple proof appears in [Ge]. Some interesting proofs and comments on the Law of Quadratic Reciprocity can also be found in [Pi].

A procedure for finding square roots modulo a prime number first appeared in [Ber]. The present proof of Theorem 1.15 is from [AMM]. Computing the index of a number x modulo a prime number p is in general an open problem. The algorithm given in theorem 1.18 is from [PH].

A complete proof of theorem 1.19 can be found [La] (part 6) or [Prac] (pp. 131 -139 .) The proof of the weaker version of the prime number theorem 1.20 presented in subsection 1.15 is due to Chebyshev and follows closely the presentation of Zagier in ([Z].) For more information the reader can consult the beatiful expository articles: Prime Numbers, by Mardzanisvili and Postnikov in [Ma] and Die ersten 50 Millionen Primzahlen, by Zagier in [Z].

2 PROBABILITY THEORY

2.1 INTRODUCTION

The present section introduces the reader to the fundamental concepts of probability theory. The development of the concepts is limited to the material necessary to understand the proofs in the sections on pseudo random generators and public key cryptosystems.

Subsection 2.2 includes all the necessary introductory notions i.e. σ-algebra, probability space, product and sum of events. The notion of random variable is developed in subsection 2.3. Further, this subsection includes the fundamental theorems for computing expectations and variances of random variables. The binomial distribution, which is studied in subsection 2.4, will be the only probability distribution to be exhibited in the present monograph. Chebyshev's law of large numbers is proved in subsection 2.5. The strengthening of the weak law of large numbers, proved in subsection 2.6, will be very useful in the development of the general theory of the security of pseudo random generators and public key cryptosystems. An introduction to the Monte Carlo method, is exhibited in subsection 2.7.

2.2 BASIC NOTIONS

A $\sigma-$algebra A on a nonempty set Ω is a nonempty set of subsets of Ω which satisfies the following three properties:

1. $\Omega \in A$.
2. If $E \in A$ then $\Omega - E \in A$.
3. If $\{E_n : n \geq 0\} \subseteq A$ then $(\bigcup_{n \geq 0} E_n) \in A$.

Example 2.1 *The set* $\{\Omega, \emptyset\}$ *is a* $\sigma-$ *algebra.*

Example 2.2 *The set of all subsets of a nonempty set* Ω *is a* $\sigma-$ *algebra.*

A probability space is a triple (Ω, A, Pr), where

1. Ω is a nonempty set,
2. A is a $\sigma-$ algebra on the set Ω, and
3. Pr is an experiment on the $\sigma-$ algebra A i.e. Pr is a function $Pr : A \longrightarrow [0,1]$, with domain the $\sigma-$ algebra A and range a subset of the unit interval $[0,1]$ such that
   (a) $Pr[\Omega] = 1$ and $Pr[\emptyset] = 0$,

(b) For any family $\{E_n : n \geq 0\} \subseteq A$ of pairwise disjoint subsets of Ω,

$$Pr\left[\bigcup_{n\geq 0} E_n\right] = \sum_{n=0}^{\infty} Pr[E_n].$$

The subsets of Ω are called **events**, while the subsets of Ω which belong to the $\sigma-$ algebra A are called **observed events**; the elements $\omega \in \Omega$ are called the **possible outcomes** of the experiment Pr. An event E is called **certain** (respectively **impossible**) if $Pr[E] = 1$ (respectively if $Pr[E] = 0$). The set Ω is called the **sample space** of the experiment.

Example 2.3 *The experiment determined by the flipping of a fair coin consists of the sample space $\Omega = \{H, T\}$, where $H =$ Head and $T =$ Tail, the $\sigma-$ algebra A of all subsets of Ω and the experiment Pr which satisfies:*

$$Pr[\{H,T\}] = 1,\ Pr[\{H\}] = Pr[\{T\}] = 1/2,\ Pr[\emptyset] = 0.$$

Example 2.4 *The experiment determined by the tossing of two fair dice consists of the sample space $\Omega = \{(i,j) : 1 \leq i, j \leq 6\}$, the $\sigma-$ algebra of all subsets of Ω and the experiment Pr which satisfies*

$$Pr[E] = |E|/36, \text{ for all events } E.$$

Example 2.5 *In the above experiment one can also consider the following $\sigma-$ algebra of observed events: an event $E \in A$ if and only if for all (i,j), if $(i,j) \in E$ then $(j,i) \in E$.*

Corresponding to the set theoretic boolean operations of union, intersection and difference of sets, one can define respectively the sum, the product and the difference of events. Hence, given two events E, F one defines the **sum** (respectively **product**, **difference**) of the events E, F to be the event $E \cup F$ (respectively $E \cap F$, $E - F$).

Given a probability space (Ω, A, Pr) and an observed event K, such that $Pr[K] > 0$ the **conditional probability space** with respect to K, is the triple (Ω, A, Pr_K), where the new experiment Pr_K is defined by

$$Pr_K[E] = \frac{Pr[E \cap K]}{Pr[K]}.$$

In addition, the notation $Pr[E|K]$ will also be used as identical to $Pr_K[E]$.

Given a probability space (Ω, A, Pr) and two observed events E, F the following three rules are very useful for the study of probability theory and can be derived easily from the above defining properties of Pr:

1. **Difference Rule:** If $E \subseteq F$ then $Pr[F - E] = Pr[F] - Pr[E]$.

2. **Sum Rule:** $Pr[E \cup F] = Pr[E] + Pr[F] - Pr[E \cap F]$.

3. **Product Rule:** If $Pr[F] > 0$ then $Pr[E \cap F] = Pr_F[E] \cdot Pr[F]$.

Events E, F are called **indepedent** with respect to the probability space (Ω, A, Pr) if $Pr[E \cap F] = Pr[E] \cdot Pr[F]$.

EXERCISES

1: Show that every $\sigma-$ algebra is closed under countable intersections.
2: Show that the empty set is a member of every $\sigma-$ algebra.
3: Prove in detail the Difference, Sum and Product rules.

2.3 RANDOM VARIABLES

Let R be the set of all real numbers. A **random variable** on the probability space (Ω, A, Pr) is a real valued function $X : \Omega \longrightarrow R$ such that for any open set I of real numbers,

$$X^{-1}[I] = \{\omega \in \Omega : X(\omega) \in I\} \in A.$$

A **vector random variable** on the probability space (Ω, A, Pr) is a real vector valued function $X : \Omega \longrightarrow R^n$ such that for any open subset I of the set of n tuples of real numbers,

$$X^{-1}[I] = \{\omega \in \Omega : X(\omega) \in I\} \in A.$$

It is easy to see that if $X_1, \ldots, X_n$ are random variables on Ω then the function $(X_1, \ldots, X_n)$ is a vector random variable on Ω.

For any random variable X and any real number k, let $X = k$ denote the event $\{\omega \in \Omega : X(\omega) = k\}$. A random variable is **finite** (respectively **discrete**) if it takes on only a finite (respectively countable) number of values.

For any random variables $X_1, \ldots, X_n$ and any function $f : R^n \longrightarrow R$, let $f(X_1, \ldots, X_n)$ denote the composition of the functions $f, (X_1, \ldots, X_n)$ i.e. for all $\omega \in \Omega$, $f(X_1, \ldots, X_n)(\omega) = f(X_1(\omega), \ldots, X_n(\omega))$. If f is continuous then the inverse image under f of any open set is open. Hence, if all the $X_1, \ldots, X_n$ are random variables and f is continuous then $f(X_1, \ldots, X_n)$ is also a random variable. In particular, if X, Y are random variables so are $X+Y$, $X \cdot Y$, $\exp(X)$, etc.

For any random variable X the **(probability) mass** or **(probability) distribution function** of the random variable X is the function p_X defined for real numbers k by

$$p_X(k) = Pr[X = k].$$

Hence, if the random variable X takes on only the values $x_1, \ldots, x_n, \ldots$ then its corresponding probability distribution function p_X will take on only the values $p_X(x_1), \ldots, p_X(x_n), \ldots$.

For simplicity from now on and for the rest of this section all the random variables used will be discrete and bounded i.e. there exists a real number B such that for all $\omega \in \Omega, |X(\omega)| \leq B$. Moreover, the probability space used in each particular case will not always be explicitely mentioned, unless there is a cause of confusion.

Let X be a random variable which takes on only the values $x_1, \ldots, x_n, \ldots$ and let $p_X(x_1) = p_1, \ldots, p_X(x_n) = p_n, \ldots$ be the corresponding values of its distribution function p_X. (Here it is asssumed that $x_i \neq x_j$, for all $i \neq j$.) The expectation of the random variable X, abbreviated $E[X]$, is defined by

$$E[X] = \sum_{n=0}^{\infty} x_n \cdot p_n. \tag{1}$$

The variance of the random variable X, abbreviated $Var[X]$, is defined by

$$Var[X] = E\left[(X - E[X])^2\right] = \sum_{n=0}^{\infty} (x_n - E[X])^2 \cdot p_n. \tag{2}$$

The square root of the variance of X is called **standard deviation** of X and is denoted by $D[X]$ i.e.

$$D[X] = \sqrt{Var[X]}. \tag{3}$$

Example 2.6 *For the exact fitting of a certain part of a precision instrument it is required to make* $1, 2, \ldots, 9$ *trials. The number of trials necessary to achieve exact fitting of the part is a random variable, denoted by X. The behavior of the probability distribution function of the random variable X can be best represented in the graph of figure 1.*

Example 2.7 *Consider the random variable X given in example 2.6. Then the expectation of X is given by*

$$E[X] = 1 \cdot .1 + 2 \cdot .15 + 3 \cdot .25 + 4 \cdot .3 + 5 \cdot .2 = 3.35$$

Thus, the number of trials necessary to achieve exact fitting will on the average be 3.35 i.e. the exact fitting of 100 *instruments will on the average require* 335 *trials.*

Example 2.8 *Consider the random variable X given in example 2.6. Then the variance of X is given by*

$$Var[X] = (2.35)^2 \cdot .1 + (1.35)^2 \cdot .15 + (.35)^2 \cdot .25 + (.65)^2 \cdot .3 + (1.65)^2 \cdot .2 = 1.795$$

The standard deviation of X will be

$$D[X] = \sqrt{1.795} = 1.34.$$

Thus, the standard deviation $D[X]$ gives the magnitude of the oscillations of X around the expectation $E[X]$.

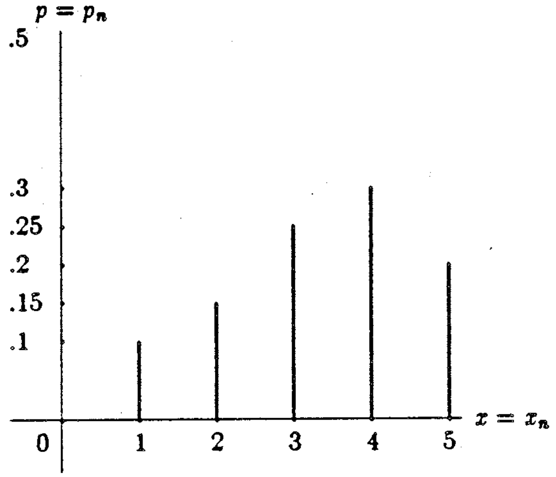


Figure 1: Graph of p_X

The covariance of two random variables X, Y, abbreviated $Cov[X, Y]$, is defined by

$$Cov[X, Y] = E[(X - E[X]) \cdot (Y - E[Y])]. \tag{4}$$

Remark: Since the random variable X is bounded the infinite series in definition (1) is absolutely convergent. Hence, the definition of $E[X]$ does not depend on the given enumeration of the values taken on by X.

The following two theorems will be useful in the sequel.

Theorem 2.1 (**The Expectation Theorem**) *Let X, Y be two random variables. Let the random variable X take on only the values $x_0, \ldots, x_n, \ldots$ and let $p_X(x_0) = p_0, \ldots, p_X(x_n) = p_n, \ldots$ be the corresponding values of its distribution function p_X. Then*

1. *$E[a \cdot X + b \cdot Y] = a \cdot E[X] + b \cdot E[Y]$, where a, b are reals.*
2. *If X, Y are indepedent then $E[X \cdot Y] = E[X] \cdot E[Y]$.*
3. *For any continuous function $f : R \longrightarrow R$, $E[f(X)] = \sum_{n=0}^{\infty} f(x_n) \cdot p_n$.*

Proof of 1: Only the proof of $E[X + Y] = E[X] + E[Y]$ will be given; the rest will be left as an exercise to the reader. Suppose that the random variable Y takes on only the values $y_0, \ldots, y_m, \ldots$ and let $p_Y(y_0) = q_0, \ldots, p_Y(y_m) =$

$q_m, \ldots$ be the corresponding values of its distribution function p_Y. Let $Z = X + Y$ and suppose that $z_0, \ldots, z_k, \ldots$ are the distinct values taken on by the random variable Z. Finally put $p_{n,m} = Pr[X = x_n \text{ and } Y = y_m]$.

From the definition of expectation,

$$E[X + Y] = E[Z] = \sum_{k \geq 0} z_k \cdot Pr[Z = z_k].$$

However, for all $k \geq 1$

$$z_k \cdot Pr[Z = z_k] = \sum_{x_n + y_m = z_k} (x_n + y_m) \cdot p_{n,m}.$$

It is then clear from the last two equations that

$$E[X + Y] = \sum_{n,m \geq 0} (x_n + y_m) \cdot p_{n,m} =$$

$$= \sum_{n \geq 0} x_n \cdot \left(\sum_{m \geq 0} p_{n,m} \right) + \sum_{m \geq 0} y_m \cdot \left(\sum_{n \geq 0} p_{n,m} \right).$$

On the other hand it is obvious that

$$p_n = \sum_{m \geq 0} p_{n,m}, \quad q_m = \sum_{n \geq 0} p_{n,m}.$$

The result now follows immediately from the definition of expectation and the last two equations.

Proof of 2: The notation of the proof of part (1) will be used in the proof of part (2) as well. Since the random variables X, Y are indepedent, it is clear that for all $n, m \geq 0$,

$$p_{n,m} = p_n \cdot q_m.$$

On the other hand using the definition of expectation, and arguing as in the proof of part 1 it can be shown that

$$E[X \cdot Y] = \sum_{n,m \geq 0} x_n \cdot y_m \cdot p_{n,m} = \sum_{n,m \geq 0} x_n \cdot y_m \cdot p_n \cdot q_m.$$

It follows that

$$E[X \cdot Y] = \left(\sum_{n \geq 0} x_n \cdot p_n \right) \cdot \left(\sum_{m \geq 0} y_m \cdot q_m \right) = E[X] \cdot E[Y],$$

which completes the proof of part (2).

Proof of 3:

Let $z_0, \ldots, z_k, \ldots$ be the distinct values taken on by the random variable $f(X)$. For each $k \geq 0$, let $I_k = \{n \geq 0 : f(x_n) = z_k\}$. Clearly, the event $(f(X) = z_k)$ occurs if and only if for some $n \in I_k$, the event $(X = x_n)$ occurs. Hence, the distribution function of $f(X)$ is given by

$$p_{f(X)}(z_k) = \sum_{n \in I_k} Pr[X = x_n] = \sum_{n \in I_k} p_n.$$

It follows from the definition of expectation that

$$E[f(X)] = \sum_{k \geq 0} z_k \cdot p_{f(X)}(z_k) =$$

$$\sum_{k \geq 0} z_k \cdot \left(\sum_{n \in I_k} p_n \right) = \sum_{n \geq 0} f(x_n) \cdot p_n.$$

This completes the proof of part (3) and hence of the theorem •

Theorem 2.2 (**The Variance Theorem**) *Let X, Y be two random variables. Then for all real numbers a, b,*

1. $Var[a \cdot X + b \cdot Y] = a^2 \cdot Var[X] + b^2 \cdot Var[Y] + 2ab \cdot Cov[X, Y]$.
2. *If X, Y are indepedent then* $Var[X + Y] = Var[X] + Var[Y]$.

Proof of 1: Only the proof of $Var[X+Y] = Var[X] + Var[Y] + 2Cov[X, Y]$ will be given; the rest will be left as an exercise to the reader. Let $E[X] = \mu, E[Y] = \nu$. Using the definition of the variance and the expectation theorem one can show that

$$Var[X + Y] = E[(X + Y - \mu - \nu)^2] =$$
$$E[(X - \mu)^2 + (Y - \nu)^2 + 2 \cdot (X - \mu) \cdot (Y - \nu)] =$$
$$= Var[X] + Var[Y] + 2Cov[X, Y].$$

Proof of 2: Using the definition of the covariance and the expectation theorem it is easy to see that

$$Cov[X, Y] = E[(X - E[X]) \cdot (Y - E[Y])] = E[X \cdot Y] - \mu \cdot \nu.$$

But the right side of the above equality is 0 because the random variables X, Y are indepedent. This completes the proof of (2) and hence of the theorem •

EXERCISES

1: Let X be a random variable. Show that for all real numbers a, b,

1. $E[a \cdot X + b] = a \cdot E[X] + b$
2. $Var[a \cdot X + b] = a^2 \cdot Var[X]$
3. $D[a \cdot X + b] = |a| \cdot D[X]$

2.4 THE BINOMIAL DISTRIBUTION

A random variable X which takes on only the values $0, 1, \ldots, n$ is said to have the **binomial distribution** with parameters n, p if and only if for any $0 \leq k \leq n$,

$$Pr[X = k] = \binom{n}{k} p^k (1-p)^{n-k}.$$

For any event E in a given probability space and any integer n let the random variable $B_n(E)$ denote the number of occurrences of the event E in n indepedent from one another trials. The n-**th relative frequency** of the event E, abbreviated $F_n(E)$, is the random variable

$$F_n(E) = \frac{B_n(E)}{n}.$$

The following theorem will be very useful in the sequel.

Theorem 2.3 (**The Binomial Distribution Theorem**) *For any event E in a given probability space such that $p = Pr[E]$ and any integer $n \geq 0$ the random variable $B_n(E)$ has the binomial distribution with parameters n, p. Moreover,*

1. *$E[B_n(E)] = n \cdot p, E[F_n(E)] = p$, and*
2. *$Var[B_n(E)] = n \cdot p \cdot (1-p), Var[F_n(E)] = (1/n) \cdot p \cdot (1-p)$.*

Proof: To see that $B_n(E)$ satisfies the binomial distribution, notice that the event $(B_n(E) = k)$ occurs exactly when the event E occurs k times and the event $(\Omega - E)$ occurs $(n-k)$ times. Each such event-sequence occurs with probability $p^k(1-p)^{n-k}$. Hence, the first part of the theorem follows from the fact that there exist exactly $\binom{n}{k}$ such sequences.

Proof of part 1: Using the definition of expectation, and trivial algebraic manipulations one obtains,

$$E[B_n(E)] = \sum_{k=0}^{n} k \cdot \binom{n}{k} \cdot p^k \cdot (1-p)^{n-k} =$$

$$= np \cdot \sum_{k=1}^{n} \binom{n-1}{k-1} \cdot p^{k-1} \cdot (1-p)^{n-k} =$$

$$= np \cdot (p + (1-p))^{n-1} = np.$$

The computation of the quantity $E[F_n(E)]$ follows easily from the expectation theorem.

Proof of part 2: Using the definition of variance, the result in part (1) and trivial algebraic manipulations one obtains,

$$Var[B_n(E)] = \sum_{k=0}^{n}(k-np)^2 \cdot \binom{n}{k} \cdot p^k \cdot (1-p)^{n-k} =$$

$$= \sum_{k=0}^{n} k^2 \cdot \binom{n}{k} \cdot p^k \cdot (1-p)^{n-k} +$$

$$n^2p^2 \cdot \sum_{k=0}^{n} \binom{n}{k} \cdot p^k \cdot (1-p)^{n-k} - 2np \cdot \sum_{k=0}^{n} k \cdot \binom{n}{k} \cdot p^k \cdot (1-p)^{n-k} =$$

$$-n^2p^2 + \sum_{k=0}^{n} k^2 \cdot \binom{n}{k} \cdot p^k \cdot (1-p)^{n-k} =$$

$$-n^2p^2 + \sum_{k=0}^{n} k(k-1) \cdot \binom{n}{k} \cdot p^k \cdot (1-p)^{n-k} + \sum_{k=0}^{n} k \cdot \binom{n}{k} \cdot p^k \cdot (1-p)^{n-k} =$$

$$-n^2p^2 + n(n-1)p^2 + np = np(1-p).$$

The computation of the quantity $Var[F_n(E)]$ follows easily from the variance theorem •

2.5 CHEBYSHEV'S LAW OF LARGE NUMBERS

In general, a law of large numbers gives a set of sufficient conditions to enable the arithmetic mean of a sequence of random variables to tend to a fixed constant number with high probability, when the number of summands is increasing. The first such law to be proved is based on the following inequality.

Lemma 2.1 (**Chebyshev's Inequality**) *For any random variable X, and any real number $\epsilon > 0$,*

$$Pr[|X - E[X]| \geq \epsilon] \leq \frac{Var[X]}{\epsilon^2}.$$

Proof: Let the random variable X take on only the values $x_0, \ldots, x_n, \ldots$ and let $p_X(x_0) = p_0, \ldots, p_X(x_n) = p_n, \ldots$ be the corresponding values of its distribution function p_X. Put $\mu = E[X]$. Then

$$Var[X] = \sum_{n \geq 0}(x_n - \mu)^2 \cdot p_n.$$

It follows that

$$Var[X] \geq \sum_{|x_n - \mu| \geq \epsilon} (x_n - \mu)^2 \cdot p_n$$

$$\geq \sum_{|x_n-\mu|\geq\epsilon} \epsilon^2 \cdot p_n = \epsilon^2 \cdot \sum_{|x_n-\mu|\geq\epsilon} p_n =$$

$$\epsilon^2 \cdot \sum_{|x_n-\mu|\geq\epsilon} Pr[X = x_n] = \epsilon^2 \cdot Pr[|X - \mu| \geq \epsilon] \bullet$$

As an immediate application one obtains the following

Theorem 2.4 *(***Chebyshev's Law of Large Numbers***) Let $X_1, \ldots, X_n$ be indepedent random variables and let the random variable X denote their arithmetic mean i.e.*

$$X = \frac{X_1 + \cdots + X_n}{n}.$$

Then for any $\epsilon > 0$,

$$Pr[|X - E[X]| \geq \epsilon] \leq \frac{\sum_{i=1}^{n} Var[X_i]}{n^2 \cdot \epsilon^2}.$$

Moreover,

$$Pr[|X - E[X]| \geq \epsilon] \leq \frac{\max_{1\leq i\leq n} Var[X_i]}{n \cdot \epsilon^2}.$$

Proof: The proof is immediate from the variance theorem and Chebyshev's inequality •

The next theorem, which is an immediate consequence of Chebyshev's inequality, will be applied frequently in sections 3 through 5.

Theorem 2.5 *(***Weak Law of Large Numbers, Bernoulli***) Suppose that the event E occurs with probability p. Then for any integer $n \geq 1$ and any $\epsilon > 0$,*

$$Pr[|F_n(E) - p| \geq \epsilon] \leq \frac{p \cdot (1-p)}{n \cdot \epsilon^2} \leq \frac{1}{4n \cdot \epsilon^2}.$$

Proof: This is immediate from the binomial distribution theorem, Chebyshev's inequality and the fact that $4p(1-p) \leq 1$ •

2.6 BERNSHTEIN'S LAW OF LARGE NUMBERS

The Bernoulli estimate for the weak law of large numbers given in theorem (2.5) can be substantially improved. The improvement is based on the following inequality.

Lemma 2.2 *(***Markov's Inequality***) For any random variable X, any real number $\epsilon > 0$, and any nondecreasing continuous function $f : R \longrightarrow R$, which takes on only positive values,*

$$Pr[X \geq \epsilon] \leq \frac{E[f(X)]}{f(\epsilon)}.$$

In particular, if $E[X] > 0$ then

$$Pr[X \geq \epsilon \cdot E[X]] \leq \frac{1}{\epsilon}.$$

Proof: Let the random variable X take on only the values $x_0, \ldots, x_n, \ldots$ and let $p_X(x_0) = p_0, \ldots, p_X(x_n) = p_n, \ldots$ be the corresponding values of its distribution function p_X. From the expectation theorem,

$$E[f(X)] = \sum_{n=0}^{\infty} f(x_n) \cdot p_n \geq$$

$$\sum_{x_n \geq \epsilon} f(x_n) \cdot p_n \geq \sum_{x_n \geq \epsilon} f(\epsilon) \cdot p_n =$$

$$f(\epsilon) \cdot \sum_{x_n \geq \epsilon} p_n = f(\epsilon) \cdot Pr[X \geq \epsilon].$$

This completes the first part of the lemma. To prove the second part, apply the first part to the identity function and use $\epsilon' = \epsilon \cdot E[X]$•

As an immediate consequence of the second part of Markov's inequality, with $X' = e^{\epsilon(X - E[X])}$, $\epsilon' = e^t$, one obtains that for all t,

$$Pr\left[e^{\epsilon(X-E[X])} \geq E\left[e^{\epsilon(X-E[X])}\right] e^t\right] \leq e^{-t}. \tag{5}$$

Clearly inequality (5) is equivalent

$$Pr\left[X \geq E[X] + \frac{t + \log E\left[e^{\epsilon(X-E[X])}\right]}{\epsilon}\right] \leq e^{-t}. \tag{6}$$

The next lemma constitutes the major step in proving Bernshtein's sharpening of the weak law of large numbers.

Lemma 2.3 (**Bernshtein**) *Let $X_1, \ldots, X_n$ be a sequence of indepedent random variables with zero expectations and which are bounded by the constant K. If $X = X_1 + \cdots X_n$ then*

$$Pr\left[e^{\epsilon \cdot X}\right] \leq \exp\left[\frac{\epsilon^2 \cdot Var[X]}{2}\left(1 + \frac{\epsilon K}{3} \cdot e^{\epsilon K}\right)\right].$$

Proof: It is clear from the expectation theorem that

$$E\left[e^{\epsilon \cdot X}\right] = \prod_{i=1}^{n} E\left[e^{\epsilon \cdot X_i}\right]. \tag{7}$$

Hence, using the variance theorem and equation (7) it can be assumed without loss of generality that $n = 1, X = X_1$. Let $V = Var[X]$. It follows that

$$E\left[e^{\epsilon X}\right] = E\left[\sum_{k=o}^{\infty}\frac{\epsilon^k}{k!}X^k\right] = \sum_{k=0}^{\infty}\frac{\epsilon^k}{k!}E\left[X^k\right] =$$

$$1 + \epsilon E[X] + \frac{\epsilon^2}{2}E\left[X^2\right] + \sum_{k=3}^{\infty}\frac{\epsilon^k}{k!}E\left[X^k\right] \leq$$

$$1 + \frac{\epsilon^2}{2}V + \sum_{k=3}^{\infty}\frac{\epsilon^k}{k!}E\left[X^2\right]K^{k-2} =$$

$$1 + \frac{\epsilon^2}{2}V + \sum_{k=3}^{\infty}\frac{\epsilon^k}{k!}VK^{k-2} =$$

$$1 + \frac{\epsilon^2}{2}V + \frac{\epsilon^2}{6}V\sum_{k=3}^{\infty}\frac{(\epsilon K)^{k-2}}{(k-3)!} =$$

$$1 + \epsilon^2 V\left[\frac{1}{2} + \frac{\epsilon K e^{\epsilon K}}{6}\right]. \tag{8}$$

Using the inequality $1 + u < e^u$ and equation (8) one obtains that

$$E\left[e^{\epsilon X}\right] \leq \exp\left[\epsilon^2 V\left(\frac{1}{2} + \frac{\epsilon K e^{\epsilon K}}{6}\right)\right].$$

Let $X_1, \ldots, X_n$ be a sequence of indepedent random variables with zero expectations and which are bounded by the constant K. Put $X = X_1 + \cdots X_n$. Using inequality (6) to $X, -X$, applying lemma 2.3 to the random variables $X_1, \ldots, X_n$ and $-X_1, \ldots, -X_n$, and using the fact that $E[X] = 0$ it follows that for all t,

$$Pr\left[|X| \geq \frac{t + \epsilon^2 Var[X]\left(\frac{1}{2} + \frac{\epsilon K e^{\epsilon K}}{6}\right)}{\epsilon}\right] \leq 2 \cdot e^{-t}. \tag{9}$$

Putting

$$D = D[X], \epsilon = \frac{\sqrt{2t}}{D}, \lambda = \sqrt{2t},$$

inequality (9) becomes

$$Pr\left[|X| \geq \lambda D\left(1 + \frac{\lambda K}{6D}e^{\frac{\lambda K}{D}}\right)\right] \leq 2 \cdot e^{-\frac{\lambda^2}{2}}. \tag{10}$$

Assuming that $\frac{\lambda K}{D} \leq 1$, one obtains that

$$e^{\frac{\lambda K}{D}} \leq e < 3,$$

Putting

$$\mu = \lambda \cdot \left(1 + \frac{\lambda K}{2D}\right),$$

and using the fact that $\mu \geq \lambda$ inequality (10) becomes

$$Pr[|X| \geq \mu D] \leq 2 \cdot e^{-\frac{\lambda^2}{2}} \leq \exp\left[-\frac{\mu^2}{2\left(1 + \frac{\mu K}{2D}\right)^2}\right]. \tag{11}$$

To sum up, it has been shown that

Lemma 2.4 *Let $X_1, \ldots, X_n$ be a sequence of indepedent random variables and let K be a constant such that for all i, $|X_i - E[X_i]| \leq K$. If $X = X_1 + \cdots X_n$ then for all $0 < \mu \leq \frac{D}{K}$,*

$$Pr[|X - E[X]| \geq \mu D[X]] \leq 2 \cdot e^{-\frac{\lambda^2}{2}} \leq 2 \cdot \exp\left[-\frac{\mu^2}{2\left(1 + \frac{\mu K}{2D[X]}\right)^2}\right].$$

Using the last lemma and the binomial distribution theorem one easily obtains that

Theorem 2.6 (**Bernshtein's Law of Large Numbers**) *Suppose that the event E occurs with probability $0 < p < 1$. Then for any $n \geq 1$ and any $0 < \epsilon \leq p(1-p)$,*

$$Pr[|F_n[E] - p| \geq \epsilon] \leq 2 \cdot \exp\left[\frac{-n\epsilon^2}{2p(1-p)\left(1 + \frac{\epsilon}{2p(1-p)}\right)^2}\right] \leq 2 \cdot \exp\left[\frac{-n\epsilon^2}{4p(1-p)}\right].$$

EXERCISES

1: Derive theorem 2.6 from lemma 2.4 . **Hint:** apply lemma 2.4 to the random variable $X = B_n(E)$, to $K = 1$ and $\mu = (n\epsilon)/D$.

2.7 THE MONTE CARLO METHOD

There are many computational problems whose solution via deterministic procedures is cumbersome. For such problems it was observed that statistical sampling methods can approximate the solution much faster than numerical methods based on classical analysis. An example of such a problem is the computation of π, the area of the unit circle.

Buffon's Needle Problem: Suppose that parallel lines are drawn on the floor at a distance d from one another. Let a needle of length ℓ less than d be thrown at random on the floor. What is the probability that the needle will touch one of the parallel lines?

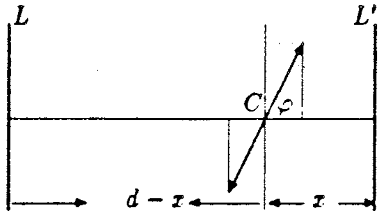


Figure 2: Buffon's Needle

Let the position of the needle be as in figure 2. Suppose that C is the center of the needle, x (respectively $d-x$) is the dictance of the center from the line L (respectively L') and φ the angle between the needle and the line perpendicular to L (see figure 2). For the sake of the argument that follows it will be assumed that the angle φ and the distance x are distributed uniformly over the range $-\pi/2 \le \varphi \le \pi/2$ and $0 \le x \le d$ respectively. It is apparent that the position of the needle is uniquely determined from the pairs of coordinates (x, φ), where $0 \le x \le d$ and $-\pi/2 \le \varphi \le \pi/2$. It is also clear from figure 2 that the needle will not touch any of the lines L, L' if and only if

$$-\frac{\pi}{2} \le \varphi \le \frac{\pi}{2}, \quad \frac{\ell}{2}\cdot\cos\varphi < x < d - \frac{\ell}{2}\cdot\cos\varphi \tag{12}$$

Let Ω be the set of pairs (x, φ) satisfying equation (12). Hence, the probability that the needle will not touch any of the parallel lines is

$$\frac{\text{area}(\Omega)}{d\cdot\pi} = \frac{d\cdot\pi - 2\cdot\ell}{d\cdot\pi} = 1 - \frac{2\cdot\ell}{d\cdot\pi}, \tag{13}$$

where the area of Ω is computed via

$$\text{area}(\Omega) = \int_{-\pi/2}^{\pi/2} d\varphi(d - \ell\cdot\cos\varphi)$$

It follows from equation (13) that the probability p that the needle will touch one of the lines is

$$p = \frac{2\cdot\ell}{d\cdot\pi}. \tag{14}$$

Equation (14) and the weak law of large numbers can be used as the basis for an experimental evaluation of π. Indeed, assume for simplicity that $d = 2$ and $\ell = 1$. Then $p = 1/\pi$. Consider an experiment in which the needle is thrown indepedently n times, and let the random variable X_i be equal to 1 if the needle intersects a line on the i-th throw and 0 otherwise. It follows from theorem 2.5 that for any $\epsilon > 0$,

$$Pr\left[\left|\frac{X_1 + \cdots X_n}{n} - \frac{1}{\pi}\right| \geq \epsilon\right] \leq \frac{1}{4n \cdot \epsilon^2}. \tag{15}$$

Hence, with high probability (the lower bound $1 - 1/(4n\epsilon^2)$ on the probability is determined from inequality 15),

$$\pi \approx \frac{n}{X_1 + \cdots X_n}.$$

In general, a **Monte Carlo method** is a statistical sampling method that can be used to approximate the solution of a certain problem. The computation necessary to find the solution is called a **Monte Carlo computation.** Although a problem might admit more than one Monte Carlo solution, there exist problems for which no Monte Carlo solution is known.

In sections 3, 4 and 5 several Monte Carlo computations will be included in the construction of circuits. As described above these computations will in fact be statistical sampling techniques which will enable the construction of polynomial size circuits solving the corresponding problems.

It should also be pointed out that an essential step in applying the Monte Carlo method for the solution of a certain problem is the ability to do random sampling. However, due to apparent technical limitations it would be unrealistic to hope that one could produce via an unbiased execution of an experiment a perfectly random sampling. Thus, one is led to replace the notion of random with that of pseudo random. Details on this last concept will be studied in the next section 3.

2.8 BIBLIOGRAPHICAL REMARKS

[GK] and [Kol] give nice introductory accounts of the theory of probability.

All the random variables considered in this section were discrete. However, this restriction would not be necessary if the reader were familiar with the notion of Lebesque integral. For a more general development of the notions of probability theory the reader should consult e.g. [F], [Gn], [Re1], [Re2] ,[Ro]. The proof of Bernshtein's law of large numbers given here is partly based on the account given in [Re1], pp. 322 - 326, and [Re2], page 200.

Buffon's needle problem is due to Buffon (1707 - 1788) and is described in his Essai d'Arithmetique morale. For more information on the Monte Carlo method the reader can consult the excellent introductory book [So]. There are numerous books and essays on the Monte Carlo method. These include [Hou], [Shr], [Br], [N], [Hal].

3 PSEUDO-RANDOM GENERATORS

3.1 INTRODUCTION

Of the four pseudo-random generators presented in this section the first two, the Linear Congruence Generator and the $1/p$-Generator (see subsections 3.2, 3.3) are predictable, while the other two, the Quadratic Residue Generator and the Index Generator (see subsections 3.10, 3.8) are unpredictable.

In subsections 3.2, 3.3 the proof of the predictability of the Linear Congruence Generator and the $1/p$-Generator respectively is studied. Subsections 3.5 and 3.6 examine questions relating to factoring and to the periodicity of quadratic residues respectively, and they will be used in the study of the security of the Quadratic Residue Generator. The definition of the model of computation to be used in the sequel, the probabilistic polynomial size circuit, is given in subsection 3.7.

The reader should notice some of the general notions emerging from the presentation in subsections 3.10, 3.8. These notions, whose study is postponed till section 5, include the notions of $1-1$, one-way function, amplification of advantage and unpredictable pseudo-random generators. An understanding of the present material will not only provide a good introduction to the general theory, but will also help introduce a number of examples essential to clarifying the development of the above concepts.

3.2 THE LINEAR CONGRUENCE GENERATOR

Let x, a, b, m be given fixed but unknown positive integers such that m is greater than $\max\{a, b, x\}$. Define the infinite sequence $x_0, x_1, \ldots, x_i, \ldots$ and the infinite sequence $x'_1, x'_2, \ldots, x'_i, \ldots$ of differences as follows:

$$x_i \equiv \begin{cases} x & \text{if } i = 0 \\ (a \cdot x_{i-1} + b) \bmod m & \text{if } i > 0, \end{cases}$$

and

$$x'_{i+1} = (x_{i+1} - x_i), \quad \text{where } i \geq 0.$$

Notice that for all $i \geq 1$,

$$x'_{i+1} \equiv a \cdot x'_i \bmod m.$$

The linear congruence generator, abbreviated $LGEN$ accepts as input a quadraple $< x, a, b, m >$ as above; the output $LGEN(x, a, b, m)$ is the infinite sequence $x_0, x_1, \ldots, x_i, \ldots$ defined from x, a, b, m as above.

Example 3.1 *LGEN(3,7,5,12) = 3,2,7,6,11,10,3,2,7,6,11,10,...*

The problem to be investigated in the sequel is the following:

Question: Does there exist an efficient algorithm which when given as input a sufficiently long initial segment of the infinite sequence $x_0, x_1, \ldots, x_i, \ldots$ will output integers a', b', m' such that for all $i, x_i \equiv (a' \cdot x_{i-1} + b') \bmod m'$?

For each $i \geq 1$, let $g_i = \gcd(x'_1, \ldots, x'_i)$.

Lemma 3.1

The least i ≥ 1 such that $g_i | x'_{i+1}$ is $\leq 2 + \lceil \log_2 m \rceil$.

Proof: Let $t =$ the least $i \geq$ such that $g_i | x'_{i+1}$. It is clear that for all i,

$$g_1 = x'_1 \text{ and } g_{i+1} = \gcd(g_i, x'_{i+1}).$$

However, if g_i does not divide x'_{i+1} then $g_{i+1} \leq g_i/2$. Consequently,

$$g_{t-1} \leq \frac{g_{t-2}}{2}, g_{t-2} \leq \frac{g_{t-3}}{2}, \ldots, g_2 \leq \frac{g_1}{2}.$$

It follows easily that,

$$g_{t-1} \leq \frac{g_1}{2^{t-2}} = \frac{x'_1}{2^{t-2}},$$

and hence,

$$t - 2 \leq \log_2 |x'_1| < \log_2 m.$$

This completes the proof of the lemma.•

Using the notation described above the following result can be proved.

Theorem 3.1 *(J. Plumstead) There is an efficient algorithm A which when given as input the sequence $x_0, x_1, \ldots, x_{t+1}$, produced by $LGEN(x, a, b, m)$, where $t =$ the least $i \geq 1$ such that $g_i | x'_{i+1}$, it will output integers a', b' such that for all $i \geq 1$,*

$$x_i \equiv (a' \cdot x_{i-1} + b') \bmod m.$$

The algorithm A runs in time polynomial in $\log_2 m$. Moreover, $t \leq 2 + \lceil \log_2 m \rceil$.

Proof: The upper bound on the size of t is an immediate consequence of lemma 3.1. The algorithm A is defined as follows:

Input: $x_0, x_1, \ldots, x_{t+1}$.

Step 1: Put $x'_i = x_i - x_{i-1}$, where $1 \leq i \leq t + 1$.

Step 2: Put $d = \gcd(x'_1, \ldots, x'_t)$.

Step 3: Compute $u_1, \ldots, u_t$ such that

$$d = \sum_{i=1}^{t} u_i \cdot x'_i$$

Output:

$$a' = \sum_{i=1}^{t} u_i \cdot \frac{x'_{i+1}}{d}$$

$$b' = x_1 - a' \cdot x_0$$

It will be shown that
Claim: $a'x'_i \equiv x'_{i+1}$ mod$m$, for all $i \geq 1$.
Proof of the claim: Let $g = \gcd(m, d)$. Then

$$ad \equiv a \sum_{i=1}^{t} \cdot u_i \cdot x'_i \equiv \sum_{i=1}^{t} a \cdot u_i \cdot x'_i \equiv$$

$$\sum_{i=1}^{t} u_i \cdot x'_{i+1} \equiv d \sum_{i=1}^{t} \cdot u_i \cdot \frac{x'_{i+1}}{d} \equiv a'd \bmod m.$$

It follows from the definition of g that

$$a \equiv a' \bmod \left(\frac{m}{g}\right).$$

However, for all $i \geq 1$, $g | \gcd(x'_i, m)$. It follows that for all $i \geq 1$,

$$a \equiv a' \bmod \left(\frac{m}{\gcd(x'_i, m)}\right). \tag{1}$$

But a is a solution of the congruence

$$u \cdot x'_i \equiv x'_{i+1} \bmod m. \tag{2}$$

An immediate consequence of the theorem on solving linear congruences is that every solution of (2) is of the form

$$a + \frac{jm}{\gcd(x'_i, m)}, \quad j = 0, 1, \ldots, \gcd(x'_i, m) - 1.$$

It follows from (1) that a' must be a solution of congruence (2). This completes the proof of the claim.

The rest of the proof of the theorem follows from the above claim and the following congruences:

$$a' \cdot x_i + b - x_{i+1} \equiv a' \cdot x_i + (x_1 - a' \cdot x_0) - x_{i+1} \equiv$$

$$a' \cdot (x_i - x_0) - (x_{i+1} - x_1) \equiv a' \sum_{k=1}^{i} x'_k - \sum_{k=1}^{i} x'_{k+1} \equiv$$

$$\sum_{k=1}^{i} (a' \cdot x'_k - x'_{k+1}) \equiv 0 \bmod m. \bullet$$

Predicting the modulus is a rather intricate problem and the reader is advised to consult [Pl] for more details. The periodicity of the linear congruence generator, as well as the problem of choice of modulus that will make the generator as secure as possible is studied in detail in subsection 3.2.1.2. of [Kn].

EXERCISES

Throughout the exercises below the notation of the above subsection will be used. Show that for all $n \geq 1$,

1: $x_n \equiv a^n x_0 + (a^{n-1} + \cdots + a + 1)b \bmod m$.

2: $x'_n \equiv a^n x'_1 \bmod m$.

Further, assume that $\gcd(a, m) = 1$ and show that

3: $x_n \equiv (x_{n+1} - b)a^{\varphi(m)-1} \bmod m$.

4: If $(a-1)|b$ then $x_n = x_{\varphi(m)+n}$. **Hint:** use exercise 1.

5: $x'_n = x'_{\varphi(m)+n}$. **Hint:** use exercise 2.

3.3 THE (1/p)-GENERATOR

Throughout the present subsection p will denote an odd prime number, g will be a fixed primitive root of Z_p^*, and $|p|$ the length of p in base g i.e. $|p| = \lceil \log_g p \rceil$. An integer r such that $0 \leq r < g$ will also be called a g-digit. Given any integer r such that $0 < r < g$ define the infinite sequence $r_0, r_1, \ldots, r_m, \ldots$ as follows

$$r_m \equiv r_0 \cdot g^m \bmod p, \; m \geq 0. \tag{3}$$

Since g is a primitive root of Z_p^*, it follows using the Euler-Fermat theorem, that for any $1 < r < p$ the period of the sequence $r_0, r_1, \ldots, r_m, \ldots$ equals $p-1$ i.e. $p-1 =$ the least m such that $r_0 = r_m$. Moreover, $\{r_0, r_1, \ldots, r_{p-2}\} = \{1, 2, \ldots, p-1\}$.

It is an immediate consequence of the Euclidean algorithm and the definition of r_m that for each $m \geq 0$ there exists a nonnegative integer $q_{m+1} < g$, such that

$$\frac{r_m}{p} = \frac{q_{m+1}}{g} + \frac{1}{g} \cdot \frac{r_{m+1}}{p}. \tag{4}$$

It follows that for all $m \geq 0$,

$$\frac{r_o}{p} = \frac{q_1}{g} + \frac{q_2}{g^2} + \cdots \frac{q_m}{g^m} + \frac{1}{g^m} \cdot \frac{r_m}{p}. \tag{5}$$

Multiplying equation (4) by $g^m p$ one obtains

$$r_o g^m = (q_1 g^{m-1} + q_2 g^{m-2} + \cdots q_m)p + r_m \tag{6}$$

For any sequence $x_1, x_2, \ldots, x_m$ of g-digits let the notation

$$x_1 x_2 \ldots x_m, \; .x_1 x_2 \ldots x_m$$

be used as an abbreviation of

$$x_1 g^{m-1} + x_2 g^{m-2} + \cdots x_m, \; \frac{x_1}{g} + \frac{x_2}{g^2} + \ldots + \frac{x_m}{g^m},$$

respectively. With these abbreviations, equations (5), (6) can be generalized, for each $i \geq 0$, to

$$\frac{r_i}{p} = .q_{i+1}q_{i+2}\cdots q_{i+m} + \frac{1}{g^m}\cdot\frac{r_{i+m}}{p}, \tag{7}$$

$$r_i g^m = (q_{i+1}q_{i+2}\cdots q_{i+m})p + r_{i+m} \tag{8}$$

An infinite sequence $x_1, x_2, \ldots, x_m, \ldots$ of g-digits is called a **de Bruijn sequence of period $p-1$ and base g** if the sequence $x_1, x_2, \ldots, x_m, \ldots$ of g-digits is periodic with period $p-1$, every finite sequence of g-digits of length $|p|-1$ occurs at least once as a segment of $x_1, x_2, \ldots, x_m, \ldots$, but every finite sequence of g-digits of length $|p|$ occurs at most once as a segment of $x_1, x_2, \ldots, x_m, \ldots$.

Example 3.2. *For $p = 3$ (= 11 in base 2), the sequence $0,1,0,1,\ldots$ is a de Bruijn sequence of period 2 and base 2.*

2. For $p = 5$ (= 101 in base 2), the sequence $0,1,1,0,0,1,1,0,\ldots$ is a de Bruijn sequence of period 4 and base 2.

The $1/p$ generator, abbreviated by *PGEN*, accepts as input the triple $< p, r, g >$, where p is a prime, $0 < r < g$ and g is a primitive root of Z_p^*; the output $PGEN(p,r,g)$ is the infinite sequence $q_1, q_2, \ldots, q_m, \ldots$ of g-digits which arises when the rational number r/p is represented in base g (see equation (7)).

The notation established above will be used during the cource of the proof of the theorems below.

Theorem 3.2 *For any primitive root g modulo p and any g-digit r, the sequence $PGEN(p,r,g)$ is a de Bruijn sequence of period $p-1$ and base g.*

Proof: In view of equation (7) one obtains that for all $k \geq 0$

$$\frac{r_k}{p} = .q_{k+1}q_{k+2}\cdots = \sum_{i=1}^{\infty}\frac{q_{k+i}}{g^i} \tag{9}$$

But the sequence $r_0, r_1, \ldots, r_m, \ldots$ is periodic with period $p-1$ and hence for all $i \geq 0$, $r_i = r_{i+p-1}$. It follows from this and equation (9) that

$$.q_{i+1}q_{i+2}\cdots = .q_{i+p}q_{i+p+1}\cdots \tag{10}$$

It follows that the period of the sequence $q_0, q_1, \ldots, q_m, \ldots$ must be $\leq p-1$. It remains to show that it is exactly equal to $p-1$. Indeed, assume on the contrary that the period is i and $0 < i < p-1$. Then it clear that

$$.q_{i+1}q_{i+2}\cdots = .q_1q_2\cdots, \tag{11}$$

and hence $r_0/p = r_i/p$, which is a contradiction. It will now be shown that the sequence $q_1, q_2, \ldots$ is a de Bruijn sequence. Let $\bar{d} = d_1, \ldots, d_t$ be an arbitrary finite sequence of g-digits of length $t \geq |p| - 1$. It is then clear that the following statements (12 - 15) are equivalent

$$\bar{d} \text{ is a segment of the sequence } q_1, q_2, \ldots \tag{12}$$

$$\text{for some } i \geq 0, \bar{d} \text{ is an initial segment of } q_{i+1}, q_{i+2}, \ldots \tag{13}$$

$$\text{for some } i \geq 0,\ \bar{d} \text{ is an initial segment of the expansion of } \frac{r_i}{p} \tag{14}$$

$$\text{for some } k \geq 0,\ \bar{d} \text{ is an initial segment of the expansion of } \frac{k}{p} \tag{15}$$

However, to each finite sequence $\bar{d} = d_1, \ldots, d_t$ of g-digits of length t there corresponds exactly one subinterval

$$\left[\frac{i}{g^t}, \frac{i+1}{g^t}\right), \text{ where } 0 \leq i < g^t,$$

of $[0, 1)$; namely, the subinterval to which the real number $.d_1 \ldots d_t$ belongs. Since, $g^{|p|-1} < p < g^{|p|}$, it is clear that

$$\frac{1}{g^{|p|}} < \frac{1}{p} < \frac{1}{g^{|p|-1}}. \tag{16}$$

It is now easy to see using equation (16) and the properties (12 - 15) that for each $i \geq 0$,

$$\text{there is at least one } k < p \text{ such that } \frac{k}{p} \in \left[\frac{i}{g^{|p|-1}}, \frac{i+1}{g^{|p|-1}}\right).$$

$$\text{there is at most one } k < p \text{ such that } \frac{k}{p} \in \left[\frac{i}{g^{|p|}}, \frac{i+1}{g^{|p|}}\right).$$

This completes the proof of the theorem.●

The unpredictability of the $1/p$ generator follows from the theorem below.

Theorem 3.3 (**Blum-Blum-Shub**) *Let g be a primitive root modulo a prime p, and $0 < r < p$. Then there there exists an algorithm A running in time polynomial in $|p|$ such that if $k = \lceil \log_g(2p^2) \rceil$ then for all $m \geq 0$,*

$$A(g, q_{m+1}, q_{m+2}, \ldots, q_{m+k}) = < p, r_m > .$$

Proof: Let $A_1/B_1, A_2/B_2, \ldots$ denote the sequence of convergents of the fraction $(q_{m+1} q_{m+2} \cdots q_{m+k})/g^k$ (see section 1). By assumption, $k = \lceil \log_g(2p^2) \rceil \geq \log_g(2p^2)$, and hence $g^k \geq 2p^2$. It is then clear, using inequality

$$\frac{1}{g^k} \cdot \frac{r_{k+m}}{p} < \frac{1}{g^k}$$

and equation (8) that

$$\left|\frac{(q_{k+1}\cdots q_{k+m})}{g^k} - \frac{r_m}{p}\right| < \frac{1}{g^k} \leq \frac{1}{2p^2}.$$

It follows that r_m/p is a convergent of $(q_{m+1}q_{m+2}\cdots q_{m+k})/g^k$ and hence,

$$\frac{r_m}{p} = \frac{A_i}{B_i}, \text{ for some } i \geq 0. \tag{17}$$

Since, $\gcd(r_m, p) = \gcd(A_i, B_i) = 1$, it follows that $r_m = A_i$, $p = B_i$. To complete the proof of the theorem it will be shown that A_i/B_i can be obtained by generating the sequence $A_1/B_1, A_2/B_2, \ldots$ of convergents until the j-th fraction A_j/B_j has $q_{m+1}, q_{m+2}, \ldots, q_{m+k}$ as its k first g-digits. Indeed, let j be the first index such that the first k g-digits of the fraction A_j/B_j are $q_{m+1}, q_{m+2}, \ldots, q_{m+k}$. It follows from equation (17) and the minimality of j that $j \leq i$. Assume on the contrary that $A_i/B_i \neq A_j/B_j$. Then it is clear that

$$\frac{1}{B_iB_j} \leq \left|\frac{A_iB_j - A_jB_i}{B_iB_j}\right| = \left|\frac{A_i}{B_i} - \frac{A_j}{B_j}\right| < \frac{1}{g^k}.$$

Since $j \leq i$, it must be true that $B_j \leq B_i$ and hence $2B_i^2 = 2p^2 \leq g^k < B_iB_j \leq B_i^2$, which is a contradiction. The amount of steps needed to compute A_i, B_i is O(number of steps needed to compute the i-th Fibbonacci number), and hence both A_i and B_i can be computed in polynomial in $|p|$ many steps.•

EXERCISES

Let g be a primitive root modulo p, p a prime, $0 < r < p$.

1: There exists a polynomial in $|p|$ time algorithm, A_1, such that for all m, $A_1(p, g, q_{m+1}, q_{m+2}, \ldots, q_{m+|p|}) = r_m$.

2: There exists a polynomial in $|p|$ time algorithm, A_2, such that for all m, i, $A_2(p, g, r_m, i) =< r_{m-1}, r_{m+i}, q_m, \ldots, q_{m+i} >$.

3: There exists a polynomial in $|p|$ time algorithm, A_3, such that for all m, if $r_m g \neq r_{m+1}$ then $A_3(g, r_m, r_{m+1}) = p$. **Hint:** Let S be the set $\{(gr_m - r_{m+1})/i \ : \ i = 0, 1, \ldots, g-1\}$. By equation (4), $p = (gr_m - r_{m+1})/q_{m+1} \in S$. Show that p is the unique $x \in S$ such that for all $i = 1, \ldots, g$, $\gcd(x, i) = 1$.

3.4 QUADRATIC RESIDUES IN CRYPTOGRAPHY

In constructing cryptographic protocols one considers integers $n = pq$, where p, q are two distinct odd primes. For such integers n it will be necessary to study the behavior of the Langrange-Jacobi symbol modulo n. **From now on and for the rest of this subsection it will be assumed that $n = pq$, where p, q are two distinct odd primes.**

Let x be a quadratic residue modulo n. Call u **a square root** of x modulo n, if $u^2 \equiv x \bmod n$. The next theorem determines the number of square roots of any given qudratic residue.

Theorem 3.4 *Any quadratic residue has exactly four square roots modulo n.*

Proof: By the Chinese Remainder theorem there exist integers a, b such that

$$a \equiv 1 \bmod p \text{ and } a \equiv -1 \bmod q,$$

$$b \equiv -1 \bmod p \text{ and } b \equiv 1 \bmod q$$

Since both p, q are odd it is clear that $a, b, 1, -1$ are distinct modulo n. Moreover, $a^2 \equiv b^2 \equiv 1 \bmod n$. It follows that $1, -1, a, b$ are four distinct modulo n square roots of 1. The rest of the proof follows from exercise 1 of the subsection on the homomorphism theorem. •

For the rest of this subsection it will be assumed that $p \equiv q \equiv 3 \bmod 4$, i.e. both $(p-1)/2$ and $(q-1)/2$ are odd. It follows that $(-1|p) = (-1|q) = -1$ and $(-1|n) = (-1|p)(-1|q) = 1$. Hence, for all $x \in Z_n^*, (-x|n) = (x|n)$.

Theorem 3.5 *(i) If $x^2 \equiv y^2 \bmod n$, and $x, y, -x, -y$ are distinct modulo n then $(x|n) = -(y|n)$.*
(ii) The mapping $x \longrightarrow x^2 \bmod n (x \in QR_n, x^2 \bmod n \in QR_n)$ is $1-1$ and onto i.e. every quadratic residue has a unique square root which is also a quadratic residue modulo n.

Proof:(i) Assume that x and y are as above. Then, it is clear that $n = pq | (x^2 - y^2) = (x-y)(x+y)$. Since $x, -x, y, -y$ are distinct modulo n, neither p nor q can divide both $x - y, x + y$. Without loss of generality assume that $p|(x-y)$, and $q|(x+y)$ (the other case is treated similarly.) Then $x \equiv y \bmod p$ and $x \equiv -y \bmod q$. It follows that $(x|p) = (y|p)$ and $(x|q) = -(y|q)$, and hence the proof of part (i) is complete.

(ii) Let a be any quadratic residue modulo n. By the previous theorem a has exactly four square roots modulo n, say $x, -x, y, -y$. By part (i) $(x|n) = -(y|n)$. Let x be the square root of a such that $(x|n) = +1$. It follows that either $(x|p) = (x|q) = +1$ or $(-x|p) = (-x|q) = +1$. Thus, one of $x, -x$ must be a quadratic residue modulo n. This completes the proof of the theorem. •

The above theorem implies that the mapping

$$x \longrightarrow x^2 \bmod n, \text{ where } (x \in QR_n, x^2 \bmod n \in QR_n)$$

is $1-1$ and onto, and hence it has an inverse which will be denoted by

$$x \longrightarrow \sqrt{x} \bmod n, \text{ where } (x \in QR_n, \sqrt{x} \bmod n \in QR_n).$$

It is immediate from the above considerations that every quadratic residue x modulo n has four square roots x_1, x_2, x_3, x_4 which satisfy: $(x_1|p) = (x_1|q) = +1, (x_2|p) = (x_2|q) = -1, (x_3|p) = -(x_3|q) = +1, -(x_4|p) = (x_4|q) = +1$. Moreover, the square root x_1 is also a quadratic residue modulo n.

EXERCISES

1: Show that $|QR_n| = \varphi(n)/4, |Z_n^*(+1)| = \varphi(n)/2, |Z_n^*(-1)| = \varphi(n)/2$.

2: The mapping $x \longrightarrow x^2$ mod$p(x \in QR_p, x^2$ mod$p \in QR_p)$ is $1-1$ and onto i.e. every quadratic residue has a unique square root which is also a quadratic residue modulo p.

3.5 FACTORING AND QUADRATIC RESIDUES

The main theorem of the present subsection is due to Rabin. In a sense it shows that the problems of factoring a composite number, and solving quadradic congruences modulo a composite number are equivalent.

Theorem 3.6 *(Rabin) The following statements are equivalent:*

(i) There is an efficient algorithm A such that for all n, if n is the product of two distinct odd primes both congruent to 3 modulo 4 then $A(n) = p$, where p is a prime factor of n.

(ii) There is an efficient algorithm B such that if n is the product of two distinct odd primes both congruent to 3 modulo 4 and $x \in QR_n$ then $B(n,x) = \sqrt{x}$ mod$n$.

Proof: (ii) $\Rightarrow$ (i)

Assume the algorithm B is given. The algorithm A is defined as follows:

```
Input: $n$
Step 1: Choose a random $y$ such that $(y|n) = -1$.
Step 2: Compute $x \equiv y^2$ mod$n$.
Step 3: Compute $z = B(n,x)$.
Output: $\gcd(y+z, n)$.
```

It remains to show that this algorithm works. Indeed, it is clear that $x \equiv y^2 \equiv z^2$ mod$n$. Hence, $n|(y^2 - x^2) = (y-z)(y+z)$. Assume that $n = pq$. It follows that $pq|(y-z)(y+z)$. But, $(y|n) = -(z|n) = -1$ and therefore $y \not\equiv z$ mod$n$. Consequently, $\gcd(y+z,n)$ must be one of the prime factors of n.

(i) $\Rightarrow$ (ii)

The algorithm B uses the Adelman-Manders-Miller algorithm for computing square roots modulo a prime and is defined as follows:

```
Input: $n, x$
Step 1: Let $p = A(n), q = n/p$.
Step 2: Compute $u \in QR_p$, $v \in QR_q$ such that $x \equiv u^2$ mod$p$, $x \equiv v^2$ mod$q$.
Step 3: Compute $a, b$ such that $1 = ap + bq$.
Step 4: Compute $c = bq$ and $d = ap$.
```

Output: $cu + dv$.

Let $w = cu + dv$. It is clear from the above algorithm that $w^2 \equiv x \bmod n$ It remains to show that $w \in QR_n$. $(w|n) = 1$. Indeed, notice that $c \equiv 0 \bmod q$ and $c \equiv 1 \bmod p, d \equiv 0 \bmod p$ and $d \equiv 1 \bmod q$. Hence, $(w|p) = (u|p) = 1$ and $(w|q) = (v|q) = 1$. Thus, $w \in QR_p$ and $w \in QR_q$ and consequently, $w \in QR_n$.•

EXERCISES

1: Show that the following statements are equivalent:

(i) There is an efficient algorithm A such that for all n, if n is the product of two distinct primes then $A(n) = p$, where p is a prime factor of n.

(ii) There is an efficient algorithm B such that if n is the product of two distinct primes then $B(n) = \varphi(n)$. **Hint:** Notice that $\varphi(n) = n - p - q + 1$.

3.6 PERIODICITY OF QUADRATIC RESIDUES

For each n and each x in Z_n^* the **order** of x with respect to n, abbreviated $od_n(x)$, is the least nonnegative exponent e such that $x^e \equiv 1 \bmod n$. **Throughout this subsection** $n = pq$, where p, q are two distinct odd primes such that $p \equiv q \equiv 3 \bmod 4$. For each quadratic residue $x \in QR_n$ define the infinite sequence

$$\ldots, x_{n,-2}, x_{n,-1}, x_{n,0} = x, x_{n,1}, x_{n,2}, \ldots$$

of quadratic residues as follows:

$$x_{n,i} \equiv \begin{cases} x^{2^i} \bmod n & \text{if } i \geq 0 \\ \sqrt{x_{n,i+1}} & \text{if } i < 0 \end{cases}$$

The modulus n used as a subscript in $x_{n,i}$ will usually be omitted, but this will cause no confusion because n will be easily understood from the context.

Call **period** of x, abbreviated $\overline{\pi}(x)$, the least positive integer i such that $x_i = x$. The purpose of the theorems below is to determine the size of $\overline{\pi}(x)$.

Theorem 3.7 *(***Blum-Blum-Shub***)*

$$od_n(x) = \lambda(n)/2 \text{ and } od_{\lambda(n)/2}(2) = \lambda(\lambda(n)) \Rightarrow \lambda(\lambda(n)) = \overline{\pi}(x).$$

Proof: Put $\overline{\pi} = \overline{\pi}(x)$. By assumption $\lambda(n)/2$ is the least exponent e such that $x^e \equiv 1 \bmod n$. But $x_{\overline{\pi}} \equiv x \equiv x^{2^{\overline{\pi}}} \bmod n$. It follows that $x^{2^{\overline{\pi}}-1} \equiv 1 \bmod n$, and consequently $\lambda(n)/2|(2^{\overline{\pi}} - 1)$. Thus, $2^{\overline{\pi}} \equiv 1 \bmod(\lambda(n)/2)$. Using the hypothesis $od_{\lambda(n)/2}(2) = \lambda(\lambda(n))$ one obtains that $\lambda(\lambda(n))$ = the least exponent e such that $2^e \equiv 1 \bmod(\lambda(n)/2)$. It follows that $\lambda(\lambda(n))|\overline{\pi}$.

Hence, the theorem will follow from the following

Claim: $\overline{\pi}|\lambda(\lambda(n))$

Proof of the Claim: First notice that if $a \equiv b^2 \bmod n$ then $od_n(a)|od_n(b)$. Indeed, set $e = od_n(b)$. Then $b^e \equiv 1 \bmod n$. Hence, $a^e \equiv b^{2e} \equiv 1 \bmod n$.

Thus $od_n(a)|od_n(b) = e$. It follows from the above observation that for all $i, od_n(x_{i+1})|od_n(x_i)$. However $x_0 = x_{\overline{\pi}}$. It is therefore clear that for all $i, od_n(x_i)$ $= od_n(x_0)$. Further, it can be proved that $od_n(x)$ is odd. Indeed, assume on the contrary that $od_n(x) = 2^e m$, where m is odd, $e > 0$. Then $x^{2^e m} \equiv x_1^{2^{e-1} m} \equiv$ $1 \bmod n$, which contradicts $od_n(x_1) = od_n(x_0)$.

It is immediate from the definition of $\overline{\pi}(x) = \overline{\pi}$ that $\overline{\pi} =$ the least exponent e such that $2^e \equiv 1 \bmod(od_n(x))$. Since $\gcd(2, od_n(x)) = 1$, it follows from Carmichael's theorem that $\overline{\pi}|\lambda(od_n(x))$. But $od_n(x)|\lambda(n)$, and consequently $\lambda(od_n(x))|\lambda(\lambda(n))$. This is enough to complete the proof of the claim, and hence of the theorem.●

The above theorem gives necessary hypothesis which imply that $\lambda(\lambda(n)) =$ $\overline{\pi}(x)$. Next it will be determined for which integers are these conditions satisfied. A prime number p is called special if there exist prime numbers p_1, p_2 such that $p = 2p_1 + 1, p_1 = 2p_2 + 1$ and $p_2 > 2$. The number $n = pq$ is called special if both primes p and q are special.

Remark: It is conjectured that there exist infinitely many special primes. Some examples of special primes are obtained for $q = 11, 23, 83$, in which case $p = 2q + 1$ is a special prime. For a detailed discussion of this conjecture, as well as for a table of bigger special primes see [Scha] (pp. 28 - 30).

Theorem 3.8 *(***Blum-Blum-Shub***) Let $n = pq$ be special, such that $p = 2p_1 + 1, p_1 = 2p_2 + 1, q = 2q_1 + 1, q_1 = 2q_2 + 1$, and p_1, p_2, q_1, q_2 are primes. If 2 is a quadratic residue modulo at most one of p_1, q_1 then*

$$od_{\lambda(n)/2}(2) = \lambda(\lambda(n)).$$

Proof: It is an immediate consequence of the definition of the Carmichael function that $\lambda(n) = 2p_1q_1, \lambda(n)/2 = p_1q_1, \lambda(\lambda(n)) = 2p_2q_2, \lambda(\lambda(n)/2) = 2p_2q_2$. It follows from Carmichaels theorem that $od_{\lambda(n)/2}(2)|\lambda(\lambda(n)/2) = 2p_2q_2$. Assume on the contrary that $od_{\lambda(n)/2}(2) = 2p_2q_2$. In each of the three cases below *a* contradiction will be derived

Case 1: $od_{\lambda(n)/2}(2)|2p_2$

It is clear that $2^{2p_2} \equiv 1 \bmod(\lambda(n)/2) \equiv 1 \bmod(p_1q_1)$. Hence, $2^{2p_2} \equiv$ $1 \bmod q_1$. By the Euler-Fermat theorem it is true that $2^{2q_2} \equiv 1 \bmod q_1$. It follows that $2^{\gcd(2p_2, 2q_2)} \equiv 1 \bmod q_1$, and consequently $2^2 \equiv 1 \bmod q_1$, since $\gcd(2p_2, 2q_2) = 2$. But this contradicts the fact that $q_1 > 3$.

Case 2: $od_{\lambda(n)/2}(2)|2q_2$

This is similar to case 1.

Case 3: $od_{\lambda(n)/2}(2) = p_2q_2$

It is clear that $2^{2p_2q_2} \equiv 1 \bmod(\lambda(n)/2) \equiv 1 \bmod(p_1q_1)$. Hence, $2^{2p_2q_2} \equiv$ $1 \bmod q_1$. Since p_2 is an odd prime the last congruence implies that $2^{q_2} \not\equiv$ $-1 \bmod q_1$. By Euler's criterion, and since $q_2 = (q_1 - 1)/2$, $2^{q_2} \equiv (2|q_1) \bmod q_1$. It follows that $(2|q_1) \equiv 1$, and hence $2 \in QR_{q_1}$ Similarly $2 \in QR_{p_1}$, which is a contradiction.●

Theorem 3.9 $|\{x \in QR_n : od_n(x) = \lambda(n)/2\}| = \Omega(n/((\log\log n)^2)$

Proof: By assumption $n = pq$ is a product of two primes. Z_p^* (respectively Z_q^*) has exactly $\varphi(\varphi(p))$ (respectively $\varphi(\varphi(q))$) generators. Let $g \in Z_p^*$ (respectively $h \in Z_q^*$) be a generator of Z_p^* (respectively Z_q^*). By the Chinese Remainder theorem there exist a unique modulo n integer a such that $a \equiv g \bmod p$ and $a \equiv h \bmod q$. It follows that $od_n(a) = \lambda(n) = \mathrm{lcm}(p-1, q-1)$. Consequently, there exist at least $\varphi(\varphi(p)) \cdot \varphi(\varphi(q))$ elements in Z_n^* of order $\lambda(n)$. It follows from a theorem of E. Landau that for all $x > 2$,

$$\frac{x}{\varphi(x)} < 6\log\log x.$$

To complete the proof of the theorem notice that

$$\varphi(\varphi(p)) \cdot \varphi(\varphi(q)) = \varphi(p-1) \cdot \varphi(q-1) \geq$$

$$\frac{p-1}{6\log\log(p-1)} \cdot \frac{q-1}{6\log\log(q-1)} \geq \frac{n-p-q-1}{(6\log\log n)^2} \geq$$

$$(n/2)/(6\log\log n)^2 = \Omega(n/((\log\log n)^2).$$

But the mapping $x \longrightarrow x^2 \bmod n (x \in Z_n^*, x^2 \bmod n \in QR_n)$ is $4-1$. Moreover, if $x \in Z_n^*$ is of order $\lambda(n)$ then $x^2 \bmod n \in QR_n$ is of order $\lambda(n)/2$. Using this observation one can complete the proof of the theorem easily.•

The next theorem establishes the connection between computing the period of quadratic residues and the factoring problem.

Theorem 3.10 (**Blum-Blum-Shub**) *Assume there exist efficient algorithms A, A' such that for all n which is the product of two distinct odd primes, for all $x \in QR_n$ and all $i \geq 0$,*

$$A(n,x) = \overline{\pi}(x) \quad \text{and} \quad A'(n,x,i) = x_i$$

Then there exists an efficient algorithm which given as input an integer n which is the product of two distinct odd primes, it will output a prime factor of n .

Proof: The factoring algorithm is defined as follows:

Input: n

Step 1: Choose a random y such that $(y|n) = -1$.

Step 2: Compute $x \equiv y^2 \bmod n$.

Step 3: Compute $\overline{\pi} = A(n,x)$.

Step 4: Compute $z = A'(n,x,\overline{\pi}-1)$.

Output: $\gcd(z-y,n)$.

The proof that the above algorithm works is similar to the proof of theorem 3.6 and uses the fact that in the above algorithm $x \equiv y^2 \equiv z^2 \bmod n$.•

EXERCISES

1: Prove results similar to those of theorems 3.7, 3.8, 3.9 for quadratic residues modulo p, where p is prime; to be more specific show that for $x \in QR_p$:

(i) If $od_p(x) = \lambda(p)/2$ and $od_{\lambda(p)/2}(2) = \lambda(\lambda(p))$ then $\lambda(\lambda(p)) = \overline{\pi}(x)$.

(ii) If p is special and $2 \in QNR_{(p-1)/2}$ then $od_{\lambda(p)/2}(2) = \lambda(\lambda(p))$.

(iii)$|\{x \in QR_p : od_p(x) = \lambda(p)/2\}| = \Omega(p/\log\log p)$.

3.7 THE CIRCUIT AS MODEL OF COMPUTATION

An (n,t) circuit is an acyclic, labeled (i.e. with labeled nodes), digraph (i.e. with directed edges) consisting of

1. a list of n distinguished **input** nodes each of which has indegree 0 (i.e. no entering edges) and outdegree 1 (i.e. no exiting edges),

2. **internal** nodes each of which has outdegree 1 and is labeled with one of the symbols $\oplus, \cdot$,

3. a list of t distinguished **output** nodes each of which has outdegree 0 and is labeled with one of the symbols $\oplus, \cdot$.

The nodes of the circuit are also called **gates**. Each gate of the circuit can hold one of the two boolean values 0 or 1. An **assignment** of the input nodes of an (n,t) circuit is an n- tuple $(x_1,\ldots,x_n) \in \{0,1\}^n$. If an internal $\oplus$ (respectively $\cdot$) gate has indegree k then the output of this gate on input $(u_1,\ldots,u_k) \in \{0,1\}^k$ is $u_1 \oplus \cdots \oplus u_k$ (respectively $u_1 \cdots u_k$.) The value of the circuit on the input assignment $(x_1,\ldots,x_n) \in \{0,1\}^n$ is the value of the circuit obtained at the t output gates when one evaluates the output of each of the internal gates in topological order along the circuit.

Thus, every (n,t) circuit C determines a function

$$f_C : \{0,1\}^n \longrightarrow \{0,1\}^t$$

such that $f_C(x_1,\ldots,x_n) = (y_1,\ldots,y_t)$, where $(y_1,\ldots,y_t)$ is the value of the t output gates of the circuit C when the input assignment is $(x_1,\ldots,x_n)$.

Example 3.3 *The circuit in figure 1 computes the function*

$$f(x_1,\ldots,x_9) = [(x_1 \oplus x_2 \oplus x_3) \oplus (x_4 \cdot x_5)] \oplus [(x_6 \cdot x_7) \cdot (x_8 \oplus x_9)].$$

The **size** $|C|$ of the circuit C is the total number of its gates and the **depth** $d(C)$ of the circuit is the length of its longest path.

An (n,m,t) **probabilistic circuit** C is an $(n+m,t)$ circuit with two distinct types of input gates:

1. a list of n distinguished input gates called **deterministic gates**,

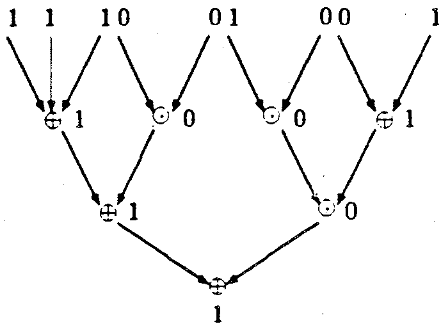


Figure 1: A Deterministic Circuit

2. a list of m distinguished input gates called **random gates**.

Let C be an (n,m,t) probabilistic circuit. To evaluate the value of C on the input assignment $(x_1,\dots,x_n)$, one assigns the deterministic input gates the values $(x_1,\dots,x_n)$, the random gates the values $(y_1,\dots,y_m)$ each with probability $1/2$ and then computes the output of the circuit C on the input assignment

$$(x_1,\dots,x_n,y_1,\dots,y_m).$$

A **polynomial size family of probabilistic circuits** is a family $C = \{C_n : n \geq 1\}$ of probabilistic circuits such that

1. each circuit C_n has n many deterministic input gates, and
2. there exists a polynomial with positive integer coefficients of degree ≥ 1 such that $|C_n| \leq P(n)$, for all $n \geq 1$.

From now on and for the rest of the present monograph all the circuits considered will be probabilistic, unless otherwise mentioned. For that reason, the name circuit whenever used will be identical to probabilistic circuit. In addition, in order to simplify the notation, for any circuit C considered, mention of its random gates will usually be suppressed. Thus, if C is an $(n,m,1)$ probabilistic circuit and $\epsilon > 0$ then the symbol

$$Pr[C(x) = 0] \geq \epsilon$$

will mean that with probability $\geq \epsilon$ the circuit C will output 0 on input x, where the probability space is the set $\{0,1\}^m$.

3.8 THE QUADRATIC RESIDUE GENERATOR

Throughout the present subsection n will range over integers which are the product of two distinct odd primes p, q such that $p \equiv q \equiv 3 \bmod 4$; $N = \{N_k : k \in I\}$ will denote a family of nonempty sets N_k of nonegative integers such that I is an infinite set of indices, and for all $n \in N_k$ the integer n has binary length exactly k. By theorem 3.5 the squaring mapping $x \longrightarrow x^2 \bmod n$ $(x \in QR_n, x^2 \bmod n \in QR_n)$ is $1-1$ and onto, and hence it has an inverse which will be denoted by $x \longrightarrow \sqrt{x} \bmod n$ $(x \in QR_n, \sqrt{x} \bmod n \in QR_n)$.

From now on and for the rest of this section the capital roman letters P, Q with subscripts or superscripts will range over nonzero polynomials with one indeterminate, positive coefficients, and degree ≥ 1, and the lowercase greek letters ϵ, δ with subscripts or superscripts will range over positive real numbers.

Definition 3.1 *A polynomial size circuit $C = \{C_k : k \geq 1\}$ has a $1/P$-advantage for computing the parity function for the family N, and this will be abbreviated by $APAR(C, N, 1/2 + 1/P)$, if for all but a finite number of indices $k \in I$ the following property holds for all $n \in N_k$,*

$$Pr[x \in QR_n : C_k(n, x) = \text{par}(\sqrt{x} \bmod n)] \geq \frac{1}{2} + \frac{1}{P(k)}.$$

Definition 3.2 *A polynomial size circuit $C = \{C_k : k \geq 1\}$ has a $1/P$-advantage for determining quadratic residuosity for the family N, and this will be abbreviated by $AQR(C, N, 1/2 + 1/P)$, if for all but a finite number of indices $k \in I$ the following property holds for all $n \in N_k$,*

$$\frac{1}{2}(Pr[C_k(n, x) = 1 \mid x \in QR_n] + Pr[C_k(n, x) = 0 \mid x \notin QR_n]) \geq \frac{1}{2} + \frac{1}{P(k)},$$

where for each $n \in N_k$, x ranges over $Z_n^*(+1)$.

Theorem 3.11 *For all polynomials P,*

$$(\exists C) APAR(C, N, 1/2 + 1/P) \Rightarrow (\exists C) AQR(C, N, 1/2 + 1/P)$$

Proof: The proof is based on the following

Claim: For all $x \in Z_n^*(+1)$, $x \in QR_n \Leftrightarrow \text{par}(x) = \text{par}(\sqrt{x^2} \bmod n)$.

Proof of the claim: $(\Rightarrow)$ Assume $x \in QR_n$. Then x is the unique square root modulo n of $x^2 \bmod n$. Hence, $x = \sqrt{x^2} \bmod n$. Conversely, $(\Leftarrow)$ suppose that $x \notin QR_n$ and put $x_0 = \sqrt{x^2} \bmod n$. Let $n = p \cdot q$. By assumption, $(x|n) = 1$ and $x \notin QR_n$. Since, both x, x_0 are square roots of $x^2 \bmod n$, it follows that $(x|p) = (x|q) = -1$ and $x = -x_0$. Thus, $\text{par}(x) \neq \text{par}(\sqrt{x^2} \bmod n)$, which is a contradiction.

Based on the claim one can give the proof of the theorem. Let C be a circuit such that $APAR(C, N, 1/2 + 1/P)$. To find a circuit $C' = \{C'_k : k \geq 1\}$ such that $AQR(C'.N, 1/2 + 1/P)$. Define

$$C'_k(n, x) = C_k(n, x^2 \bmod n) \oplus \text{par}(x) \oplus 1. \tag{18}$$

It is clear from the definitions that

$$C'_k(n, x) = 1 \Leftrightarrow C_k(n, x^2 \bmod n) = \text{par}(x). \tag{19}$$

Consider the sets $A_n = \{x \in QR_n : C_k(n, x) = \text{parity}(\sqrt{x} \bmod n)\}$. $X_n = \{x \in QR_n : x^2 \bmod n \in A_n\}$, $Y_n = \{x \in Z_n^*(+1) - QR_n : x^2 \bmod n \in A_n\}$. $W_n = \{x \in Z_n^*(+1) : x^2 \bmod n \in A_n\}$. It is then clear that $W_n = X_n \cup Y_n$ and $|X_n| = |A_n| = |Y_n|$. It follows that

$$Pr[x \in Z_n^*(+1) : x \in W_n] = \frac{|W_n|}{|Z_n^*(+1)|} = \tag{20}$$

$$\frac{|X_n| + |Y_n|}{2|QR_n|} = \frac{|A_n|}{|QR_n|} = Pr[x \in QR_n : x \in A_n]$$

As a consequence of equations (20) and definition 3.1 one obtains easily that

$$\frac{1}{2}(Pr[C'_k(n, x) = 1 \mid x \in QR_n] + Pr[C'_k(n, x) = 0 \mid x \notin QR_n]) =$$

$$Pr[x \in Z_n^*(+1) : x \in X_n] + Pr[x \in Z_n^*(+1) : x \in Y_n] =$$

$$Pr[x \in QR_n : x \in A_n] \geq \frac{1}{2} + \frac{1}{P(k)}.$$

This completes the proof of the theorem.•

A strengthening of definition 3.2 is given in the following

Definition 3.3 *A polynomial size circuit $C = \{C_k : k \geq 1\}$ has a $(1/2 - 1/P)$-advantage for determining quadratic residuosity for the family N, and this will be abbreviated by $AQR(C, N, 1 - 1/P)$, if for all but a finite number of indices $k \in I$ the following property holds for all $n \in N_k$,*

$$\frac{1}{2}(Pr[C_k(n, x) = 1 \mid x \in QR_n] + Pr[C_k(n, x) = 0 \mid x \notin QR_n]) \geq 1 - \frac{1}{P(k)},$$

where for each $n \in N_k$, x ranges over $Z_n^*(+1)$.

Theorem 3.12 (**Goldwasser-Micali**)

$$(\exists C)(\exists P)AQR(C, N, 1/2 + 1/P) \Rightarrow (\forall Q)(\exists C)AQR(C, N, 1 - 1/Q)$$

Proof: Assume that C is a polunomial size circuit and P is a polynomial such that the inequality of definition 3.2 holds. Put

$$p_n = Pr[C_k(n,x) = 1 \mid x \in QR_n]\text{and}q_n = Pr[C_k(n,x) = 1 \mid x \notin QR_n].$$

Then it is clear that for all but a finite number of indices $k \in I$,

$$\frac{p_n + (1-q_n)}{2} \geq \frac{1}{2} + \frac{1}{P(k)},$$

and therefore

$$\frac{p_n}{2} - \frac{q_n}{2} \geq \frac{1}{P(k)}. \tag{21}$$

The aim of the construction below, which is based on the Weak Law of Large Numbers, is to construct for every polynomial Q a new circuit C' that will satisfy the conclusion of the theorem. Indeed, let Q be given and define the circuit C' as follows:

Input: $k \geq 1$, $n \in N_k$, $x \in Z_n^*(+1)$.
Step 1: Put $m = 4 \cdot Q(k) \cdot P(k)^2$.
Step 2: Select m random quadratic residues $s_1^2, \ldots, s_m^2 \in QR_n$.
Step 3: Compute the following two integers:

$$R_n = |\{1 \leq i \leq m \ : \ C_k(n, s_i^2 \bmod n) = 1\}| \text{ and}$$

$$\overline{R}_{n,x} = |\{1 \leq i \leq m \ : \ C_k(n, x \cdot s_i^2 \bmod n) = 1\}|$$

Step 4: Compute $d_{n,x} = |R_n - \overline{R}_{n,x}|$

Output :

$$C_k'(n,x) = \begin{cases} 1 & \text{if } d_{n,x} \leq \frac{1}{P(k)} \\ 0 & \text{if } d_{n,x} > \frac{1}{P(k)} \end{cases}$$

It remains to show that the above polynomial size circuit C' satisfies property $AQR(C', N, 1 - 1/Q)$. First notice, see exercise 5 at the end of the present subsection, that if $x \in QR_n$ (respectively $x \notin QR_n$) then

$$xs_1^2 \bmod n, \ldots, xs_m^2 \bmod n$$

is a sequence of m random quadratic residues (respectively nonresidues). Let the notation $Pr_A[E]$ abbreviate the conditional probability $Pr[E|A]$ of the event E under the condition that the event A holds. Next, the Weak Law of Large Numbers implies that,

$$Pr_{QR_n}\left[x \in Z_n^*(+1) : \left|p_n - \frac{R_n}{m}\right| > \frac{1}{2P(k)}\right] < \frac{1}{4Q(k)},$$

$$Pr_{QR_n}\left[x \in Z_n^*(+1) : \left|p_n - \frac{\overline{R}_n}{m}\right| > \frac{1}{2P(k)}\right] < \frac{1}{4Q(k)},$$

$$Pr_{QNR_n}\left[\left|q_n - \frac{\overline{R}_{n,x}}{m}\right| > \frac{1}{2P(k)}\right] < \frac{1}{4Q(k)}.$$

Now, the following two claims will be proved (in the proofs below x ranges over $Z_n^*(+1)$):

Claim 1: $Pr_{QR_n}[|R_n/m - \overline{R}_{n,x}/m| \leq 1/P(k)] > 1 - 1/Q(k)$.

Indeed,

$$Pr_{QR_n}\left[\left|\frac{R_n}{m} - \frac{\overline{R}_{n,x}}{m}\right| \leq \frac{1}{P(k)}\right] =$$

$$Pr_{QR_n}\left[\left|\frac{R_n}{m} - p_n - \left(\frac{\overline{R}_{n,x}}{m} - p_n\right)\right| \leq \frac{1}{P(k)}\right] \geq$$

$$Pr_{QR_n}\left[\left|\frac{R_n}{m} - p_n\right| \leq \frac{1}{2P(k)} \text{ and } \left|\frac{\overline{R}_{n,x}}{m} - p_n)\right| \leq \frac{1}{2P(k)}\right] =$$

$$Pr_{QR_n}\left[\left|\frac{R_n}{m} - p_n\right| \leq \frac{1}{2P(k)}\right] \cdot Pr_{QR_n}\left[\left|\frac{\overline{R}_{n,x}}{m} - p_n)\right| \leq \frac{1}{2P(k)}\right] \geq$$

$$\left(1 - \frac{1}{4Q(k)}\right) \cdot \left(1 - \frac{1}{4Q(k)}\right) > 1 - \frac{1}{Q(k)}.$$

Claim 2: $Pr_{QNR_n}[|R_n/m - \overline{R}_{n,x}/m| > 1/P(k)] > 1 - 1/Q(k)$.

By the assumption in equation (21),

$$p_n - q_n \geq \frac{2}{P(k)}.$$

Since,

$$Pr_{QNR_n}\left[\left|\frac{R_n}{m} - p_n\right| > \frac{1}{2P(k)}\right] < \frac{1}{4Q(k)} \text{ and}$$

$$Pr_{QNR_n}\left[\left|\frac{\overline{R}_n}{m} - q_n\right| > \frac{1}{2P(k)}\right] < \frac{1}{4Q(k)},$$

with probability $\geq 1 - 1/(4Q(k))$, R_n/m must lie outside the closed interval

$$\left[q_n - \frac{1}{2P(k)}, p_n + \frac{1}{2P(k)}\right];$$

For the same reason, $\overline{R}_{n,x}/m$ must lie outside the same closed interval. It follows that

$$Pr_{QNR_n}\left[\left|\frac{R_n}{m} - \frac{\overline{R}_{n,x}}{m}\right| > \frac{1}{P(k)}\right] \geq \left(1 - \frac{1}{4Q(k)}\right)^2 > 1 - \frac{1}{Q(k)}.$$

This completes the proof of the theorem.•

Recall that to each $x \in QR_n$ an infinite sequence

$$\ldots, x_{n,-2}, x_{n,-1}, x_{n,0} = x, x_{n,1}x_{n,2}\cdots$$

of quadratic residues was associated as follows:

$$x_{n,i} \equiv \begin{cases} x^{2^i} \bmod n & \text{if } i \geq 0 \\ \sqrt{x_{n,i+1}} & \text{if } i < 0 \end{cases}$$

For each $x \in QR_n$ and each integer i, define the bits

$$b_{n,i}(x) = \text{par}(x_{n,i}).$$

The Quadratic Residue Generator, abbreviated $QRGEN$, accepts as input a pair $< x, n >$, where $x \in QR_n$; the output is the infinite sequence $\ldots, b_{n,i-1}(x), b_{n,i}(x), b_{n,i+1}(x), \ldots$ of bits.

Remark: The sequence $\ldots, b_{n,i-1}(x), b_{n,i}(x), b_{n,i+1}(x), \ldots$ of bits, can also be defined as follows. Given an integer n as above define the function

$$f_n : QR_n \longrightarrow QR_n : x \longrightarrow f_n(x) = x^2 \bmod n.$$

Further, let the functions f_n^i be defined as follows:

$$f_n^i(x) = \begin{cases} f_n(x) & \text{if } i = 1 \\ f_n(f_n^{i-1}(x)) & \text{if } i > 1 \end{cases}$$

For each n, and each $x \in QR_n$ define the bits

$$b'_{n,i}(x) = B_n(f_n^i(x)),$$

where for $x \in QR_n$,

$$B_n(x) = \text{par}(x).$$

It is easy to show that for all n, x as above

$$b'_{n,i}(x) = b_{n,i}(x)$$

Definition 3.4 *A polynomial size circuit $C = \{C_k : k \geq 1\}$ has a $1/P$-advantage for predicting sequences of bits of length $Q(k)$ produced by the generator $QRGEN$ for the family N, abbreviated by $APR(C, N, Q, 1/2 + 1/P)$, if for all but a finite number of indices $k \in I$ the following property holds for all $n \in N_k$,*

$$Pr[C_k(b_{n,0}(x), \ldots, b_{n,Q(k)-1}(x)) = b_{n,-1}(x)] \geq \frac{1}{2} + \frac{1}{P(k)}. \tag{22}$$

Theorem 3.13 *For all polynomials P,*

$$(\exists C)(\exists Q)APR(C, N, Q, 1/2 + 1/P) \Rightarrow (\exists C)APAR(C, N, 1/2 + 1/P)$$

Proof: Let P, Q be polynomials and C a polynomial size circuit such that inequality (3.4) holds. Define a new polynomial size circuit via the equation below:

$$C'_k(x) = \text{par}(C_k(b_{n,0}(x),\ldots,b_{n,Q(k)-1}(x)) \qquad (23)$$

Notice that

$$C_k(b_{n,0}(x),\ldots,b_{n,Q(k)-1}(x)) = b_{n,-1}(x) \Rightarrow C'_k(x) = \text{par}(\sqrt{x} \bmod n)$$

One can then verify easily that the circuit C' satisfies the inequality in definition (3.1).•

Remark: Theorem 3.13 will be further improved in section 5.

EXERCISES

In the exercises below the notation of subsection 3.8 is used.

1: The **location function**, denoted by loc_n, is defined by:

$$loc_n(x) = \begin{cases} 0 & \text{if } x < n/2 \\ 1 & \text{if } x > n/2 \end{cases}$$

Show that for all $x \in Z_n^*$, $x < n/2 \Leftrightarrow 2x \bmod n$ is even. Moreover, for all $x \in QR_n$, $\text{par}(2\sqrt{x} \bmod n) = \text{loc}_n(\sqrt{x} \bmod n)$.

2: A polynomial size circuit $C = \{C_k : k \geq 1\}$ has a $1/P$-advantage for computing the location function for the family N, and this will be abbreviated by $ALOC(C, N, 1/2 + 1/P)$, if for all but a finite number of indices $k \in I$ the following property holds for all $n \in N_k$

$$Pr[x \in QR_n \ : \ C_k(n,x) = \text{loc}_n(\sqrt{x} \bmod n)] \geq \frac{1}{2} + \frac{1}{P(k)}$$

Show that for all polynomials P,

$$(\exists C)ALOC(C, N, 1/2 + 1/P) \Rightarrow (\exists C)APAR(C, N, 1/2 + 1/P)$$

Hint: Let $C = \{C_k \ : \ k \geq 1\}$ be a polynomial size circuit such that the above hypothesis $ALOC(C, N, 1/2 + 1/P)$ is true. It is required to find a circuit $C' = \{C'_k : k \geq 1\}$ such that $APAR(C', N, 1/2 + 1/P)$ is true. Use exercise 1 to show that the circuit $C'_k(n,x) = C_k(n, 4^{-1}x \bmod n)$, satisfies the requirements of the conclusion.

3: Define, by analogy to the definitions of the previously defined predicates $ALOC(C,N,1/2+1/P)$, $APAR(C,N,1/2+1/P)$, $AQR(C,N,1/2+1/P)$, $AQR(C,N,1-1/P)$, $APR(C,N,Q,1/2+1/P)$ the notions $ALOC(C,N,1/2+\epsilon)$, $APAR(C,N,1/2+\epsilon)$,$AQR(C,N,1/2+\epsilon)$,$AQR(C,N,1-\epsilon)$, where $\epsilon > 0$ is a constant. Show that for all circuits C

1. $(\exists\epsilon)ALOCR(C,N,1/2+\epsilon) \Rightarrow (\forall P)ALOCR(C,N,1/2+1/P)$

2. $(\exists\epsilon)APAR(C, N, 1/2 + \epsilon) \Rightarrow (\forall P)APAR(C, N, 1/2 + 1/P)$
3. $(\exists\epsilon)AQR(C, N, 1/2 + \epsilon) \Rightarrow (\forall P)AQR(C, N, 1/2 + 1/P)$
4. $(\exists P)AQR(C, N, 1 - 1/P) \Rightarrow (\forall\epsilon)AQR(C, N, 1 - \epsilon)$
5. $(\exists\epsilon)APR(C, N, Q, 1/2 + \epsilon) \Rightarrow (\forall P)APR(C, N, Q, 1/2 + 1/P)$
6. Prove corresponding versions of theorems 3.11,3.12 3.13 for the above defined notions of advantage.

4: Show that for any family N the following statements are equivalent:

1. $(\exists C)(\exists\epsilon)AQR(C, N, 1/2 + \epsilon)$
2. $(\exists C)(\exists\epsilon)AQR(C, N, 1 - \epsilon)$
3. $(\exists C)(\exists P)AQR(C, N, 1/2 + 1/P)$
4. $(\exists C)(\exists P)AQR(C, N, 1 - 1/P)$
5. $(\forall P)(\exists C)AQR(C, N, 1 - 1/P)$
6. $(\forall\epsilon)(\exists C)AQR(C, N, 1 - \epsilon)$

5: Let $x \in Z_n^*(+1)$ be fixed. Show that:

1. If $x \in QR_n$ then $QR_n = \{xs^2 : s \in Z_n^*\}$.
2. If $x \in QNR_n$ then $QNR_n = \{xs^2 : s \in Z_n^*\}$.

3.9 QUADRATIC RESIDUOSITY ASSUMPTION

As in subsection 3.8, throughout the present subsection n will range over integers which are the product of two distinct odd primes p, q such that $p \equiv q \equiv 3 \bmod 4$; $N = \{N_k : k \in I\}$ will denote a family of nonempty sets N_k of nonegative integers such that I is an infinite set of indices and for all $n \in N_k$ the integer n has binary length exactly k.

The notions of advantage defined in subsection 3.8 will now be altered in order to reflect the fact that this advantage is valid only for a certain fraction of the $n \in N_k$.

Definition 3.5 *A polynomial size circuit $C = \{C_k : k \geq 1\}$ has a $1/P$-advantage for computing the parity function for a fraction $1/P'$ of the integers in N_k, and this will be abbreviated by $\overline{APAR}(C, N, 1/P', 1/2 + 1/P)$, if for all but a finite number of indices $k \in I$ the following property holds*

$$\left|\left\{n \in N_k : Pr[x \in QR_n : C_k(n, x) = \mathrm{par}(\sqrt{x} \bmod n)] \geq \frac{1}{2} + \frac{1}{P(k)}\right\}\right| \geq \frac{|N_k|}{P'(k)}$$

The remaining overlined versions of the previously defined notions of advantage can be defined as above. In addition, one can prove the following theorem exactly as before.

Theorem 3.14 *For all polynomials P, P',*

1.$(\exists C)\overline{APAR}(C, N, 1/P', 1/2 + 1/P) \Rightarrow (\exists C)\overline{AQR}(C, N, 1/P', 1/2 + 1/P)$

2.$(\exists C, P)\overline{AQR}(C, N, 1/P', 1/2 + 1/P) \Rightarrow (\forall Q)(\exists C)\overline{AQR}(C, N, 1/P', 1 - 1/Q)$

3.$(\exists C, Q)\overline{APR}(C, N, Q, 1/P', 1/2 + 1/P) \Rightarrow$

$$(\exists C)\overline{APAR}(C, N, 1/P', 1/2 + 1/P)\bullet$$

Given a circuit C, an integer $n \in N_k$, and an $x \in Z_n^*(+1)$, $C_k(n, x)$ decides correctly if $x \in QR_n$ if and only if $C_k(n, x) = 1$ assuming that $x \in QR_n$, and $C_k(n, x) = 0$ assuming that $x \notin QR_n$. Recall that from definition 3.2

$$\frac{1}{2}(Pr[C_k(n, x) = 1 \mid x \in QR_n] + Pr[C_k(n, x) = 0 \mid x \notin QR_n]) =$$

$$= Pr[C_k(n, x) \text{ decides correctly if } x \in QR_n]$$

Definition 3.6 *The Quadratic Residuosity Assumption for the family $N = \{N_k : k \in I\}$, abbreviated $QRA(N)$, is the following statement: if $C = \{C_k : k \geq 1\}$ is a polynomial size, 0, 1-valued circuit and P, P' are polynomials with positive integer coefficients then for all but a finite number of indices $k \in I$ the following holds*

$$\left|\left\{n \in N_k \;:\; Pr[C_k(n, x) \text{ decides correctly if } x \in QR_n] \geq 1 - \frac{1}{P(k)}\right\}\right| \leq \frac{|N_k|}{P'(k)}$$

Theorem 3.15

$$QRA(N) \Leftrightarrow \neg(\exists P)(\exists P')(\exists Q)\overline{APR}(C, N, Q, 1/P', 1 - 1/P)$$

Proof: Assume that the hypothesis $QRA(N)$ is true, but that the conclusion $\neg(\exists P)(\exists P')(\exists Q)\overline{APR}(C, N, Q, 1/P', 1/2 + 1/P)$ fails. By theorem 3.14 there exist polynomials P, P' with positive coefficients, and a polynomial size circuit C such that $\overline{AQR}(C, N, 1/P', 1 - 1/P)$. Consider the polynomial $P''(k) = P'(k) + 1$. On the one hand, the definition of $\overline{AQR}(C, N, 1/P', 1 - 1/P)$, implies that for all but a finite number of indices $k \in I$ the following property holds

$$\left|\left\{n \in N_k : Pr[x \in QR_n : C_k(n, x) = \text{par}(\sqrt{x} \bmod n)] \geq 1 - \frac{1}{P(k)}\right\}\right| \geq \frac{|N_k|}{P'(k)}$$

On the other hand, $QRA(N)$ implies that for all but a finite number of indices $k \in I$ the following property holds

$$\left|\left\{n \in N_k : Pr[C_k(n,x) \text{ decides correctly if } x \in QR_n] \geq 1 - \frac{1}{P(k)}\right\}\right| \leq \frac{|N_k|}{P''(k)}$$

But this is a contradiction, because $N_k \neq \emptyset$. The proof of the other direction is similar.•

Remark: A typical example of a family N to which the above results apply is defined as follows: let N_k be the set of all integers n such that n is a product of two primes p, q such that $p \equiv q \equiv 3\text{mod}4$ and $||p| - |q|| \leq 1$, where $|p|$ (respectively $|q|$) is the binary length of p (respectively q). The quadratic residuosity assumption for this family is abbreviated by QRA.

EXERCISES

1: Give the proof of theorem 3.14.

2: Define explicitely the remaining overlined notions of advantage and show that each of them is implied by its corresponding nonoverlined counterpart.

3: (**Blum-Blum-Shub**) The location loc_n function defined in exercise 1 of subsection 3.8 gives rise to a pseudo-random generator. Define this generator and use $QRA(N)$ to show that it is unpredictable.

3.10 THE INDEX GENERATOR

Let g be a primitive root modulo the odd prime number p. Let $x \in QR_p$ be an arbitrary quadratic residue modulo p. It is known that $\text{index}_{p,g}(x) = 2t$, for some integer $t < (p-1)/2$ (see also exercise 4 in the subsection on indices). The principal square of x with respect to p, g, abbreviated $PQR(p,g,x)$, is the integer $g^t\text{mod}p$; the nonprincipal square root of x with respect to p, g, abbreviated $NPQR(p,g,x)$, is the integer $g^{t+(p-1)/2}$ mod$p$. For each p, g as above define the predicate $B_{p,g}$ as follows:

$$B_{p,g}(x) = \begin{cases} 1 & \text{if } x = PQR(p,g,x^2\text{mod}p) \\ 0 & \text{if } x = NPQR(p,g,x^2\text{mod}p) \end{cases}$$

It is now easy to see that

$$B_{p,g}(g^t\text{mod}p) = \begin{cases} 1 & \text{if } t < (p-1)/2 \\ 0 & \text{if } t \geq (p-1)/2 \end{cases}$$

A very significant observation is that the existence of an efficient algorithm for computing the above defined function $B_{p,g}$, leads to an efficient algorithm for computing the function $\text{index}_{p,g}$ (see exercise 4 below.) The theorem below shows that the existence of an efficient algorithm to compute PQR leads to the existence of an efficient algorithm to compute the function $\text{index}_{p,g}$, something that will be used in the sequel. For each p let $|p|$ denote the binary length of p.

Theorem 3.16 *Suppose there exists an algorithm A running in time polynomial in $|p|$ such that for any odd prime p, any primitive root $g \in Z_p^*$, and any $x \in Z_p^*$,*

$$A(p,g,x) = PQR(p,g,x).$$

Then there exists an algorithm A' running in time polynomial in $|p|$ such that for any odd prime p, any primitive root $g \in Z_p^*$, and any $x \in Z_p^*$,

$$A'(p,g,x) = \mathrm{index}_{p,g}(x).$$

Proof: Assume that A is an algorithm that satisfies the hypothesis of the theorem. Then on input p, g, x the algorithm A', using the integer c as a counter, outputs the sequence d of bits, which constitutes the binary representation of $\mathrm{index}_{p,g}(x)$, and is defined as follows:

Input: p prime, g primitive root modulo p, $x \in Z_p^*$.

Step 1: Put $d = \emptyset$, $c = 0$.

Step 2: Test if $x \in QR_p$.

Step 3: Put $d = b(x)d$, $c = c+1$, where

$$b(x) = \begin{cases} 0 & \text{if } x \in QR_p \\ 1 & \text{if } x \notin QR_p \end{cases}$$

Step 4: Put

$$x \equiv \begin{cases} x & \text{if } x \in QR_p \\ xg^{-1}\mathrm{mod} p & \text{if } x \notin QR_p \end{cases}$$

Step 5: Put $x := A(p,g,x)$.

Output: If $c < |p| - 1$ then goto Step 2 with this new x, else output d and stop.

The proof that this algorithm works is easy.•

Throughout the rest of the present subsection p will range over odd primes. $N = \{N_k : k \in I\}$ will denote a family of nonempty sets N_k such that I is an infinite set of indices and for all $n \in N_k$ the integer n is an odd prime of binary length exactly k. **From now on and for the rest of this section the capital roman letters P, Q with subscripts or superscripts will range over nonzero polynomials with one indeterminate, positive coefficients, and degree ≥ 1, and the lowercase greek letters ϵ, δ with subscripts or superscripts will range over positive real numbers.**

Definition 3.7 *A polynomial size circuit $C = \{C_k : k \geq 1\}$ has a $1/P$-advantage for determining the index for the family N, and this will be abbreviated by $AIND(C, N, 1/2 + 1/P)$, if for all but a finite number of indices $k \in I$ the following property holds for all $p \in N_k$, and all primitive roots g modulo p,*

$$Pr[x \in Z_p^* : C_k(p,g,x) = \mathrm{index}_{p,g}(x)] \geq \frac{1}{2} + \frac{1}{P(k)}$$

Similarly, one can define the notion $AIND(C, N, 1 - 1/P)$.

Definition 3.8 *A polynomial size circuit $C = \{C_k : k \geq 1\}$ has a $(1/2 - 1/P)$-advantage for determining the index for the family N, and this will be abbreviated by $AIND(C, N, 1 - 1/P)$, if for all but a finite number of indices $k \in I$ the following property holds for all $p \in N_k$, and all primitive roots g modulo p,*

$$Pr[x \in Z_p^* : C_k(p, g, x) = \text{index}_{p,g}(x)] \geq 1 - \frac{1}{P(k)}$$

For technical reasons, to become apparent in the proofs below, the following notion will also be used.

Definition 3.9 *For any polynomial Q let $E(p, g, Q)$ denote the event:*

$$\text{index}_{p,g}(x) \in \left[1, \frac{p-1}{Q(k)}\right] \geq \frac{1}{2} + \frac{1}{P(k)}$$

Definition 3.10 *A polynomial size circuit $C = \{C_k : k \geq 1\}$ computes the indices which lie in the closed interval $[1, (p-1)/Q(k)]$ for primes p which belong to N_k with $1/P$-advantage, and this will be abbreviated by $IND(C, N, 1/Q, 1/P)$, if for all but a finite number of indices $k \in I$ the following property holds for all $p \in N_k$, and all primitive roots g modulo p,*

$$Pr_{E(p,g,Q)}\left[x \in Z_p^* : C_k(p, g, x) = \text{index}_{p,g}(x)\right] \geq \frac{1}{2} + \frac{1}{P(k)}$$

Theorem 3.17 *(***Blum-Micali***)*

$$(\exists C)(\exists P, Q) IND(C, N, 1/Q, 1/P) \Rightarrow (\forall P)(\exists C) AIND(C, N, 1 - 1/P)$$

Proof: Assume that P, Q are polynomials, and C is a polynomial size circuit such that the inequality of definition 3.10 holds. The circuit C' is defined as follows:

Input: $p \in N_k$, g primitive root modulo p, $x \in Z_p^*$.

Step 1: Guess an integer i such that

$$\text{index}_{p,g}(x) \in \left[\frac{i(p-1)}{Q(k)}, \frac{(i+1)(p-1)}{Q(k)}\right]$$

Step 2: Compute $x_i = xg^{-i(p-1)/Q(k)} \bmod p$.

Step 3: Compute $d = C_k(p, g, x_i)$

Output: If $x \equiv g^{d + i(p-1)/Q(k)} \bmod p$ then output $\text{index}_{p,g}(x_i) + i(p-1)/Q(k)$

else put $i = i + 1$ and goto Step 2.

The probability that the above circuit C' will give the wrong answer depends on the probability that the circuit C will give the wrong answer; in fact, this probability is $\leq 1 - 1/Q(k)$. Therefore, repeating the above algorithm a sufficient number of times the advantage will be amplified as much as desired.•

Definition 3.11 *A polynomial size circuit $C = \{C_k : k \geq 1\}$ has a $1/P$-advantage for computing the function $B_{p,g}$ for the family N, and this will be abbreviated by $AB(C, N, 1/2 + 1/P)$, if for all but a finite number of indices $k \in I$ the following property holds for all $p \in N_k$, and all primitive roots g modulo p,*

$$Pr[x \in QR_p \ : \ C_k(p, g, x) = B_{p,g}(x)] \geq \frac{1}{2} + \frac{1}{P(k)}$$

Definition 3.12 *A polynomial size circuit $C = \{C_k : k \geq 1\}$ has a $1/P$-advantage for computing the function PQR for the family N, and this will be abbreviated by $APQR(C, N, 1/2 + 1/P)$, if for all but a finite number of indices $k \in I$ the following property holds for all $p \in N_k$, and all primitive roots g modulo p,*

$$Pr[x \in QR_p \ : \ C_k(p, g, x) = PQR(p, x, g)] \geq \frac{1}{2} + \frac{1}{P(k)}$$

Definition 3.13 *A polynomial size circuit $C = \{C_k : k \geq 1\}$ has a $1/P$-advantage for computing the function PQR for indices which lie in the interval $[1, (p-1)/Q(k)]$, for the family N, abbreviated by $APQR(C, N, 1/Q, 1/2 + 1/P)$, if for all but a finite number of indices $k \in I$ the following property holds for all $p \in N_k$, and all primitive roots g modulo p,*

$$Pr_{E(p,g,Q)}[x \in QR_p : C_k(p, g, x) = PQR(p, x, g)] \geq \frac{1}{2} + \frac{1}{P(k)}$$

Theorem 3.18 *(***Blum-Micali***)*

$$(\exists C)(\exists P)AB(C, N, 1/2 + 1/P) \Rightarrow (\forall Q)(\exists C, P')APQR(C, N, 1/P', 1 - 1/Q)$$

Proof: Let C be a polynomial size circuit which computes the function $B_{p,g}$ with a $1/P$-advantage. For each $e \in QR_p$ let ϵ', ϵ'' denote the two square roots of e modulo the prime p. The function PQR^C computes the principal square root with the aid of the circuit C and is defined as follows:

$$PQR^C(p, g, e) = \begin{cases} e' & \text{if } C_k(p, g, \epsilon') > C_k(p, g, \epsilon'') \\ e'' & \text{if } C_k(p, g, e') < C_k(p, g, \epsilon'') \\ \mathtt{random}\{\epsilon', e''\} & \text{if } C_k(p, g, e') = C_k(p, g, \epsilon'') \end{cases}$$

Let Q be any polynomial, and let the polynomial P' be defined by $P'(k) = 4 \cdot P(k) \cdot Q(k)^2$. It will be shown that there exists a polynomial size circuit C' such that the property $APQR(C', N, 1/P', 1 - 1/Q)$ holds, i.e. for all but a finite number of $k \in I$, and all primitive roots g modulo p,

$$Pr_{E(p,g,Q)}[x \in QR_p : C_k(p, g, x) = PQR(p, g, x)] \geq 1 - \frac{1}{Q(k)}$$

The circuit C' is defined as follows:

Input: $p \in N_k$, g primitive root modulo p, $e \in QR_p$ such that $\text{index}_{p,g}(e) \leq (p-1)/P'(k)$.

Step 1: Compute the two square roots e', e'' of e modulo p.

Step 2: Put $m = P'(k) = 4 \cdot P(k) \cdot Q(k)^2$.

Step 3: Select m random integers $r_1, \ldots, r_m$ such that $2r_1, \ldots, 2r_m \leq p-1$.

Step 4: Compute $e_i \equiv eg^{2r_i} \bmod p$, where $i = 1, \ldots, m$.

Step 5: Compute $e_i' \equiv e'g^{r_i} \bmod p$, and $e_i'' \equiv e''g^{r_i} \bmod p$, where $i = 1, \ldots, m$.

Step 6: Compute the following two integers:

$$L'(p,g,e) = |\{1 \leq i \leq m \ : \ PQR^C(p,g,e_i) = e_i'\}|$$

$$L''(p,g,e) = |\{1 \leq i \leq m \ : \ PQR^C(p,g,e_i) = e_i''\}|$$

Output:

$$C_k'(e,p,g) = \begin{cases} e' & \text{if } L'(p,g,e') > L''(p,g,e'') \\ e'' & \text{if } L'(p,g,e') < L''(p,g,e'') \end{cases}$$

It remains to show that the above circuit C' works. Let $2s = \text{index}_{p,g}(e)$, $T = \{1 \leq i \leq m \ : \ 2s + 2r_i \leq p-1\}$, and $t = |T|$. It follows from exercise 3 that there exist at least t many i's such that

$$2r_i \in \left[\frac{(m-1)(p-1)}{m}, p-1\right].$$

However, the above closed interval is the rightmost subinterval of the partition

$$\left\{\left[\frac{i(p-1)}{m}, \frac{(i+1)(p-1)}{m}\right] \ : \ i \leq m\right\}$$

of the closed interval $[(p-1)/m, p-1]$, into closed subintervals each of length $(p-1)/m$. Since the integers $2r_1, \ldots, 2r_m$ are randomly chosen from the closed interval $[1, p-1]$, it follows that t must be small. Moreover, for all $i \in T$ if y is a square root of e then by exercise 2,

$$yg^{r_i} = PQR(p,g,e_i) \Leftrightarrow y = PQR(p,g,e).$$

Next, the following two cases can be considered.

Case 1: If $e' = PQR(p,g,e)$

In this case, using the fact that the circuit C has a $1/P$ advantage for computing the function $B_{p,g}$, the expected value of $L'(p,g,e)$ is $m/2 + m/P(k)$. Similarly, the expected value of $L''(p,g,e)$ is $m/2 - m/P(k)$. Thus, using the Weak Law of Large Numbers, with probability $\geq 1 - 1/Q(k)$, $L'(p,g,e) > L''(p,g,e)$.

Case 2: If $e'' = PQR(p,g,e)$

In this case, using the fact that the circuit C has a $1/P$ advantage for computing the function $B_{p,g}$, the expected value of $L'(p,g,e)$ is $m/2 - m/P(k)$. Similarly, the expected value of $L''(p,g,e)$ is $m/2 + m/P(k)$. Thus, using the Weak Law of Large Numbers, with probability $\geq 1 - 1/Q(k)$, $L''(p,g,e) > L'(p,g,e)$.

This completes the proof of the theorem.•

As an application of theorems 3.16 and 3.18 one obtains the following:

Theorem 3.19 (**Blum-Micali**)

$$(\exists C)(\exists P)AB(C,N,1/2+1/P) \Rightarrow (\exists C)(\exists P,Q)AIND(C,N,1/Q,1/2+1/P)$$

Proof: Apply the result of theorem 3.18 to the polynomial $Q(k) = 2k$ to find a polynomial $P'(k)$ and a polynomial size circuit C' such that

$$Pr_{E(p,g,Q)}[x \in QR_p \ : \ C'_k(p,g,x) = PQR(p,g,x)] \geq 1 - \frac{1}{2k}$$

Next, apply the algorithm of theorem 3.16, but use the circuit C'_k insted of the algorithm A used there. Call C'' the resulting circuit. As before, the circuit C'_k will be applied $|p| = k$ times. Each time C'_k will supply the correct answer with probability $\geq (1 - 1/2k)$. Thus, C''_k will supply the correct answer with probability

$$\geq \left(1 - \frac{1}{2k}\right)^k \approx \exp\left(\frac{-1}{2}\right)$$

It follows that there exists a polynomial $P(k)$ such that

$$\left(1 - \frac{1}{2k}\right)^k \geq \frac{1}{2} + \frac{1}{P(k)}$$

This completes the proof of the theorem.•

Given an odd prime p and a primitive root modulo p consider the function

$$f_{p,g} \ : \ Z_p^* \longrightarrow Z_p^* : x \longrightarrow f_{p,g}(x) = g^x \bmod p.$$

Further, let the functions $f^i_{p,g}$ be defined as follows:

$$f^i_{p,g}(x) = \begin{cases} f_{p,g}(x) & \text{if } i = 1 \\ f_{p,g}(f^{i-1}_{p,g}(x)) & \text{if } i > 1 \end{cases}$$

For each odd prime p, for each primitive root g modulo p and each $x \in Z_p^*$ define the bits

$$b_{p,g,i}(x) = B_{p,g}(f^i_{p,g}(x))$$

The index generator, abbreviated INDGEN, accepts as inputs the triples $< p,g,x >$, where p is an odd prime, g is a primitive root modulo p and $x \in Z_p^*$; the output is the infinite sequence $b_{p,g,0}(x), b_{p,g,1}(x), \ldots, b_{p,g,i}(x), \ldots$ of bits.

Definition 3.14 *A polynomial size circuit $C = \{C_k : k \geq 1\}$ has a $1/P$-advantage for predicting sequences of bits of length $Q(k)$ produced by $INDGEN$, for the family N, and this will be abbreviated by $APR(C, N, Q, 1/2 + 1/P)$, if for all but a finite number of indices $k \in I$ the following property holds for all $p \in N_k$, and all primitive roots g modulo p,*

$$Pr[C_k(b_{p,g,1}(x), \ldots, b_{p,g,Q(k)-1}(x)) = b_{p,g,0}(x)] \geq \frac{1}{2} + \frac{1}{P(k)}. \qquad (24)$$

Theorem 3.20 *For all polynomials P,*

$$(\exists C)(\exists Q)APR(C, N, Q, 1/2 + 1/P) \Rightarrow (\exists C)AB(C, N, 1/2 + 1/P)$$

Proof: Let Q be a polynomial and let C be a polynomial size circuit such the inequality in definition 3.14 holds. Define a new circuit C' as follows:

$$C'_k(x) = C_k(b_{p,g,1}(x), \ldots, b_{p,g,Q(k)-1}(x))$$

It is now easy to see that the circuit C' must satisfie the inequality of definition 3.11.•

Remark: Theorem 3.20 will be further improved in section 5.

EXERCISES

1: Complete the details of the proof of theorem 3.17.

2: Assume that $x \in QR_p$ and $2r + \text{index}_{p,g}(x) < p - 1$. Show that for any square root of y of x modulo p,

$$yg^r = PQR(p, g, xg^{2r}) \Leftrightarrow y = PQR(p, g, x).$$

3: If $1 \leq \text{index}_{p,g}(x) \leq (p-1)/m$ and $2 \leq 2r \leq p - 1$ then

$$2r + \text{index}_{p,g}(x) \geq p - 1 \Rightarrow 2r \geq \frac{(m-1)(p-1)}{m}.$$

4:(Blum-Micali) Repeat the proof of theorem 3.16 to show that if there exists a polynomial in $|p|$ time algorithm such that for any odd prime p, any primitive root $g \in Z_p^*$, and any $x \in Z_p^*$,

$$A(p, g, x) = B_{p,g}(x).$$

then there exists a polynomial in $|p|$ time algorithm A' such that for all p, g, x as above,

$$A'(p, g, x) = \text{index}_{p,g}(x).$$

Hint: Steps 1 - 4 remain exactly the same. The new step 5 is the following:

Step 5: Use the Adelman-Manders-Miller algorithm to compute the two square roots of x modulo p, say x', x''.

The new step 6 to replace the old step 5 is the following:

Step 6: Put

$$x = \begin{cases} x' & \text{if } A(p, g, x') = 1 \\ x'' & \text{if } A(p, g, x') = 0 \end{cases}$$

3.11 DISCRETE LOGARITHM ASSUMPTION

As in subsection 3.10, throughout the present subsection p will range over odd primes, $N = \{N_k : k \in I\}$ will denote a family of nonempty sets N_k of nonegative integers such that I is an infinite set of indices and for all $n \in N_k$ the integer n is an odd prime of length exactly k.

From the notions of advantage defined in subsection 3.10 one can define overlined notions, just like in subsection 3.9, as follows:

Definition 3.15 *A polynomial size circuit $C = \{C_k \ : \ k \geq 1\}$ has a $1/P$-advantage for computing the function $B_{p,g}$ for a fraction $1/P'$ of the primes in N_k, and this will be abbreviated by $\overline{AB}(C, N, 1/P', 1/2 + 1/P)$, if for all but a finite number of indices $k \in I$, for all $p \in N_k$, and all primitive roots g modulo p the following property holds*

$$\left|\left\{p \in N_k : Pr[x \in Z_p^* : C_k(p,g,x) = B_{p,g}(x)] \geq \frac{1}{2} + \frac{1}{P(k)}\right\}\right| \geq \frac{|N_k|}{P'(k)}$$

The remaining overlined versions of the previously defined notions of advantage can be defined as above. In addition, one can prove the following theorem exactly as before.

Theorem 3.21 *For all polynomials P, P',*

1.$(\exists C, P, Q)\overline{IND}(C, N, 1/Q, 1/P', 1/2 + 1/P) \Rightarrow$

$$(\forall P)(\exists C)\overline{AIND}(C, N, 1/P', 1 - 1/P)$$

2.$(\exists C, P)\overline{AB}(C, N, 1/P', 1/2 + 1/P) \Rightarrow$

$$(\exists C, P, Q)\overline{IND}(C, N, 1/Q, 1/P', 1/2 + 1/P)$$

3.$(\exists C, Q)\overline{APR}(C, N, Q, 1/P', 1/2 + 1/P) \Rightarrow$

$$(\exists C)\overline{AB}(C, N, 1/P', 1/2 + 1/P)\bullet$$

Definition 3.16 *For any circuit C_k let $F(p, g, C)$ denote the event:*

$$C_k(p, g, x) = \text{index}_{p,g}(x).$$

Definition 3.17 *The Discrete Logarithm Assumption for the family $N = \{N_k : k \in I\}$, abbreviated $DLA(N)$, is the following statement: if $C = \{C_k : k \geq 1\}$ is a polynomial size, 0, 1-valued circuit and P, P' are polynomials with positive integer coefficients then for all but a finite number of indices $k \in I$ the following holds*

$$\left|\left\{p \in N_k : (\forall g)\left(Pr[E(p,g,C)] \geq 1 - \frac{1}{P(k)}\right)\right\}\right| \leq \frac{|N_k|}{P'(k)},$$

where g ranges over primitive roots modulo p.

Now, it is not difficult to show that

Theorem 3.22

$$DLA(N) \Leftrightarrow \neg(\exists P)(\exists P')(\exists Q)\overline{APR}(C,N,Q,1/P',1-1/P)\bullet$$

Remark 1: A typical example of a family N to which the above results apply is defined as follows: let N_k be the set of all primes such that $|p| = k$, where $|p|$ (respectively $|q|$) is the binary length of p (respectively q). The discrete logarithm assumption for this family is abbreviated by DLA.

Remark 2: The DLA is related to the Pohlig-Hellman algorithm given in section 1.

EXERCISES

1: Give the proof of theorems 3.21, 3.22.

2: Define explicitely the remaining overlined notions of advantage and show that each of them is implied by its corresponding nonoverlined counterpart.

3.12 BIBLIOGRAPHICAL REMARKS

The linear congruence generator,$LGEN$, defined in subsection 3.2, is one of the most popular pseudo-random generators in use today, and is based on a scheme first devised by Lehmer. The general theorem 3.1 on the predictability of the linear congruence generator is the main result of [Pl]. Additional information on the linear congruence generator can be found in [Kn] (pp. 1 - 37).

The $1/p$-generator is due to Blum, Blum and Shub. The predictability of the $1/p$ generator proved in theorem 3.3,as well as the exercises at the end of the subsection are from [BBS].

The equivalence of factoring and computing square roots modulo a composite number (theorem 3.6) was first discovered by Rabin (see [Rab].) The periodicity of x^2mod$n$ is useful in the study of the security of the x^2mod$n$ generator. The results of subsection 3.6 appear in [BBS].

The definition of the probabilistic polynomial size circuit given in subsection 3.7 is based partly on the definition given in [AB].

A study of the security of the $x^2 \bmod n$ generator, as well as the reduction of its unpredictability to the Quadratic Residuosity Assumption can be found in [BBS]. Theorem 3.12 on amplifying the advantage in predicting quadratic residues is extracted from [GM]. The security of the $\mathrm{index}_{p,g}$ generator, as well as the reduction of its unpredictability to the Discrete Logarithm Assumption can be found in [BM].

It is interesting that both the Discrete Logarithm Assumption and the Quadratic Residuosity Assumption were first considered by Gauss in [Ga].

4 PUBLIC KEY CRYPTOSYSTEMS

4.1 INTRODUCTION

The systems presented in this section have been chosen in order to illuminate the recent developments in public key cryptosystems following the suggestions of Diffie and Hellman in [DH].

Subsection 4.2 sets the ground by giving all the necessary definitions appropriate to understanding the importance of public key cryptosystems. The RSA system is developed in subsection 4.3, the Rabin system in subsection 4.5, and the Merkle-Hellman system in subsection 4.7. Subsection 4.9 presents the Quadratic Residue System which is based on probabilistic encryption.

In addition, the security of the RSA and the Rabin systems is studied relative to the security of single RSA and Rabin bits respctively (see subsections 4.4 and 4.6 respectively.) The single iteration Merkle-Hellman system is not secure; Shamir's cryptanalytic attack is presented in subsection 4.8.

4.2 WHAT IS A PUBLIC KEY CRYPTOSYSTEM

Suppose that user S (**sender**) wants to transmit a given message P to another user R (**receiver**) via a certain communication channel in such a way that it will be very difficult to any unauthorized user to read the message P. To accomplish this task the sender resorts to **encryption** or **enciphering** of the message P i.e. he scrambles the original message P, also called **plaintext**, and transmits the resulting scrambled text, say C. The scrambled text C thus obtained from the plaintext P is also called **ciphertext**.

The receiver must now convert the ciphertext C back into the original plaintext P. This conversion process is also called **decryption** or **deciphering**. In addition, the encryption and decryption processes mentioned above are in fact efficient algorithms, called the **encryption algorithm** and the **decryption algorithm** respectively, transforming a given message into another one. The function E (respectively D) determined by the encryption (respectively decryption) algorithm is called **encryption** (respectively **decryption**) **function**.

An **interceptor** is a user other than the sender or the receiver who gets hold of the transmitted ciphertext C. An interceptor who tries to reconstruct the original plaintext P from the inercepted ciphertext C is called a **cryptanalyst**, and the deciphering analysis he applies is called **cryptanalysis** (see figure 1).

In order to make the cryptanalysis even more difficult the encryption and decryption functions depend on a set K of parameters, also called the **set of keys**; each $k \in K$ is called **key**.

Thus, a nonpublic key cryptosystem, abbreviated $NPKC$, consists of two families $\{E_k : k \in K\}, \{D_k : k \in K\}$, of encryption and decryption functions respectively such that

1. For all $k \in K$, E_k is the inverse of D_k.

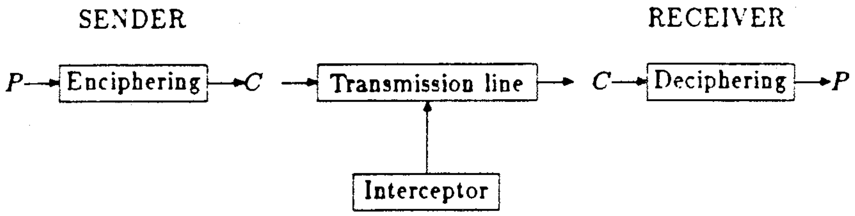


Figure 1: Message Transmission

2. For all $k \in K$, the algorithms E_k, D_k are efficient.
3. It is difficult to compute the plaintext P from the ciphertext $E_k(P)$ alone without prior knowledge of the decryption function D_k used.

To transmit messages the sender and the receiver agree in advance on a key, say k, chosen from the set K of keys; the sender transmits the ciphertext $E_k(P)$ to the receiver; the receiver uses $D_k(E_k(P)) = P$ in order to obtain the plaintext P (see figure 2.)

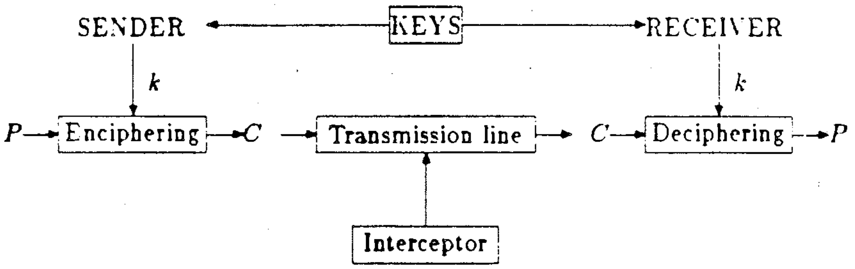


Figure 2: Nonpublic key cryptosystem

Example 4.1 The Vernam System: *Let both plaintexts and keys be represented by sequences of bits. Let $k = (k_0, \ldots, k_n)$ be the key agreed by the sender and the receiver. Let $\oplus$ represent modulo 2 addition between bits. In this system*

$E_k = D_k$ and for any plaintext $P = (P_0, \ldots, P_n)$,

$$E_k(P) = (k_0 \oplus P_0, \ldots, k_n \oplus P_n),$$

*Clearly, the Vernam system requires a key of length at least as as long as the message transmitted. This is accomplished by providing the key in a long enough tape; the section of the tape used is then discarded (***one-time-pad***).*

As was noted above, a nonpublic key cryptosystem requires the in advance exchange of a key between the sender and the receiver. However, such a limitation is indeed impractical for today's electronic communication requirements. A public key cryptosystem, abbreviated PKC, overcomes this limitation by allowing the existence of a private file as well as a public file (see table 3). Thus, for each user U, the public file of U is made available to all potential users; each such public file includes the encryption function E_U. However, the private file of U is known only to U itself and consists of the decryption function D_U. Moreover, the construction of the encryption and decryption functions is based on the notion of **trapdoor function**. Loosely speaking, a trapdoor function is a function f such that the following properties hold:

USER	PUBLIC FILE	PRIVATE FILE
A	E_A	D_A
B	E_B	D_B
C	E_C	D_C
...	...	...
...	...	...
...	...	...

Figure 3: The Files in a PKC

1. f is easy to compute.

2. f^{-1} is difficult to compute.

3. f^{-1} is easy to compute when a trapdoor or trick becomes available.

A function f satisfying only (1),(2) above is also called $1-1$, **one-way**.

Consequently, a PKC consists of two families $\{E_U\}, \{D_U\}$, where U ranges over the set of all potential users, of encryption and decryption functions respectively such that

1. For all U, E_U is the inverse of D_U.

2. For all U, E_U is in the public file.

3. For all U, E_U is a trapdoor function.

To transmit a plaintext P the sender S transmits the ciphertext $E_R(P)$ to the receiver, where E_R is the public encryption function of the receiver R. The receiver uses $D_R(E_R(P)) = P$ in order to obtain the plaintext P (see also figure 4).

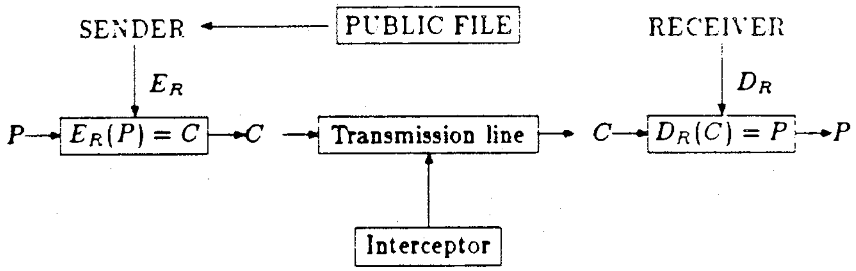


Figure 4: Public key cryptosystem

4.3 THE RSA SYSTEM

The first system to be examined is called RSA, named after the initials of the last names of its three inventors: Rivest, Shamir and Adelman. In the RSA system each user selects a pair p, q of distinct odd primes, that he keeps secret, and publicizes $N = p \cdot q$; further, each user chooses integers $e, d < N$ such that

$$\gcd(e, \varphi(N)) = 1, \ e \cdot d \equiv 1 \bmod \varphi(N)$$

The encryption and decryption functions respectively are

$$E(x) = x^e \bmod N, \ D(x) = x^d \bmod N$$

Figure (5) describes the RSA system.

Since N is the product of the two primes p, q, $\varphi(N) = (p-1)\cdot(q-1)$. Thus, any prime $e > \max(p, q)$ will be relatively prime to $\varphi(N)$. Using the Euclidean algorithm one can now determine an integer d such that $e \cdot d \equiv 1 \bmod N$. It follows from results of the section on Number Theory that both RSA encryption and RSA decryption are easy (see subsection on modular exponentiation.) It remains to show that the functions E, D are the inverse of each other i.e. to show that for all $x \in Z_N^*$,

$$E(D(x)) = x, \ D(E(x)) = x. \tag{1}$$

USER	PUBLIC FILE	PRIVATE FILE
1	$e_1, N_1 = p_1 \cdot q_1$	d_1, p_1, q_1
2	$e_2, N_2 = p_2 \cdot q_2$	d_2, p_2, q_2
3	$e_3, N_3 = p_3 \cdot q_3$	d_3, p_3, q_3
...	...	...
...	...	...
...	...	...

Figure 5: The RSA System

To prove (1) notice that

$$E(D(x)) = E(x^d \bmod N) = x^{e \cdot d} \bmod N.$$

However, $e \cdot d \equiv 1 \bmod \varphi(N)$. Hence, there exists an integer k (computed easily via the Euclidean algorithm) such that

$$e \cdot d = 1 + k \cdot \varphi(N)$$

It follows that

$$E(D(x)) = x^{e \cdot d} \bmod N \equiv x^{1 + k \cdot \varphi(N)} \bmod N$$

$$\equiv x \cdot x^{k \cdot \varphi(N)} \bmod N \equiv x \cdot (x^{\varphi(N)})^k \bmod N \equiv x \bmod N,$$

using the Euler-Fermat theorem.

4.4 RSA BITS

In studying the security of RSA it is reasonable to examine specific bits of the transmitted message. One might hope that it might be easier for a cryptanalyst to devise an algorithm that will output a specific bit of the original message, given the encrypted message. To be more specific the present subsection is motivated by the following

Question: If a cryptanalyst knows an efficient algorithm which given as input an RSA message $x^e \bmod N$ (of a specific instance of RSA) will output a certain bit of the original message x, can he devise an efficient algorithm which given as input an RSA message $x^e \bmod N$ (of the same instance of RSA) will output the whole message x?

Nevertheless, it might come as a surprise that for specific bits (to be studied below) devising an algorithm that will output a specific bit of the original message, given the encrypted message, is just as difficult as devising an algorithm that will output the entire original message, given the encrypted message.

If the representation of the integer N in the binary system is

$$N = \sum_{i=0}^{n-1} N_i \cdot 2^i$$

then let $\text{bit}(N)$ denote the sequence $N_{n-1} \ldots N_0$. Conversely, given a sequence $S = N_{n-1} \ldots N_0$ of bits let the **representation** of S, abbreviated $\text{rep}(S)$ be

$$\text{rep}(S) = \sum_{i=0}^{n-1} N_i \cdot 2^i.$$

For any instance N, e of RSA define the following bit functions:
Location Function:

$$loc_{N,e}(x^e \bmod N) = \begin{cases} 0 & \text{if } x < N/2 \\ 1 & \text{if } x > N/2 \end{cases}$$

s-th Bit Function:

$$\text{bit}^s_{N,e}(x^e \bmod N) = x_s,$$

where $\text{bit}(x) = x_{n-1} \ldots x_s \ldots x_0$.

As a special case one obtains the
Last Bit Function:

$$\text{bit}^0_{N,e}(x^e \bmod N) = \begin{cases} 0 & \text{if } x \text{ is even} \\ 1 & \text{if } x \text{ is odd} \end{cases}$$

For any odd integer N such that $\text{bit}(N) = N_{n-1} \ldots N_0$, it makes sense to define the **significant position** of N, abbreviated $s(N)$, by

$$s(N) = \text{ the largest } k \text{ such that } N_{k+1} = 0 < N_k = \cdots = N_0 = 1.$$

Notice that since N is odd, $s(N) \geq 1$.

The following result formalizes and answers the question stated above.

Theorem 4.1 (Goldwasser-Micali-Tong) *Given any instance N, e of RSA and any $0 \leq s \leq s(N)$ the following statements are equivalent*

1. *There is an efficient algorithm A such that*

$$A(x^e \bmod N) = x, \text{ for all } x \in Z_N^*.$$

2. *There is an efficient algorithm computing the function $\text{bit}^0_{N,e}$.*
3. *There is an efficient algorithm computing the function $loc_{N,e}$.*

4. There is an efficient algorithm computing the function $\mathrm{bit}^{s}_{N,e}$.

Proof: Fix any $0 \leq s \leq s(N)$. It is obvious that (1) implies each of the statements (2), (3) and (4). Since N is odd $N, 2^e$ are relativily prime; hence there exists an integer I such that $I \cdot 2^e \equiv 1 \bmod N$ (such an I can be computed using the Euclidean algorithm.)

Proof of (2) $\Leftrightarrow$ (3): It is clear that for all $x \in Z^*_N$,

$$x < \frac{N}{2} \Leftrightarrow 2x \bmod N \text{ is even} \tag{2}$$

It follows from (2) that

$$loc_{N,e}(x) = \mathrm{bit}^0_{N,e}(2^e \cdot x \bmod N), \tag{3}$$

$$\mathrm{bit}^0_{N,e}(x) = loc_{N,e}(I \cdot x \bmod N), \tag{4}$$

(see exercise 1). Now, the proof of (2) $\Leftrightarrow$ (3) can be completed easily.

The rest of the proof will require the following simple lemma, whose proof is left as an exercise (see exercise 2).

Lemma 4.1 *For* I, N, e *as above the following statements hold*

1. $N - x^e \equiv (N - x)^e \bmod N$

2. If x *is even then* $I \cdot x^e \equiv (\frac{x}{2})^e \bmod N$

3. If x *is odd then* $I \cdot (N - x^e) \equiv (\frac{N-x}{2})^e \bmod N$

From now on and for the rest of the proof of the present theorem the subscripts of $\mathrm{bit}^s_{N,e}$ **will be omitted.** For any sequence $u = u_{n-1}, \ldots, u_0$ of bits let $\ell(u) = n$ denote the length of u and let $u \uparrow i$ denote the sequence $u_{n-1}, \ldots, u_{n-i}$, i.e. the sequence consisting of the first i bits of u. Hence, if $i \geq \ell(u)$ then $u \uparrow i = u$. For any sequences u, u' of bits let $u \Delta u'$ denote the last $\ell(u')$ bits in the binary representation of the number $\mathrm{rep}(u) - \mathrm{rep}(u')$, where $\mathrm{rep}(u) \geq \mathrm{rep}(u')$; further, let $u \frown u'$ denote the concatenation of u, u' i.e. the sequence obtained from u by adjoining at the end the bits of u'. It is then easy to prove (see exercise 3) that

Claim 1: For any $x < N/2$ there exists a sequence of bits w such that

$$\mathrm{bit}(N - 2x) = w \frown [\,\mathrm{bit}(N) \Delta (\,\mathrm{bit}(x) \frown 0)]$$

Proof of (2) $\Rightarrow$ (1)

Let A be the efficient algorithm computing the last bit function. The idea of the proof is based on repeating the following algorithm $\ell(\,\mathrm{bit}(N))$ times:

Input: $x^e \bmod N$

Step 1: Compute $b = A(x^e \bmod N)$

Step 2:

1. If $b = 0$ then compute $I \cdot x^e \bmod N = (\frac{x}{2})^e \bmod N$

2. If $b = 1$ then compute $I \cdot (N - x^e) \bmod N = (\frac{N-x}{2})^e \bmod N$

Step 3: Use the number computed in Step 2 as new input, and repeat the process.

The sequence of bits given in successive applications of Step 1, constitute the binary representation of x.

The formal aspects of the proof are given in the sequel. Let $n = \ell(\text{ bit}(N))$. Define r_i, a_i, t_i, where $i = 1, \dots, n$, by induction as follows. Let

$$r_1 = x^e \bmod N; a_i = \text{ bit}^0(r_i),$$

$$r_i = \begin{cases} I \cdot r_{i-1} \bmod N & \text{if } a_{i-1} = 0 \\ I \cdot (N - r_{i-1}) \bmod N & \text{if } a_{i-1} = 1 \end{cases}$$

Also, define by reverse induction $t_n = a_n$ and

$$t_{i-1} = \begin{cases} t_i \frown 0 & \text{if } a_i = 0 \\ \text{bit}(N) \triangle [t_i \frown 0] & \text{if } a_i = 1 \end{cases}$$

Clearly, for all i, $\ell(t_i) = n - i + 1$. Also, for each i there exists a sequence u_i such that $u_i^e \equiv r_i \bmod N$. Put $v_i = \text{ bit}(u_i)$. Then one can prove, by reverse induction on i, that for all i there exists a sequence w_i such that

Claim 2: $v_i = w_i \frown t_i$.

Indeed, the case $i = n$ is immediate from the definitions. Assume that the claim is true for i, and let w_i be the sequence such that $v_i = w_i \frown t_i$. To find a w_{i-1} such that $v_{i-1} = w_{i-1} \frown t_{i-1}$. If on the one hand $a_{i-1} = 0$ then $r_i = I \cdot r_{i-1} \bmod N$. Thus,

$$u_{i-1}^e \equiv r_{i-1} \equiv 2^e \cdot r_i \equiv 2^e \cdot u_i^e \equiv (2 \cdot u_i)^e \bmod N.$$

Consequently, $2 \cdot u_i = u_{i-1}$. It follows from the induction hypothesis that

$$v_{i-1} = \text{ bit}(u_{i-1}) = \text{ bit}(u_i) \frown 0 = w_i \frown t_i \frown 0.$$

If on the other hand $a_{i-1} = 1$ then $r_i = I \cdot (N - r_{i-1}) \bmod N$. Thus,

$$u_{i-1}^e \equiv r_{i-1} \equiv N - 2^e \cdot r_i \equiv N - 2^e \cdot u_i^e \equiv$$

$$N - (2 \cdot u_i)^e \equiv (N - 2 \cdot u_i)^e \bmod N.$$

Consequently, $N - 2 \cdot u_i = u_{i-1}$. It follows from claim 1 that there exists a sequence w'_{i-1} such that

$$v_{i-1} = \text{ bit}(u_{i-1}) = \text{ bit}(N - 2 \cdot u_i) = w'_{i-1} \frown [\text{ bit}(N) \triangle (v_i \frown 0)]$$

Hence, the result follows easily from the induction hypothesis.

Finally, claim 2 implies that $w_1 = \emptyset$ and hence $v_1 = \text{bit}(x) = t_1$.

Proof of (4) $\Rightarrow$ (1)

Define $r_i, a_i, f_{i,s}, \ldots, f_{i,0}$, where $i = 1, \ldots, n$, by induction on i:

$$f_{1,s-1} = \cdots = f_{1,0} = 0,\ f_{i,s} = \text{bit}^s(r_i),$$

$$r_1 = x^e \bmod N,\ a_i = f_{i,0}.$$

r_i is now defined exactly as before i.e.

$$r_i = \begin{cases} I \cdot r_{i-1} \bmod N & \text{if } a_{i-1} = 0 \\ I \cdot (N - r_{i-1}) \bmod N & \text{if } a_{i-1} = 1 \end{cases}$$

Further, put

$$f_{i,s-1} \ldots f_{i,0} = \begin{cases} f_{i-1,s} \ldots f_{i-1,1} & \text{if } a_{i-1} = 0 \\ (1 - f_{i-1,s}) \ldots (1 - f_{i-1,1}) & \text{if } a_{i-1} = 1 \end{cases}$$

The sequence t_i is defined by reverse induction; one sets $t_n = f_{n,s} \ldots f_{n,0}$, and for $i \geq s+1$ one puts

$$t_{i-1} = \begin{cases} t_i \frown 0 & \text{if } a_{i-1} = 0 \\ \text{bit}(N) \Delta [t_i \frown 0] & \text{if } a_{i-1} = 1 \end{cases}$$

Clearly, for all $1 \leq i \leq n - s$, $\ell(t_{n+1-i}) = s + i$. Hence, $\ell(t_{s+1}) = n$. As before, let u_i be such that $u_i^e \equiv r_i \bmod N$. It will be shown by induction on $s+1 \geq i \geq 1$ that

Claim 3: $[f_{i,s} \ldots f_{i,0}] \upharpoonright i = [\text{last } s+1 \text{ bits of } \text{bit}(u_i)] \upharpoonright i$

Proof of Claim 3: The case $i = 1$ is trivial. Assume the claim is true for i. On the one hand, if $a_i = 0$ then

$$[f_{i+1,s} \ldots f_{i+1,0}] \upharpoonright (i+1) = f_{i+1,s} f_{i,s} f_{i,s-1} \ldots f_{i,s-i+1} =$$
$$\text{bit}^s(r_{i+1}) f_{i,s} f_{i,s-1} \ldots f_{i,s-i+1} =$$
$$[(s+1) - \text{st bit of } \text{bit}(u_{i+1})] f_{i,s} f_{i,s-1} \ldots f_{i,s-i+1}.$$

The claim now follows from the fact that $f_{i,0} = 0$, $u_i = 2u_{i+1}$ (see claim 1). On the other hand, if $a_i = 1$ then

$$[f_{i+1,s} \ldots f_{i+1,0}] \upharpoonright (i+1) =$$
$$[(s+1) - st \text{ bit of } \text{bit}(u_{i+1})](1 - f_{i,s}) \ldots (1 - f_{i,s-i+1}).$$

The claim now follows from the fact that $f_{i,0} = 1$, $u_i = N - 2u_{i+1}$ (see claim 1).

Next, one can show as in claim 2 above that for all $i \geq s+1$ there exists a sequence w_i such that $v_i = w_i \frown t_i$. In particular, $v_{s+1} = t_{s+1}$ and hence, $u_{s+1} = \text{bit}(t_{s+1})$. It follows from the definition of u_i that for all i,

$$2 \cdot u_{i+1} \equiv u_i \bmod N \text{ or } 2 \cdot u_{i+1} \equiv -u_i \bmod N.$$

In particular, since $x = u_1$

$$2^s \cdot u_{s+1} \equiv x \bmod N \text{ or } 2^s \cdot u_{s+1} \equiv -x \bmod N.$$

It is now clear that if one puts $y \equiv 2^s \cdot \text{rep}(t_{s+1}) \bmod N$ then

$$x = \begin{cases} y & \text{if } y^e \equiv x^e \bmod N \\ N - y & \text{if } y^e \not\equiv x^e \bmod N \end{cases}$$

The above recursive construction can be easily converted into an efficient algorithm for computing x from $x^e \bmod N$•

EXERCISES 1: Give the proof of equations (3) and (4).

2: Give the proof of lemma 4.1. **Hint:** Use the fact that e is odd and that $I \cdot 2^e \equiv 1 \bmod N$.

3: Give the proof of Claim 1.

4.5 THE RABIN SYSTEM

The main strength of *RSA* is based on the (supposed) difficulty of factoring. Thus, if a cryptanalyst knows how to factor efficiently he will also be able to break *RSA*. However, it is not known if the converse of this last statement is true. Rabin, in an attempt to resolve this intricate situation, has proposed a public key cryptosystem, to be described below, for which the problem of factoring is equivalent to that of breaking his system.

In the Rabin system each user selects a pair p, q of distinct odd primes, that he keeps secret, and publicizes $N = p \cdot q$; further, each user chooses an integer $b < N$. The **encryption** function is

$$E_{N,b}(x) = x \cdot (x + b) \bmod N.$$

The **decryption** function $D_{N,b}$ supplies for each given encoded message m a solution u (there are four possible such solutions) of the quadratic equation $x \cdot (x + b) \equiv m \bmod N$. Figure (6) describes the Rabin system.

It is clear that the encryption $E_{N,b}(x) \equiv x \cdot (x + b) \bmod N$ requires one addition, one multiplication and one division modulo N. Decryption is also easy if the factorization $N = p \cdot q$ of N is known. Indeed, given an encrypted message m (such that $p, q \not| m$) use the Adelman, Manders and Miller algorithm to compute the roots r, s of the congruences $x \cdot (x + b) \equiv m \bmod p$ and $x \cdot (x + b) \equiv m \bmod q$ respectively. Next, use the Euclidean algorithm to compute integers k, l such that $k \cdot p + l \cdot q = 1$. It is now easy to see that $lqr + kps$ is a solution of the congruence $x \cdot (x + b) \equiv m \bmod N$. Further, it is easy to show that the functions $E_{N,b}, D_{N,b}$ are the inverse of each other.

It will simplify the remaining proofs if one notices that the congruence $x \cdot (x + b) \equiv m \bmod N$ has a solution if and only if the congruence $y^2 \equiv m + \frac{b^2}{4} \bmod N$

USER	PUBLIC FILE	PRIVATE FILE
1	$b_1, N_1 = p_1 \cdot q_1$	p_1, q_1
2	$b_2, N_2 = p_2 \cdot q_2$	p_2, q_2
3	$b_3, N_3 = p_3 \cdot q_3$	p_3, q_3
...	...	...
...	...	...
...	...	...

Figure 6: The Rabin System

has a solution. To see this last claim, one merely has to complete the squares in the congruence $x^2 + x \cdot b \equiv m \bmod N$; this can be done since N is odd; one merely has to define 4^{-1} $mod N$, the inverse of 4 modulo N. Thus, from now on only congruences of the form $x^2 \equiv m \bmod N$ will be considered.

As promised, it remains to show that decryption is equivalent to factorization. This is proved in the theorem below.

Theorem 4.2 (**Rabin's Factorization Theorem**) *Let N be the product of two odd primes. Then the following statetements are equivalent:*

1. *There is an efficient algorithm A such that for all $m < N$, $A(N, m)$ is a solution of the congruence $x^2 \equiv m \bmod N$.*

2. *There is an efficient algorithm for factoring N.*

Proof: The proof of (2) $\Rightarrow$ (1) was given in the above discussion. Thus, it remains to prove (1) $\Rightarrow$ (2). Choose at random an integer a such that $\gcd(a, N) = 1$ and let $m \equiv a^2 \bmod N$. If $u = A(N, m)$ then both a, u are solutions of the congruence $x^2 \equiv m \bmod N$. So, on the one hand if $u \notin \{a, N - a\}$ then $\gcd(N, u + a)$ is a prime factor of N; on the other hand if $u \in \{a, N - a\}$ then choose another a and repeat the above procedure. Since, with probability $1/2$, $u \notin \{a, N - a\}$, it is expected that after two trials one will be able to factor N. More details on the proof can be found in subsection 3.5 •

A closer examination of the proof of the previous theorem can also show the following

Theorem 4.3 (**Rabin**) *Let A be an efficient algorithm such that for any N which is the product of two odd primes, $A(N, m)$ outputs in $F(N)$ steps a solution of the congruence $x^2 \equiv m \bmod N$ with probability at least $\frac{1}{e(N)}$. Then there exists an efficient algorithm B such that for any N which is the product of two odd primes, $B(N)$ will output the factors of N in at most $2 \cdot e(N) \cdot F(N) + 2 \cdot log_2 N$ steps* •

EXERCISES

1: Prove theorem 4.3 using an argument similar to that of theorem 4.2.

4.6 RABIN BITS

Just like in the case of the *RSA* system it is reasonable to examine the secutity of specific bits of messages transmitted via the Rabin system. To be more specific the present subsection is motivated by the following

Question: If a cryptanalyst knows an efficient algorithm which given as input a Rabin message $x^2 \bmod N$ (of a specific instance of Rabin's system) will output a certain bit of the original message x, can he devise an efficient algorithm which given as input a Rabin message $x^2 \bmod N$ (of a specific instance of Rabin's system) will output the whole message x?

Without further ado the notation of subsection 4.4 will be used in the present subsection.

Each $x \in QR_N$ has exactly four square roots; let x^+ (respectively x^-) denote the square root of x which is $< N/2$ and such that the Jacobi symbol of x^+ (respectively of x^-) with respect to N is $+1$ (respectively -1.) For any instance N of Rabin's system define the following bit functions whose domain is the set QR_N of quadratic residues modulo N.

Parity Function:

$$Par_N(x) = \text{ parity of } x^+,$$

Parity Comparison Function:

$$CPar_N(x) = \begin{cases} 0 & \text{if parity of } x^+ = \text{parity of } x^- \\ 1 & \text{if parity of } x^+ \neq \text{parity of } x^-. \end{cases}$$

The following result formalizes and answers the question stated above.

Theorem 4.4 (**Goldwasser-Micali-Tong**) *Given any N which is the product of two odd primes p, q such that either $p \equiv q \equiv 1 \bmod 8$ or $p \equiv q \equiv -1 \bmod 8$ or $p \equiv q \equiv 3 \bmod 8$ or $p \equiv q \equiv -3 \bmod 8$ the following statements are equivalent*

1. *There is an efficient algorithm for factoring N*

2. *There is an efficient algorithm computing the function Par_N.*

3. *There is an efficient algorithm computing the function $CPar_N$.*

Proof: Fix any N as above. It is obvious that (1) implies each of the statements (2) and (3). Since N is odd $N, 4$ are relativily prime; hence there exists an integer I such that $I \cdot 4 \equiv 1 \bmod N$ (such an I can be computed using the Euclidean algorithm.)

Proof of (2) ⇒ (1): The proof is similar to that of theorem 4.1. The following algorithm factors N:

```
Input: $N$
Step 1: Choose $a < N/2$ at random such that $(a|N) = -1$.
Step 2: Compute $r_1 \equiv a^2 \bmod N, a_1 = Par_N(r_1)$.
Step 3: Compute the length $n = \ell(\text{ bit}(N))$ of $N$.
Step 4: For $i = 1$ to $n$ compute
            $r_i \equiv I \cdot r_{i-1} \bmod N, \quad a_i = Par_N(r_i)$
Step 5: Compute $t_n = a_n$
Step 6: For $i = n$ down to 2 compute
            $t_{i-1} = \begin{cases} t_i \frown 0 & \text{if } a_{i-1} = 0 \\ \text{bit}(N) \Delta [t_i \frown 0] & \text{if } a_{i-1} = 1 \end{cases}$
Output: $\gcd(a + \text{rep}(t_1), N)$.
```

For each i let u_i be the unique root of $x^2 \equiv r_i \bmod N$ such that $u_i < N/2$, $(u_i|N) = +1$. However, recall the following properties of the Jacobi symbol:

$$(-1|N) = (-1)^{(N-1)/2}, \ (2|N) = (-1)^{(N^2-1)/8}.$$

Hence, for the N considered in the present theorem $(-1|N) = (2|N) = +1$. Using this and arguing as in the proof of theorem 4.4 one can show that

$$a_{i-1} = 0 \Rightarrow u_{i-1} = 2 \cdot u_i,$$

$$a_{i-1} = 1 \Rightarrow u_{i-1} = N - 2 \cdot u_i.$$

For each i let $v_i = \text{rep}(u_i)$. As in the proof of theorem 4.4 it can be shown that for all i there exists a w_i such that $v_i = w_i \frown t_i$. In particular, $v_1 = t_1$. It follws that $\gcd(a + u_1, N)$ is a prime factor of N

The proof of (3) ⇒ (1) is left as an exercise to the reader •

EXERCISES

1: Complete the details of the proof of (2) ⇒ (1) in theorem 4.4.

2: Give the proof of (3) ⇒ (1) in theorem 4.4.

4.7 THE MERKLE-HELLMAN SYSTEM

In the Merkle-Hellman system each user selects a pair w, m of positive integers, that he keeps in his private file, such that $\gcd(w, m) = 1$; w is called the **multiplier** and m is called the **modulus**. In addition, each user keeps in his private file a **superincreasing** sequence $a' = (a'_1, \ldots, a'_n)$ i.e. a sequence that satisfies

$$a'_i > \sum_{j=i+1}^{n} a'_j \text{ for all } i \geq 1, \text{ and } m > \sum_{j=1}^{n} a'_j.$$

The user publicizes the sequence $a = (a_1, \ldots, a_n)$ which is defined by

$$a_i \equiv w \cdot a'_i \bmod m \text{ for all } i \geq 1.$$

A message $x = (x_1, \ldots, x_n)$ (which is a sequence of 0,1 bits) is encrypted via the encryption function

$$E(x) = \sum_{i=1}^{n} x_i \cdot a_i.$$

The **decryption** function D supplies for each given encoded message S a solution u of the equation

$$S = \sum_{i=1}^{n} u_i \cdot a_i. \tag{5}$$

Figure (7) describes the Merkle-Hellman system.

USER	PUBLIC FILE	PRIVATE FILE
1	$a(1) = (a_1(1), \ldots, a_{n_1}(1))$	$w(1), m(1), a'(1) = (a'_1(1), \ldots, a'_{n_1}(1))$
2	$a(2) = (a_1(2), \ldots, a_{n_2}(2))$	$w(2), m(2), a'(2) = (a'_1(2), \ldots, a'_{n_2}(2))$
3	$a(3) = (a_1(3), \ldots, a_{n_3}(3))$	$w(3), m(3), a'(3) = (a'_1(3), \ldots, a'_{n_3}(3))$
...	...	...
...	...	...
...	...	...

Figure 7: The Merkle-Hellman System

Equation (5) is based on a knapsack problem and is in general difficult to solve. However, the following lemma indicates how one can solve efficiently knapsack problems for superincreasing sequences.

Lemma 4.2 *Let $a' = (a'_1, \ldots, a'_n)$ be a superincreasing sequence of positive integers and let $S' > 0$. Then the following equation has at most one solution $x = (x_1, \ldots, x_n) \in \{0,1\}^n$*

$$S' = \sum_{i=1}^{n} x_i \cdot a'_i. \tag{6}$$

In fact, equation (6) has a solution if and only if

$$S' \leq \sum_{i=1}^{n} a'_i.$$

Proof: The proof is straightforward. One merely needs to observe that for all $i = 1, \ldots, n$,

$$x_i = 1 \Leftrightarrow S' \geq a'_i + \sum_{j=i+1}^{n} x_j \cdot a'_j \bullet$$

It remains to show that encryption is easy. Indeed, when the user receives the encrypted message S he is supposed to solve equation (5) to obtain the original message $x = (x_1, \ldots, x_n)$. Instead, he computes w^{-1}, the inverse of w modulo m, and solves the equivalent knapsack problem

$$w^{-1} \cdot S \equiv \sum_{i=1}^{n} x_i \cdot a'_i \mod N. \tag{7}$$

Since the sequence $a' = (a'_1, \ldots, a'_n)$ is superincreasing, equation (7) can be solved easily using lemma 4.2.

An obvious generalization of the above system is the iterated Merkle-Hellman system, also considered by Merkle and Hellman. In such a system one successively applies pairs w^k, m^k of multipliers and moduli respectively (such that $\gcd(w^k, m^k) = 1$), where $k = 1, \ldots, r-1$ to the original vector a to obtain a sequence $a = a^0, a^1, \ldots, a^r$ of vectors satisfying

$$a^k = w^{k+1} * a^{k+1} \mod m^{k+1}, \text{ for } k = 0, \ldots, r-1,$$

(here, the symbol $*$ is used to indicate multiplication of a scalar with a vector.) The last vector a^r is chosen in advance to constitute a superincreasing sequence. For more details the reader should consult [MH].

4.8 THE SECURITY OF THE MERKLE-HELLMAN SYSTEM

Let $w, m, a' = (a'_1, \ldots, a'_n), a = (a_1, \ldots, a_n)$ be an instance of the Merkle-Hellman system. A cryptanalyst is in possession of the sequence a, but not of w, m, a'. In order to analyze the above instance, the cryptanalyst might try to compute a trapdoor pair for the sequence a i.e. a pair $\overline{w}, \overline{m}$ of integers such that the sequnce $\overline{a} = (\overline{a}_1, \ldots, \overline{a}_n)$ defined by

$$\overline{a}_i \equiv a_i \cdot \overline{w} \mod \overline{m} \tag{8}$$

is superincreasing and satisfies

$$\sum_{i=1}^{n} \overline{a}_i < \overline{m}. \tag{9}$$

It is clear from the argument in subsection 4.7 that any trapdoor pair could be used to decrypt easily any transmitted message of the above instance of the Merkle-Hellman system.

Dividing congruences (8), (9) by $\overline{m}$ one obtains that

$$\frac{\overline{a}_i}{\overline{m}} \equiv (a_i \cdot \frac{\overline{w}}{\overline{m}}) \bmod 1, \qquad (10)$$

$$\sum_{i=1}^{n} a_i \cdot \overline{r} \bmod 1 < 1, \qquad (11)$$

where $\overline{r} = \frac{\overline{w}}{\overline{m}}$ The function $(\overline{a}_i \cdot \overline{r}) \bmod 1$ is represented in figure 8. (Notice that for convenience the unit length in the horizontal axis is smaller than the unit length in the vertical axis.)

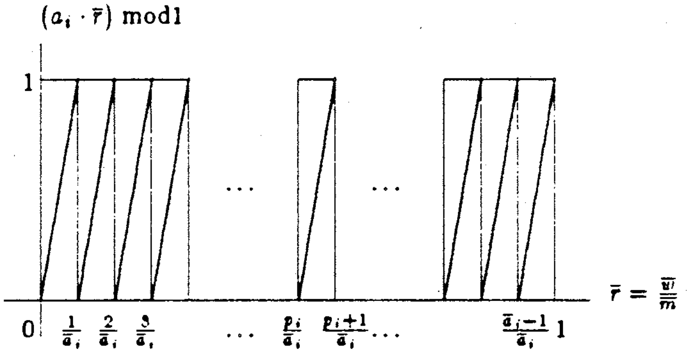


Figure 8: The i-th sawtooth function

In order to compute such a trapdoor pair one first determines a point $\overline{r}_0$ on the $\overline{r}$- axis such that inequality (11) is valid (of course such a point is guaranteed to exist because this is required in the construction of the Merkle-Hellman system). Hence, an interval $[r_1, r_2]$ must also exist such that for all points $\overline{r} \in [r_1, r_2]$ inequality (11) is valid. Let p_i = the p_i-th minimum of the i-th sawtooth curve. One obtains the following two systems of inequalities with the integral unknowns $p_1, \ldots, p_n$.

$$\begin{array}{ll} 1 \leq p_1 \leq a_1 - 1, & -\epsilon_2 \leq p_1/a_1 - p_2/a_2 \leq \epsilon'_2 \\ 1 \leq p_2 \leq a_2 - 1, & -\epsilon_3 \leq p_1/a_1 - p_3/a_3 \leq \epsilon'_3 \\ \vdots & \vdots \\ 1 \leq p_n \leq a_n - 1, & -\epsilon_n \leq p_1/a_1 - p_n/a_n \leq \epsilon'_n \end{array} \qquad (12)$$

where $\epsilon_1, \ldots, \epsilon_n, \epsilon'_1, \ldots, \epsilon'_n$ are the acceptable deviations to the right and to the left of p_1/a_1 respectively (the deviations should be chosen in such a way that inequality (11) will be valid). System (12) can be solved using Lenstra's integer programming algorithm (see [Len]).

Let p_1 be one of the values determined by the above procedure and let $\bar{r}_1, \ldots, \bar{r}_k$ be a list of all discontinuity points such that

$$\bar{r}_1, \ldots, \bar{r}_k \in \left[\frac{p_1}{a_1}, \frac{p_1+1}{a_1}\right),$$

arranged in increasing order, of all the sawtooth curves. Between any two such discontinuity points each sawtooth curve looks like a line segment; moreover, the linear segment corresponding to the i-th sawtooth curve is represented by the formula $\bar{r} \cdot a_i - q_i^t$, where $q_i^t =$ the number of minima of the i-th sawtooth curve which lie in the interval $(0, \bar{r}_t)$. Thus, for each $1 \leq t \leq k$ conditions (10), (11) can now be formulated as the following system of linear inequalities with unknown $\bar{r}$, such that $\bar{r}_t \leq \bar{r} < \bar{r}_{t+1}$:

$$\sum_{i=1}^{n} \bar{r} \cdot a_i - q_i^t < 1, \tag{13}$$

$$(\bar{r} \cdot a_i - q_i^t) > \sum_{i=1}^{j-1} (\bar{r} \cdot a_i - q_i^t), \text{ for } j = 1, \ldots, n. \tag{14}$$

The solution of the above system provides a subinterval of $[\bar{r}_t, \bar{r}_{t+1})$. Any $\bar{r} = \overline{w}/\overline{m}$ lying in this subinterval gives a trapdoor pair $\overline{w}, \overline{m}$. For more details the reader should consult [Sham1].

4.9 THE QUADRATIC RESIDUE SYSTEM

The quadratic residue system, abbreviated QRS, to be described below, replaces the notion of trapdoor function with the notion of probabilistic encryption i.e. to encrypt a given message the user will use the result of a sequence of coin tosses in order to scramble the original message.

Suppose that a sender S wants to send a binary message $M = (m_1 \cdots m_r)$ to receiver R. S obtains the numbers n, y, where $y \notin QR_n$, corresponding to R from the public file; n is the product of two odd primes p, q known only to R such that $p \equiv q \equiv 3 \bmod 4$ (see figure 9). S encrypts the message M by choosing a random sequence $x = (x_1, \ldots, x_r)$ of r elements of $Z_n^* = \{1 \leq x < n : \gcd(x, n) = 1\}$ and letting

$$E_n(x; M) = (y^{m_1} \cdot x_1^2 \bmod n, \ldots, y^{m_r} \cdot x_r^2 \bmod n).$$

Given $(e_1, \ldots, e_r)$ the receiver who knows the factorization of n reconstructs the message $M = D_n(e_1, \ldots, e_r) = (m_1, \ldots, m_r)$ via

$$m_i = \begin{cases} 1 & \text{if } e_i \in QR_n \\ 0 & \text{if } e_i \notin QR_n, \end{cases}$$

USER	PUBLIC FILE	PRIVATE FILE
1	$y_1 \notin QR_{n_1}, n_1 = p_1 \cdot q_1$	p_1, q_1
2	$y_2, \notin QR_{n_2}, n_2 = p_2 \cdot q_2$	p_2, q_2
3	$y_3, \notin QR_{n_3}, n_3 = p_3 \cdot q_3$	p_3, q_3
...	...	...
...	...	...
...	...	...

Figure 9: The Quadratic Residue System

where $i = 1, \ldots, r$.

The following definitions will be needed in the study of the security of the QRS. The **signature** of an integer $x \in Z_n^*$ is defined by

$$\sigma_n(x) = \begin{cases} 1 & \text{if } x \in QR_n \\ 0 & \text{if } x \notin QR_n \end{cases}$$

Thus, the above definition of the decryption function D_n implies that

$$D_n(e_1, \ldots, e_r) = (\sigma_n(e_1), \ldots, \sigma_n(e_r)).$$

The **r-signature** of the $r-$ tuple $x = (x_1, \ldots, x_r)$, where each $x_i \in Z_n^*$, is defined by

$$\sigma_{n,r}(x_1, \ldots, x_r) = (\sigma_n(x_1), \ldots, \sigma_n(x_r)).$$

If $\ell = (\ell_1, \ldots, \ell_r)$ is a given sequence of bits let

$$\Omega_{r,n,\ell} = \{(x_1, \ldots, x_r) \in (Z_n^*)^r : \sigma_{n,r}(x_1, \ldots, x_r) = \ell\}.$$

Given two sequences $a = (a_1, \ldots, a_r), b = (b_1, \ldots, b_r)$ of bits the **distance** between a, b, abbreviated dis(a, b), is the number of indices $1 \leq i \leq r$ such that $a_i \neq b_i$; a, b are called **adjacent** if their distance is equal to 1.

Recall the notation of the subsection on the Quadratic Residue Generator. $N = \{N_k : k \in I\}$ will denote a family of nonempty sets N_k of nonegative integers such that I is an infinite set of indices, and for all $n \in N_k$ the integer n has binary length exactly k; throughout the present subsection n will range over integers which are the product of two distinct odd primes p, q such that $p \equiv q \equiv 3$ mod4. The capital roman letters P, Q, R with subscripts or superscripts will range over nonzero polynomials with one indeterminate, positive coefficients, and degree ≥ 1, and the lowercase greek letters ϵ, δ with subscripts or superscripts will range over positive real numbers.

A **decision function** is any family $d = \{d_n : n \in N_k, k \in I\}$ of functions $d_n : (Z_n^*)^{P(k)} \longrightarrow \{0, 1\}$, where P is a polynomial. For any decision function d as above, any sequence $\ell(n)$ of $P(k)$ bits, and any $n \in N_k$ let

$$P_{d,n}(\ell(n)) = Pr[d_n(x) = 1 | x \in \Omega_{P(k),n,\ell(n)}]$$

For technical reasons, to become apparent below, the definition of advantage for determining quadratic residuosity will be extended to the definition of advantage for determining quadratic residuosity assuming a quadratic nonresidue is known.

Definition 4.1 *A polynomial size circuit $C = \{C_k : k \geq 1\}$ has a $(1/2-1/P)$-advantage for determining quadratic residuosity for the family N, assuming a quadratic nonresidue is known, and this will be abbreviated by $AQR^+(C, N, 1 - 1/P)$, if for all but a finite number of indices $k \in I$ the following property holds for all $n \in N_k$,*

$$\frac{1}{2}Pr[C_k(n,x,y) = 1 \mid x \in QR_n \text{ and } y \notin QR_n] +$$

$$\frac{1}{2}Pr[C_k(n,x,y) = 0 \mid x \notin QR_n \text{ and } y \notin QR_n] \geq 1 - \frac{1}{P(k)},$$

where for each $n \in N_k$, x, y range over $Z_n^*(+1)$.

The following two lemmas will be used in the study of QRS.

Lemma 4.3 (**Goldwasser-Micali**)

$$(\exists C)(\exists P)AQR^+(C, N, 1 - 1/P) \Rightarrow (\forall Q)(\exists C)AQR(C, N, 1 - 1/Q)$$

Proof: (Outline) Let $C = \{C_k : k \in I\}$ be a polynomial size circuit and P a polynomial as in definition 4.1. Let $n \in N_k$ and let Q be any given fixed polynomial. Put $m = 4 \cdot Q(k) \cdot P(k)^2$ and select at random m quadratic residues

$$s_1^2 \text{ mod} n, \ldots, s_m^2 \text{ mod} n.$$

Further, select at random m elements

$$y_1, \ldots, y_m \in Z_n^*(+1),$$

and put $Y = \{y_1, \ldots, y_m\}$. The idea is now the following: one of the elements of Y is a quadratic nonreridue with high probability (in fact the probability is $1 - 2^{-m}$); it is natural to search for such an element, say $z \in Y$, and then use the circuit C' defined by

$$C'_k(n,x) = C_k(n,x,z), \text{ where } x \in Z_n^*(+1). \tag{15}$$

To search for such an element $z \in Y$ it is enough to check the performance of the circuit C.

Thus, for $t = 1, \ldots, m$ do:

Step 1: Compute the integers

$$R_{n,t} = |\{1 \leq i \leq m : C_k(n, s_i^2 \text{ mod} n, y_t) = 1\}|$$

$$R_{n,t,j} = |\{1 \leq i \leq m \ : \ C_k(n, y_j \cdot s_i^2 \ \mathrm{mod} n, y_t) = 1\}|$$

Step 2: Compute

$$d_{n,t,j} = |R_{n,t} - R_{n,t,j}|$$

until for some $j = 1, \ldots, m$, the following holds

$$d_{n,t,j} > \frac{1}{P(k)} \tag{16}$$

If a j can be found such that (16) is true then define circuit C' as in (15) with $z = y_t$. The rest of the proof is an application of the weak law of large numbers and will be left as an exercise (see exercise 1) •

Lemma 4.4 (Goldwasser-Micali) *Let P, R be polynomials and let d be an easy to compute decision function such that the following statement holds for all but a finite number of $k \in I$: for all $n \in N_k$ one can efficiently compute $u, u' \in \{0,1\}^{P(k)}$ such that*

$$|P_{d,n}(u) - P_{d,n}(u')| > \frac{1}{R(k)} \tag{17}$$

Then one can prove that

$$(\forall Q)(\exists C) AQR^+(C, N, 1 - 1/Q).$$

Proof: Without loss of generality it can be assumed that u, u' in (17) are adjacent. To see this let $n \in N_k$ and let u, u' witness the validity of (17). If $\Delta = \mathrm{dis}(u, u')$ then there exists a sequence $u_0 = u, u_1, \ldots, u_\Delta = u' \in \{0,1\}^{P(k)}$ such that for all $1 \leq i \leq \Delta$, u_{i-1}, u_i are adjacent. It follows that

$$\sum_{i=1}^{\Delta} |P_{d,n}(u_{i-1}) - P_{d,n}(u_i)| \geq |P_{d,n}(u) - P_{d,n}(u')| > \frac{1}{R(k)}.$$

Hence, there exists an $1 \leq i \leq \Delta$ such that

$$|P_{d,n}(u_{i-1}) - P_{d,n}(u_i)| > \frac{1}{\Delta \cdot R(k)} \geq \frac{1}{R'(k)},$$

where $R'(k) = R(k) \cdot P(k)$ (here one uses the fact that $\Delta \leq P(k)$.)

Next, for any polynomial Q define a circuit C as follows:

Input: $n \in N_k, x \in Z_n^*(+1), y \in Z_n^*(+1) - QR_n$.

Step 1: Put $m = 4 \cdot Q(k) \cdot R(k)^2$.

Step 2: Choose at random m quadratic residues

$$s_1^2 \ \mathrm{mod} n, \ldots, s_m^2 \ \mathrm{mod} n \in QR_n$$

and put

$$y_1 \equiv x \cdot s_1^2 \ \mathrm{mod} n, \ldots, y_m \equiv x \cdot s_m^2 \ \mathrm{mod} n.$$

Step 3: Compute $u = (u_1, \ldots, u_{P(k)}), u' = (u'_1, \ldots, u'_{P(k)})$ adjacent, witnessing the validity of (17); let r be such that $u_r \neq u'_r$ (without loss of generality assume $u_r = 1, u'_r = 0$.)

Step 4: Choose at random m elements $w_1, \ldots, w_m \in \Omega_{P(k),n,u}$ and m elements $w'_1, \ldots, w'_m \in \Omega_{P(k),n,u'}$.

Step 5: **For** $i = 1, \ldots, r-1, r+1, \ldots, P(k)$ **do**

For $j = 1, \ldots, m$ draw $z_j \in Z_n^*(+1)$ at random and put

$$y_{j,i} \equiv y^{u_i \oplus 1} \cdot z^2 \bmod n.$$

Moreover, put

$$y_{j,r} = y_j, \quad \text{for } j = 1, \ldots, m.$$

Step 6: For each $j = 1, \ldots, m$ put

$$x_j = (y_{j,1}, \ldots, y_{j,r-1}, y_j, y_{j,r+1}, \ldots, y_{j,P(k)}).$$

Step 7: Compute

$$d_{x,y} = \left| \frac{d(x_1) + \cdots + d(x_m)}{m} - \frac{d(w_1) + \cdots + d(w_m)}{m} \right|,$$

$$d'_{x,y} = \left| \frac{d(x_1) + \cdots + d(x_m)}{m} - \frac{d(w'_1) + \cdots + d(w'_m)}{m} \right|.$$

Output:

$$C_k(n, x, y) = \begin{cases} 1 & \text{if } d_{x,y} < \frac{1}{2R(k)} \\ 0 & \text{if } d'_{x,y} < \frac{1}{2R(k)} \end{cases}$$

To show that this circuit works notice that for all $j = 1, \ldots, m$

$$\sigma_n(y_{i,j}) = \begin{cases} u_i & \text{if } i \neq r \\ \sigma_n(y_j) & \text{if } i = r \end{cases}$$

However, by definition either all the y_j are quadratic residues or else they are all quadratic nonresidues. It follows that either $\{x_1, \ldots, x_m\} \subseteq \Omega_{P(k),n,u}$ or $\{x_1, \ldots, x_m\} \subseteq \Omega_{P(k),n,u'}$ depending on whether x is a quadratic residue or not. The rest of the proof is an application of the weak law of large numbers. Indeed, let $A_n = \{(x,y) : x \in QR_n, y \notin QR_n\}$ and $B_n = \{(x,y) : x \notin QR_n, y \notin QR_n\}$.

$$Pr_{A_n}\left[\left|\frac{d(x_1) + \cdots + d(x_m)}{m} - P_{d,P(k),u}\right| \geq \frac{1}{2R(k)}\right] \leq \frac{1}{Q(k)},$$

$$Pr_{B_n}\left[\left|\frac{d(x_1) + \cdots + d(x_m)}{m} - P_{d,P(k),u'}\right| \geq \frac{1}{2R(k)}\right] \leq \frac{1}{Q(k)}.$$

Hence the result follows from the above inequalities as well as

$$Pr\left[\left|\frac{d(w_1)+\cdots+d(w_m)}{m} - P_{d,P(k),u}\right| \geq \frac{1}{2R(k)}\right] \leq \frac{1}{Q(k)},$$

$$Pr\left[\left|\frac{d(w_1')+\cdots+d(w_m')}{m} - P_{d,P(k),u'}\right| \geq \frac{1}{2R(k)}\right] \leq \frac{1}{Q(k)}\bullet$$

Let P be a fixed polynomial, $n \in N_k$. For each integer k let $\Theta_{P(k)}$ be the set of all messages of length $P(k)$. For any message $M \in \Theta_{P(k)}$ let $M^{(e)}$ be the set of all encodings of M i.e.

$$M^{(e)} = \{E_n(x;M) : x = (x_1,\ldots,x_{P(k)}) \in (Z_n^*)^{P(k)}\}.$$

It is easy to see that for any two messages $M, \overline{M} \in \Theta_{P(k)}$, $|M^{(e)}| = |\overline{M}^{(e)}|$ (see exercise 2.)

A predicate S on the family of messages $\{\Theta_{P(k)} : k \in I\}$ is a family $S = \{S_k : k \in I\}$ such that for all $k \in I$, $S_k : \Theta_{P(k)} \longrightarrow \{0,1\}$.

Theorem 4.5 *(Goldwasser-Micali) Let P, Q be polynomials, S an easy to evaluate predicate on the family $\{\Theta_{P(k)} : k \in I\}$ and $C = \{C_k : k \in I\}$ a polynomial size circuit such that C_k has $P(k)$ input gates and one output gate. Further, assume that for all but a finite number of $k \in I$, and all $n \in N_k$,*

$$Pr[C_k(n, E_n(x;M)) = S_k(M)] \geq Pr[S_k(M) = 1] + \frac{1}{Q(k)}. \tag{18}$$

Then there exists a polynomial R and an easy to compute decision function $d = \{d_n : n \in N_k, k \in I\}$, where $d_n : (Z_n^*)^{P(k)} \longrightarrow \{0,1\}$, such that for all but a finite number of $k \in I$, and for all $n \in N_k$ one can efficiently compute $u, u' \in \{0,1\}^{P(k)}$ such that

$$|P_{d,n}(u) - P_{d,n}(u')| > \frac{1}{R(k)} \tag{19}$$

Proof: Let $k \in I, n \in N_k$ be fixed and let χ be the common value of $|M^{(e)}|$, where $M \in \Theta_{P(k)}$. Further, put $\Theta = \Theta_{P(k)}$ and $\theta = |\Theta$. For any $M \in \Theta$, and any $i \in \{0,1\}$ let $G^i(M)$ be the number of encodings e of M such that $C_k(n,e) = i$ i.e.

$$G^i(M) = |\{E_n(x;M) : C_k(n, E_n(x;M)) = i\}|$$

Finally let

$$G(M) = \begin{cases} G^1(M) & \text{if } S_k(M) = 1 \\ G^0(M) & \text{if } S_k(M) = 0 \end{cases}$$

It is then clear that

$$Pr[C_k(n, E_n(x : M)) = S_k(M)] = \frac{1}{\chi \cdot \theta} \sum_{M \in \Theta} G(M). \tag{20}$$

Partition the set Θ into $R(k) = 10 \cdot Q(k)$ sets $\{\Theta(t) : t = 1, \ldots, R(k)\}$ defined by

$$M \in \Theta(t) \Leftrightarrow \frac{t-1}{R(k)} \leq \frac{G^1(M)}{\chi} < \frac{t}{R(k)}. \tag{21}$$

Since,

$$\theta = \sum_{t=1}^{R(k)} |\Theta(t)|, \tag{22}$$

it follows that there exists $1 \leq t \leq R(k)$ such that

$$|\Theta(t)| > \frac{\theta}{R(k)^2}.$$

The main part of the proof of the theorem consists of proving the following

Claim: There exist $1 < s + 1 < t \leq R(k)$ such that

$$|\Theta(s)|, |\Theta(t)| > \frac{\theta}{R(k)^2}. \tag{23}$$

Proof of the Claim: Assume that there are no $1 < s + 1 < t \leq R(k)$ such that (23) holds. Then one of the following two cases can occur:

1. There exists exactly one t such that (23) holds for $|\Theta(t)|$

2. There exists exactly one t such that (23) holds for $|\Theta(t-1)|$ and $|\Theta(t)|$.

Put $p_k = Pr[S_k(M) = 1]$. In case 1, $\sum_{M \in \Theta(i)} G(M)$ is maximum when $i = R(k)$ and $(\forall M)(S_k(M) = 1 \Rightarrow M \in \Theta(R(k)))$; thus, using (18) one can show that

$$\begin{aligned} p_k + \frac{1}{Q(k)} &\leq \frac{1}{\chi \cdot \theta} \left[\sum_{M \in \Theta(R(k))} G(M) + \sum_{M \in \Theta(i), i < R(k)} G(M) \right] \\ &\leq \frac{1}{\chi \cdot \theta} \left[\theta \chi p_k + \frac{\theta}{R(k)^2} R(k) \chi \right] = p_k + \frac{1}{R(k)}, \end{aligned}$$

which is a contradiction. In case 2, $\sum_{M \in \Theta(i-1)} G(M) + \sum_{M \in \Theta(i)} G(M)$ is maximum when $i = R(k)$ and $(\forall M)(S_k(M) = 1 \Rightarrow M \in \Theta(R(k)))$ and $(\forall M)(S_k(M) = 0 \Rightarrow M \in \Theta(R(k) - 1))$; thus, using (18) one can show that

$$p_k + \frac{1}{Q(k)} \leq$$

$$\frac{1}{\chi \cdot \theta}\left[\sum_{M\in\Theta(R(k)-1)} G(M) + \sum_{M\in\Theta(R(k))} G(M) + \sum_{M\in\Theta(i), i<R(k)-1} G(M)\right]$$
$$\leq \frac{1}{\chi \cdot \theta}\left[\theta\chi p_k + 2(1-p_k)\frac{\theta\chi}{R(k)} + \frac{\theta\chi}{R(k)}\right] < p_k + \frac{1}{2Q(k)},$$

which is a contradiction. This completes the proof of the claim.

To define the decision function d, for each $k \in I, n \in N_k$ let

$$d_n(x) = C_k(n, x), \text{ where } x \in \{0,1\}^{P(k)}.$$

Let $1 < s+1 < t \leq R(k)$ be such that (23) holds. Then it is clear that for all $u \in \Theta(s), u' \in \Theta(t)$,

$$\left|\frac{G^1(u)}{\chi} - \frac{G^1(u')}{\chi}\right| > \frac{1}{R(k)}. \tag{24}$$

Using a Monte Carlo computation one can easily compute $u \in \Theta(s), u' \in \Theta(t)$. However,

$$P_{d,n}(u) = \frac{G^1(u)}{\chi}, P_{d,n}(u') = \frac{G^1(u')}{\chi}.$$

Thus, the theorem follows from the above equations and inequality (24) •

The following interpretation of the hypothesis of theorem 4.5 will be useful. Let $S = \{S_k : k \in I\}$ be an easy to evaluate predicate on the family $\{\Theta_{P(k)} : k \in I\}$ of sets of messages. Call S_k true of the message M, where $M \in \Theta_{P(k)}$, if $S_k(M) = 1$, and false otherwise. Then $Pr[S_k(M) = 1]$ is the probability that $S_k(M)$ is true on a random message $M \in \Theta_{P(k)}$. Let $C = \{C_k : k \in I\}$ be a polynomial size circuit such that C_k has $P(k)$ input gates and one output gate. Then the quantity

$$Pr[C_k(n, E_n(x; M)) = S_k(M)],$$

is the probability that the polynomial size circuit C guesses correctly the value of $S_k(M)$ assuming only knowledge of the encoded message $E_n(x; M)$. The hypothesis of the theorem now states:

There exists a polynomial Q, an easy to compute predicate S, and a polynomial size circuit C such that for all but a finite number of k, C_k guesses the correct value of $S_k(M)$ from a random encoding $E_n(x; M)$ of M with a $1/Q(k)$ advantage.

If one recalls that $QRA(N)$ is an abbreviation of the Quadratic Residuosity Assumption for the family N, takes into account the results in the subsection on the Quadradic Residue Generator, and combines them with lemmas 4.3, 4.4 and theorem 4.5, it is immediate that

Theorem 4.6 *Assuming $QRA(N)$ there is no polynomial Q, no easy to compute predicate S, and no polynomial size circuit C such that for all but a finite number of k, C_k guesses the correct value of $S_k(M)$ from a random encoding $E_n(x; M)$ of M with a $1/Q(k)$ advantage* •

EXERCISES

1: Complete the details of the proof of lemma 4.3.

2: Show that for any two messages $M, \overline{M} \in \Theta_{P(k)}$, $|M^{(e)}| = |\overline{M}^{(e)}|$ **Hint:** Show that the mapping $E_n(x; M) \longrightarrow E_n(x; \overline{M})$ is one to one and onto.

4.10 BIBLIOGRAPHICAL REMARKS

The recent rapid development of public key cryptosystems followed immediately after the publication of Diffie and Hellman in [DH]. Before this the security of cryptosystems was based on absolute security criteria (see [Shan1] and [Kon]). For further general remarks on cryptosystems the reader should consult [Pe], [Lem], [Bet]. Some recent works which include material on public key cryptosystems are [Kon], [D] and [DDDHL].

The *RSA* system described in subsection 4.3 was developed in [RSA] and the Rabin system described in subsection 4.5 in [Rab]. The security of *RSA* bits and Rabin bits studied in subsections 4.4 and 4.6 respectively is from [GMT].

The Merkle Hellman system is based on knapsacks and was developed in [MH]. The presentation of the security of the Merkle Hellman system presented in subsection 4.8 is based partly on [Sham1] and [EL].

The presentation of the Quadratic Residue System presented in subsection 4.9 is a continuation of the presentation of the Quadratic Residue generator and is from [GM].

5 TOWARDS A GENERAL THEORY

5.1 INTRODUCTION

The present section presents a general theory of pseudo-random generators and public key cryptosystems. Subsection 5.2 includes two security tests for pseudo-random generators. The first one, the Blum-Micali Test, is used to construct unpredictable pseudo-random generators. The second one, Yao's Statistical Test, is proved in theorem 5.2 to be equivalent to the Blum-Micali Test. This is important, because it confirms the fact that the security of the pseudo-random generators constructed via the Blum-Micali Generator theorem do not depend on the order of the bits produced by the generator.

The second subsection deals with the concept of xoring. This is very important for the construction of pseudo-random generators satisfying improved unpredictability properties. The next subsection gives a complete proof of the XOR lemma. The proof is divided into two parts. The first part gives a heuristic proof of the lemma, which will be essential to understanding the main formal proof, which follows next.

The last two subsections deal with three applications of the XOR theorem: to unapproximable predicates, to pseudo-random generators and to $1-1$, one way functions.

5.2 SECURITY TESTS

Let $S_m = \{0,1\}^m$ be the set of sequences of bits of length exactly m. Let $X = \{X_m : m \geq 0\}$ denote a family of nonempty sets such that each X_m is a subset of S_m, let $f = \{f_m : m \geq 0\}$ be a family of polynomial time computable functions such that each f_m is a permutation of X_m and let $B = \{B_m : m \geq 0\}$ be a family of polynomial time computable functions such that each $B_m : X_m \longrightarrow \{0,1\}$ is a $0,1$-valued function with domain X_m. Any such family $\{B_m : m \geq 0\}$ of functions is called a predicate on $\{X_m : m \geq 0\}$.

As usual, the capital roman letters P, Q with or without subscripts and superscripts will range over polynomials of degree ≥ 1 with positive coefficients. All the circuits considered in the present section will be probabilistic.

Further, it will be very important for the construction of polynomial size circuits to be able to generate random elements in X_m. To be more exact from now on and for the rest of this section whenever a family $X = \{X_m : m \geq 0\}$ is considered it will be assumed that there exists an algorithm running in time polynomial in m which on input m it will output a random element of X_m.

Definition 5.1 *A polynomial size circuit $C = \{C_m : m \geq 0\}$ P-predicts the predicate $B = \{B_m : m \geq 0\}$ if the following statement holds for infinitely many*

m,

$$Pr[x \in X_m : B_m(x) = C_m(x)] \geq \frac{1}{2} + \frac{1}{P(m)}$$

Definition 5.2 *The predicate $B = \{B_m : m \geq 0\}$ is unapproximable if*

$$(\forall P, C)(C \text{ does not } P - \text{predict } B).$$

Definition 5.3 *The family $f = \{f_m : m \geq 0\}$ of functions is a friendship function for the predicate $B = \{B_m : m \geq 0\}$ if both of the following two functions are computable in time polynomial in m,*

1. $< m, x > \longrightarrow f_m(x)$
2. $< m, x > \longrightarrow B_m(f_m(x))$

Remark: Notice that the function $< m, x > \longrightarrow B_m(x)$ in definition 5.3 need not be computable in time polynomial in m.

Example 5.1 *For any two primes p, q satisfying $p \equiv q \equiv 3 \bmod 4$ consider the following function and predicate:*

1. $f_n : QR_n \longrightarrow QR_n : x \longrightarrow x^2 \bmod n$
2. $B_n : QR_n \longrightarrow \{0, 1\} : x \longrightarrow B_n(x) = \text{par}(\sqrt{x} \bmod n)$.

Assuming the Quadratic Residuosity Assumption it is easy to see that the above family satisfies the requirements of definition 5.3.

Example 5.2 *For any prime p and any generator $g \in Z_p^*$ consider the following function and predicate:*

1. $f_p : Z_p^* \longrightarrow Z_p^* : x \longrightarrow g^x \bmod p$
2. $B_p : Z_p^* \longrightarrow \{0, 1\} : x \longrightarrow B_p(x)$,

where

$$B_p(x) = \begin{cases} 1 & \text{if } x = PQR(p, g, x^2 \bmod p) \\ 0 & \text{if } x = NPQR(p, g, x^2 \bmod p), \end{cases}$$

and $PQR(p, g, x^2 \bmod p)$, $NPQR(p, g, x^2 \bmod p)$ respectively denote the principal, nonprincipal square root of $x^2 \bmod p$. Assuming the Discrete Logarithm Assumption it is easy to see that the above family satisfies the requirements of definition 5.3.

Definition 5.4 *A family $G = \{G_m : m \geq 0\}$ of functions is a pseudo-random generator, if there exists a polynomial Q such that*

1. *For all m, $G_m : X_m \longrightarrow S_{Q(m)}$ and*
2. *$< m, x > \longrightarrow G_m(x)$ is computable in time polynomial in m.*

To any pseudo-random generator G as in definition 5.4 associate the sequence $b^G_{m,0}(x), \ldots, b^G_{m,Q(m)-1}(x)$ of bits generated by G, where for each index m, $b^G_{m,i}(x)$ is the i-th bit generated by G_m on input x.

Definition 5.5 *A polynomial size circuit $C = \{C_m : m \geq 0\}$ P-predicts the pseudo-random generator $G = \{G_m : m \geq 0\}$, if for infinitely many m, there exists an $i < Q(m)$ such that*

$$Pr\left[G_m(x) : C_m(b^G_{m,0}(x), \ldots, b^G_{m,i-1}(x)) = b_{m,i}(x)\right] \geq \frac{1}{2} + \frac{1}{P(m)}$$

Definition 5.6 *A pseudo-random generator $G = \{G_m : m \geq 0\}$ passess the Blum-Micali test, and the test will be abbreviated BMT, if the following statement holds,*

$$(\forall C, P)(C \text{ does not } P - \text{predict } G).$$

For any function $h : Y \longrightarrow Y$ and any integer $n \geq 0$ recall that $h^n : Y \longrightarrow Y$ stands for the function defined by induction as follows:

$$h^i(x) = \begin{cases} h(x) & \text{if } i = 1 \\ h(h^{i-1}(x)) & \text{if } i > 1 \end{cases}$$

The following theorem is very important, because it provides a technique for constructing pseudo-random generators that pass the Blum-Micali test from an unapproximable predicate $B = \{B_m : m \geq 0\}$, and a friendship function $f = \{f_m : m \geq 0\}$ for B.

Theorem 5.1 (**The Blum-Micali Generator Theorem**) *For any polynomial Q, any unapproximable predicate $B = \{B_m : m \geq 0\}$, and any friendship function $f = \{f_m : m \geq 0\}$ for B the pseudo-random generator $G^{B,f,Q} = \{G^{B,f,Q}_m : m \geq 0\}$ defined for $x \in X_m$ by*

$$G^{B,f,Q}_m(x) = < B_m(f_m^{Q(m)}(x)), \ldots, B_m(f_m^{Q(m)-j}(x)), \ldots, B_m(f_m(x)) >,$$

passes the BMT.

Proof: Consider the abbreviation

$$b^G_{m,j}(x) = B_m(f_m^{Q(m)-j}(x)), \text{ for } 0 \leq j < Q(m),$$

and assume on the contrary that the pseudo-random generator $G^{B,f,Q}$ does not pass the Blum-Micali test. It follows that there exists a polynomial size circuit $C = \{C_m : m \geq 0\}$ and a polynomial P such that C, P-predicts the generator $G^{B,f,Q}$. It follows from the definition of $G^{B,f,Q}$ that for infinitely many m, there exists an $i < Q(m)$ such that

$$Pr\left[G_m(x) : C_m(b^G_{m,0}(x), \ldots, b^G_{m,i-1}(x)) = b^G_{m,i}(x)\right] \geq \frac{1}{2} + \frac{1}{P(m)}. \qquad (1)$$

Let M be the set of indices m which satisfy inequality (1). Clearly, for each $m \in M$ there exists an integer $i_m < Q(m)$ such that

$$Pr\left[G_m(x) : C_m(b^G_{m,0}(x), \ldots, b^G_{m,i_m-1}(x)) = b^G_{m,i_m}(x)\right] \geq \frac{1}{2} + \frac{1}{P(m)}, \qquad (2)$$

Define a new circuit $C' = \{C'_m : m \geq 0\}$ as follows for $x \in X_m$,

$$C'_m(x) = C_m(B_m(f_m^{i_m}(x)), B_m(f_m^{i_m-1}(x)), \ldots, B_m(f_m(x))).$$

One can then prove that the following claim holds:

Claim: For all $m \in M$,

$$Pr\left[G_m(x) : C'_m(x) = B_m(x)\right] \geq \frac{1}{2} + \frac{1}{P(m)}$$

Proof of Claim: Fix an arbitrary $m \in M$ and put $i = i_m$, $j = i - Q(m)$, $x' = f^j(x)$, where x ranges over X_m. Then the following statements are equivalent for each $x \in X_m$,

$$C'_m(x) = B_m(x)$$

$$C_m(B_m(f_m^i(x)), B_m(f_m^{i-1}(x)), \ldots, B_m(f_m(x))) = B_m(x).$$

$$C_m(b^G_{m,0}(f_m^j(x)), b^G_{m,1}(f_m^j(x)), \ldots, b^G_{m,i-1}(f_m^j(x))) = b^G_{m,i}(f_m^j(x)).$$

$$C_m(b^G_{m,0}(x'), b^G_{m,1}(x'), \ldots, b^G_{m,i-1}(x')) = b^G_{m,i}(x').$$

However, the mapping $x \longrightarrow x'$ is a permutation of X_m. Hence, using inequality (2) one obtains that,

$$Pr\left[G_m(x') : C_m(b^G_{m,0}(x'), \ldots, b^G_{m,i-1}(x')) = b^G_{m,i}(x')\right] \geq \frac{1}{2} + \frac{1}{P(m)},$$

which completes the proof of the claim.

But, this is a contradiction since the predicate B is unapproximable. The proof of the theorem is now complete.•

Definition 5.7 A polynomial size statistical test, *abbreviated PSST, for the pseudo-random generator $G = \{G_m : m \geq 0\}$, where $G_m : X_m \longrightarrow S_{Q(m)}$ for some polynomial Q, is a polynomial size $0,1$-valued circuit $C = \{C_m : m \geq 0\}$ which for each $m \geq 0$ has $Q(m)$ input gates.*

Definition 5.8 *Let C be a PSST for the generator G. For each $m \geq 0$, consider the probabilities*

$$p_m^{C,G} = Pr[G_m(x) : C_m(G_m(x)) = 1],$$
$$p_m^{C,R} = Pr[u \in S_{Q(m)} : C_m(u) = 1].$$

Definition 5.9 *The pseudo-random generator G passes the PSST C, if for all but a finite number of integers m,*

$$(\forall P)\left[\mid p_m^{C,G} - p_m^{C,R} \mid < \frac{1}{P(m)}\right].$$

Definition 5.10 *The pseudo-random generator G passes Yao's statistical test, abbreviated YST, if for any PSST C for G, G passes C.*

Theorem 5.2 (**Yao's PSST Theorem, A. Yao**) *For any pseudo-random generator $G = \{G_m : m \geq 0\}$, the following statements are equivalent:*

1. *G passes the Blum-Micali Test.*
2. *G passes the Yao Statistical Test.*

Proof: Assume that for each $m \geq 0$, $G_m : X_m \longrightarrow S_{Q(m)}$, where Q is a polynomial.

(2) $\Rightarrow$ (1)

Assume, by way of contradiction, that (2) is true, but (1) fails. Let P be a polynomial, $C = \{C_m : m \geq 0\}$ a polynomial size circuit and M the set of integers m such that there exists an $i < Q(m)$ so that the following holds:

$$Pr\left[G_m(x) : C_m(b_{m,0}^G(x), \ldots, b_{m,i-1}^G(x)) = b_{m,i}^G(x)\right] \geq \frac{1}{2} + \frac{1}{P(m)}. \quad (3)$$

For each $m \in M$ let i_m be an integer $i < Q(m)$ that satisfies inequality (3). Define a new polynomial size circuit $C' = \{C'_m : m \geq 0\}$, which for any given $u =< u_0, \ldots, u_{Q(m)-1} >\in S_{Q(m)}$ is given by the formula

$$C'_m(u) = C_m(u_0, \ldots, u_{i_m - 1}) \oplus u_{i_m} \oplus 1.$$

It is then clear that for all $u \in S_{Q(m)}$,

$$C'_m(u) = 1 \Leftrightarrow C_m(u_0, \ldots, u_{i_m-1}) = u_{i_m}.$$

It is an immediate consequence of definition 5.8 that for all $m \in M$,

$$p_m^{C',G} = Pr[C_m(b_{m,0}^G(x), \ldots, b_{m,i_m-1}^G(x)) = b_{m,i_m}^G(x)] \geq \frac{1}{2} + \frac{1}{P(m)}. \quad (4)$$

Now it can be proved that

Claim 1: $(\forall m \in M)\left[p_m^{C',G} \geq 1/2 + 1/(2P(m))\right]$

Proof of Claim 1: Sinse G passes YST it must also pass the test C'. Thus, the following inequality holds for all but a finite number of m,

$$\left[\left| p_m^{C',G} - p_m^{C',R} \right| < \frac{1}{2P(m)}\right].$$

It follows from (4) that for all but a finite number of $m \in M$,

$$p_m^{C',R} > p_m^{C',G} - \frac{1}{2P(m)} \geq \frac{1}{2} + \frac{1}{P(m)} - \frac{1}{2P(m)} = \frac{1}{2} + \frac{1}{2P(m)},$$

which completes the proof of claim 1.

Now a contradiction can be derived easily. Indeed,

$$p_m^{C',R} = Pr[u \in S_{Q(m)} : C_m(u_0, \ldots, u_{i_m-1}) = u_{i_m}] =$$
$$Pr[C_m(u_0, \ldots, u_{i_m-1}) = 0 | u_{i_m} = 0] \cdot Pr[u_{i_m} = 0] +$$
$$Pr[C_m(u_0, \ldots, u_{i_m-1}) = 1 | u_{i_m} = 1] \cdot Pr[u_{i_m} = 0] =$$
$$\frac{1}{2}(Pr[C_m(u_0, \ldots, u_{i_m-1}) = 0 | u_{i_m} = 0] +$$
$$Pr[C_m(u_0, \ldots, u_{i_m-1}) = 1 | u_{i_m} = 1]) =$$
$$\frac{1}{2}(Pr[C_m(u_0, \ldots, u_{i_m-1}) = 0] + Pr[C_m(u_0, \ldots, u_{i_m-1}) = 1]) = \frac{1}{2},$$

a contradiction.

(1) ⇒ (2)

Given two sequences $u =< u_1, \ldots, u_m >$, $v =< v_1, \ldots, v_n >$, of bits, where $m, n \geq 0$, let the **concatenation**, of u, v, abbreviated $u \frown v$, denote the sequence $< u_1, \ldots, u_m, v_1, \ldots, v_n >$. The number m is called the **length** of u, and is abbreviated $\ell(u)$. The **inverse** sequence $< u_m, \ldots, u_1 >$ obtained from u by reversing the order of the bits in the representation of u is denoted by $*u$. Assume, by way of contradiction, that G does not pass the YST. Let $C = \{C_m : m \geq 0\}$ be a polynomial size circuit and P a polynomial such that the following inequality holds for infinitely many m,

$$| p_m^{C,G} - p_m^{C,R} | \geq \frac{1}{P(m)}. \tag{5}$$

Let M be the set of integers that satisfy (5). For each $i \leq Q(m)$, define the following subsets of $S_{Q(m)}$,

$$S_m^i = \{t \frown < b_{m,i-1}^G(x), \ldots, b_{m,0}^G(x) >: x \in X_m, \ell(t) = Q(m) - i\} \tag{6}$$

It is then apparent from definition (6) that

$$S_m^{Q(m)} = \{G_m(x) : x \in X_m\} \subseteq S_m^{Q(m)-1} \subseteq \ldots \subseteq S_m^1 \subseteq S_m^0 = S_{Q(m)}.$$

Also, for each $i \leq Q(m)$ consider the following probabilities:

$$p_m^i = Pr\left[u \in S_m^i : C_m(u) = 1\right].$$

It is then clear from definition 5.8 that

$$p_m^0 = p_m^{C,R} \text{ and } p_m^{Q(m)} = p_m^{C,G}.$$

Moreover, using (5), for all $m \in M$ one has

$$\frac{1}{P(m)} \leq | p_m^{C,G} - p_m^{C,R} | = |p_m^0 - p_m^{Q(m)}| \leq \sum_{i=0}^{Q(m)-1} | p_m^i - p_m^{i+1} |.$$

Hence, for each $m \in M$ there exists an $i < Q(m)$, call it i_m, such that

$$|p_m^{i_m} - p_m^{i_m+1}| \geq \frac{1}{Q'(m)}, \tag{7}$$

where $Q'(m) = P(m) \cdot Q(m)$.

The purpose of what follows is to construct a polynomial size circuit C'_m which will predict the generator G. The idea of the construction of the circuit C' is the following:

Input: $u_0, \ldots, u_{i_m-1}$

Step 1: Choose a random $v =< v_{Q(m)-i_m-1}, \ldots, v_0 >$ of length $\ell(v) = Q(m) - i_m$.

Output:

$$C'_m(u_0, \ldots, u_{i_m-1}) = \begin{cases} v_0 & \text{if } C_m(v \frown < u_0, \ldots, u_{i_m-1} >) = 1 \\ 1 \oplus v_0 & \text{if } C_m(v \frown < u_0, \ldots, u_{i_m-1} >) = 0 \end{cases}$$

It is not hard to see that

$$Pr[C'_m(b^G_{m,i_m-1}(x), \ldots, b^G_{m,0}(x)) = b^G_{m,i_m}(x)] \geq \frac{1}{2} + \frac{1}{Q'(m)}, \tag{8}$$

The only problem in the above circuit is to show how to choose random samples of big enough size of such v's so that the Weak Law of Large Numbers can be applied. For example, if $i_m = Q(m) - 1$ this is not possible. The proof that follows is intended to clarify this situation. To illustrate the last part of the proof it will be assumed that

Special Case: $i_m = Q(m) - 1$, for infinitely many $m \in M$.

The general case will be taken up later. The proof in the above special case runs as follows. Let $M' = \{m \in M : i_m = Q(m) - 1\}$. Let A_m (respectively A'_m, A''_m) denote the events:

$$C_m(1 \oplus u_{i_m}(x), u_{i_m-1}(x), \ldots, u_0(x)) \neq C_m(u_{i_m}(x), u_{i_m-1}(x), \ldots, u_0(x))$$

(respectively

$$C_m(1 \oplus u_{i_m}(x), u_{i_m-1}(x), \ldots, u_0(x)) > C_m(u_{i_m}(x), u_{i_m-1}(x), \ldots, u_0(x)),$$

$$C_m(1 \oplus u_{i_m}(x), u_{i_m-1}(x), \ldots, u_0(x)) < C_m(u_{i_m}(x), u_{i_m-1}(x), \ldots, u_0(x)).$$

It is clear from the above definitions that $A_m = A'_m \cup A''_m$. Inequality (7) implies that for all $m \in M$,

$$Pr[A_m] \geq \frac{1}{Q'(m)}.$$

The above inequality suggests defining for each bit b the following polynomial size circuits $C^b_{1,m}$, $C^b_{2,m}$,

$$C^b_{1,m}(u) = \begin{cases} 1 & \text{if } C_m(0, *u) > C_m(1, *u) \\ 0 & \text{if } C_m(0, *u) < C_m(1, *u) \\ b & \text{if } C_m(0, *u) = C_m(1, *u) \end{cases}$$

and

$$C^b_{2,m}(u) = \begin{cases} 1 & \text{if } C_m(0, *u) < C_m(1, *u) \\ 0 & \text{if } C_m(0, *u) > C_m(1, *u) \\ b & \text{if } C_m(0, *u) = C_m(1, *u) \end{cases}$$

where $u \in S_{Q(m)-1}$. It is clear that there exists a bit b such that the following holds for infinitely many $m \in M'$,

$$Pr[G_m(x) : b^G_{m,i_m}(x) = b | \neg A_m] \geq \frac{1}{2}. \tag{9}$$

Let M'' denote the infinite set of $m \in M'$ which satisfy the above inequality (9). For each $m \in M''$,

$$p'_m = Pr[C^b_{1,m}(b^G_{m,0}(x), \ldots, b^G_{m,i_m-1}(x) = b^G_{m,i_m}(x)] =$$

$$Pr[C^b_{1,m}(b^G_{m,0}(x), \ldots, b^G_{m,i_m-1}(x)) = b^G_{m,i_m}(x) \text{ and } G_m(x) \in A_m] +$$

$$Pr[C^b_m(b^G_{m,0}(x), \ldots, b^G_{m,i_m-1}(x)) = b^G_{m,i_m}(x) \text{ and } G_m(x) \notin A_m]$$

However, the definition of $C^b_{1,m}$ implies that

$$G_m(x) \in A'_m \Rightarrow C^b_{1,m}(b^G_{m,0}(x), \ldots, b^G_{m,i_m-1}(x)) = b^G_{m,i_m}(x),$$

$$G_m(x) \notin A_m \Rightarrow C^b_{1,m}(b^G_{m,0}(x), \ldots, b^G_{m,i_m-1}(x)) = b.$$

Consequently,

$$p'_m \geq Pr[A'_m] +$$
$$Pr[C^b_{1,m}(b^G_{m,0}(x), \ldots, b^G_{m,i_m-1}(x)) = b^G_{m,i_m}(x) | \neg A_m] \cdot (1 - Pr[A_m]).$$

Using (9), this implies that,

$$p'_m \geq Pr[A'_m] + \frac{1}{2}(1 - Pr[A_m]). \tag{10}$$

A repetition of the above proof will show that if

$$p''_m = Pr[C^b_{2,m}(b^G_{m,0}(x), \ldots, b^G_{m,i_m-1}(x) = b^G_{m,i_m}(x)]$$

then

$$p''_m \geq Pr[A''_m] + \frac{1}{2}(1 - Pr[A_m]). \tag{11}$$

Adding both sides of the above two inequalities (10), (11) and using $Pr[A_m] = Pr[A'_m] + Pr[A''_m]$ one obtains that for all $m \in M''$,

$$p'_m + p''_m \geq 1 + Pr[A_m] \geq 1 + \frac{1}{Q'(m)}, \tag{12}$$

and hence

$$\text{either } p'_m \geq \frac{1}{2} + \frac{1}{2Q'(m)} \text{ or } p''_m \geq \frac{1}{2} + \frac{1}{2Q'(m)}.$$

It is now easy, using the last inequality, to define nondeterministically a circuit that $(2Q')$-predicts G. This completes the proof for the above described special case.

The rest of the proof will be devoted to the general case. Let E_m denote the event: there exists $b \in \{0,1\}$ and there exists a sequence v of length $\ell(v) = Q(m) - i_m - 1$ such that

$$(\forall u)(C_m(u \frown < b^G_{m,i_m}(x), b^G_{m,i_m-1}(x), \ldots, b^G_{m,0} >) \neq$$
$$C_m(v \frown < b, b^G_{m,i_m-1}(x), \ldots, b^G_{m,0}(x) >)).$$

For each sequence v of length $\ell(v) = Q(m) - i_m - 1$ let Let $E_{m,v}$ denote the event: there exists $b \in \{0,1\}$ such that

$$C_m(v \frown < b, b^G_{m,i_m-1}(x), \ldots, b^G_{m,0}(x) >) \neq$$
$$C_m(v \frown < b^G_{m,i_m}(x), b^G_{m,i_m-1}(x), \ldots, b^G_{m,0}(x) >).$$

It is clear that

$$E_m \subseteq \bigcup_v E_{m,v},$$

and hence using inequality (7)

$$\sum_v Pr[E_m] \geq Pr\left[\bigcup_v E_{m,v}\right] \geq Pr[E_m] \geq \frac{1}{P(m) \cdot Q(m)}, \tag{13}$$

where in the above two inequalities v ranges over all sequences of bits of length $\ell(v) = Q(m) - i_m - 1$. In the sequel, the following two cases will be considered.

Case 1: There exists a polynomial P' such that such that

$$2^{Q(m)-i_m-1} \leq P'(m), \text{ for infinitely many } m \in M.$$

Let P' be a polynomial such the inequality in Case 1 holds and let M' be the infinite set of all integers $m \in M$ which satisfy the corresponding inequality. Since the sum in (13) ranges over at most $P'(m)$ sequences v, for each $m \in M'$ there exists a sequence $v^{(m)}$ of length $\ell(v^{(m)}) = Q(m) - i_m - 1$ such that

$$Pr[E_{v^{(m)}}] \geq \frac{1}{P'(m)P(m)Q(m)}.$$

Now, the proof of the special case described above, applies to the present case. One need only consider the set $A_m = E_{v^{(m)}}$ and introduce the parameter $v^{(m)}$ in the definition of the circuits $C^b_{i,m}, i = 1, 2$. The details are left as an exercise to the reader (see exercise 1.)

Case 2: For all polynomials P' there exists an integer m_0 depending on P' such that for all $m \in M$,

$$m \geq m_0 \Rightarrow 2^{Q(m)-i_m-1} > P'(m).$$

In this case the Weak Law of Large Numbers will be applied. This is done using the circuit C' defined before inequality (8). The details are left as an exercise to the reader (see exercise 2.) •

EXERCISES

1: Complete the details of the proof of Case 1 of theorem 5.2. **Hint:** With the notation of theorem 5.2, use the idea described in the special case: $i_m = Q(m) - 1$, for infinitely many $m \in M$.

2: Complete the details of the proof of Case 2 of theorem 5.2.

3: Show that the index i_m defined in inequality 7 can be computed in polynomial time via a Monte-Carlo computation. **Hint:** For each $i < Q(m)$ consider a random sample $v_0^i, \ldots, v_k^i$ of sequences v_j^i each of length $\ell(v_0^i) = Q(m) - i$. On input u of length i, compute the integer $R_m^i = |\{j : C_m(v_j^i \frown u) = 1\}|$. Choose i such that $|R_m^i - R_m^{i+1}| \geq 1/(3Q(m)P(m))$.

5.3 XORING

The notion of xor, to be studied below, enables one to construct predicates, pseudo random generators and public key cryptosystems with improved security properties.

Definition 5.11 *Given the functions $B_m^1, \ldots, B_m^k$, where for each index $i = 1, \ldots, k$, $B_m^i : X_m^i \longrightarrow \{0,1\}$, the* **xor** *$B_m = B_m^1 \oplus \cdots \oplus B_m^k$ of the predicates $B_m^1, \ldots, B_m^k$, is defined for each $< x_1, \ldots, x_m > \in X_m^1 \times \cdots \times X_m^k$ by*

$$B_m(x_1, \ldots, x_k) = B_m^1(x_1) \oplus \cdots \oplus B_m^k(x_k).$$

Definition 5.12 *Given the families of predicates $B^i = \{B_m^i : m \geq 0\}$, where $i = 0, \ldots, k$, the* **xor family** *of $B^1, \ldots, B^k$, abbreviated $B = B^1 \oplus \cdots \oplus B^k$, is the following family of predicates $B = \{B_m : m \geq 0\}$, defined for each $m \geq 0$ by*

$$B_m = B_m^1 \oplus \cdots \oplus B_m^k.$$

Definition 5.13 *Given an infinite sequence $B^i = \{B_m^i : m \geq 0\}$, where $i = 1, 2, \ldots$, of families of predicates, and a function g with domain the set of positive integers and range a subset of the set of positive integers, the* **g-xor family** *of $B^1, B^2, \ldots$, abbreviated $B^{(g)} = \{B_m^{(g)} : m \geq 0\}$, is defined by*

$$B_m^{(g)} = B_m^1 \oplus \cdots \oplus B_m^{g(m)}.$$

If in the above definition the function g is costant i.e. $g(m) = k$, for all m, then the g-xor $B^{(g)}$ will also be denoted by $B^{(k)}$.

This and the next subsection will be concerned with answering the following

Question: Given an infinite sequence $B^i = \{B_m^i : m \geq 0\}$, where $i = 1, 2, \ldots$, of families of predicates, and a polynomial size computable function g with domain the set of positive integers and range a subset of the set of positive integers, if a polynomial size circuit approximates the g-xor family $B^{(g)} = \{B_m^{(g)} : m \geq 0\}$, with a certain advantage, do there exist polynomial size circuits approximating each of the predicates $B^1, B^2, \ldots$? Do the advantages of the approximations of the predicates $B^1, B^2, \ldots$ obtained via the approximation for $B^{(g)}$, amplify the original advantage of $B^{(g)}$?

Theorem 5.3 *(**The Projection Theorem**) Let $B^{(g)}$ be the g-xor of the predicates $B^1, B^2, \ldots$, where g is a function with domain the set of positive integers and range a subset of the set of positive integers. Let $C = \{C_m : m \geq 0\}$ be a polynomial size circuit and let $\{\epsilon_m : m \geq 0\}$ be a family of positive real numbers. If*

$$\Pr[x \in X_m^1 \times \cdots \times X_m^{g(m)} : C_m(x) = B_m^{(g)}(x)] \geq \frac{1}{2} + \epsilon_m,$$

then for all $i \geq 1$,

$$Pr[x_i \in X_m^i : C_m^i(x_i) = B_m^i(x_i)] \geq \frac{1}{2} + \epsilon_m.$$

Proof: It will be assumed, without loss of generality, that for all m, $g(m) = 2$. The following picture will be helpful in understanding the proof that follows.

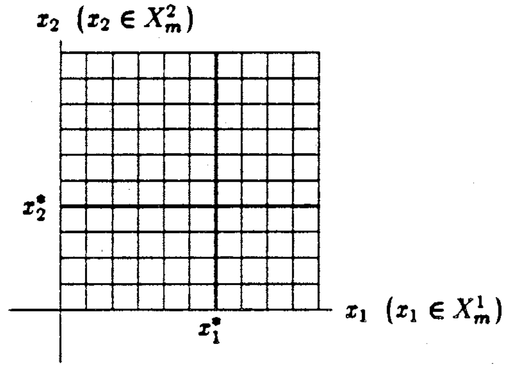


Figure 1: The XOR-predicate

Fix $m \geq 0$. The assumption of the theorem asserts that for at least a ratio $(1/2)+\epsilon_m$ of the points in the $< x_1, x_2 >$-plane, the circuit C_m correctly predicts the value of the xor $B = B^1 \oplus B^2$. Let

$$p(x_1) = Pr[x_2 \in X^2_m : C_m(x_1, x_2) = B^1_m(x_1) \oplus B^2_m(x_2)],$$

and

$$p(x_2) = Pr[x_1 \in X^1_m : C_m(x_1, x_2) = B^1_m(x_1) \oplus B^2_m(x_2)]$$

However, it is true that

$$Pr[< x_1, x_2 >\in X^1_m \times X^2_m : C_m(x_1, x_2) = B_m(x_1, x_2)] =$$

$$\frac{1}{|X^1_m|} \cdot \sum_{x_1 \in X^1_m} p(x_1) = \frac{1}{|X^2_m|} \cdot \sum_{x_2 \in X^2_m} p(x_2) \geq \frac{1}{2} + \epsilon_m.$$

It follows that there exist points $x_1^* \in X^1_m, x_2^* \in X^2_m$, such that

$$p(x_1^*) \geq \frac{1}{2} + \epsilon_m, p(x_2^*) \geq \frac{1}{2} + \epsilon_m.$$

Now, it is easy to see that the following two polynomial size circuits satisfy the requirements of the theorem:

$$C^1_m(x_1) = C_m(x_1, x_2^*) \oplus B^2_m(x_2^*), C^2_m(x_2) = C_m(x_1^*, x_2) \oplus B^1_m(x_1^*). \bullet \qquad (14)$$

Theorem 5.3 will be very useful in the sequel, but what makes an XOR theorem interesting is some amplification of advantage in the passage from an approximation of the xor of two predicates to an approximation of each of the predicates which form the xor. Such a theorem is obtained below.

Theorem 5.4 *Let B be an unapproximable predicate with a friendship function f, on the family $X = \{X_m : m \geq 0\}$, and g a polynomial time computable function from positive integers to positive integers such that for all $m, g(m) \geq 2$. For each polynomial size circuit C and each $u \in X_m$, let*

$$p_u^1[C] = Pr\left[x \in (X_m)^{g(m)} : B_m^{(g)}(x) = C_m(x) | u = 1 - \text{st component of } x\right]$$

If there exists polynomials P, P' and a polynomial size circuit C such that for infinitely many m,

$$\frac{1}{|X_m|}\left|\left\{u \in X_m : p_u^1[C] \geq \frac{1}{2} + \frac{1}{P(m)}\right\}\right| \geq \frac{1}{P'(m)},$$

then for any polynomial Q there exists a polynomial size circuit $C' = \{C'_m : m \geq 0\}$ such that for infinitely many m,

$$Pr[u \in X_m : B_m(u) = C'_m(u)] \geq 1 - \frac{1}{Q(m)}.$$

Proof: Assume that C, P, P' are as in the hypothesis of the theorem and let Q be an arbitrary polynomial. The circuit C' is defined as follows:

Input: u.

Step 1: Put $k = 4P'(m)P(m)Q(m)^2$.

Step 2: For each $1 \leq i \leq k$ let $v_{i,2}, \ldots, v_{i,g(m)-1}$ be a random sample, where each $v_{i,j} \in X_m$.

Step 3: For each $i = 1, \ldots, k$ compute

$$b_i = C'_m(u, f_m(v_{i,2}), \ldots, f_m(v_{i,g(m)-1}) \oplus$$
$$B_m(f_m(v_{i,2})) \oplus \cdots \oplus B_m(f_m(v_{i,g(m)-1})).$$

Step 4: Compute

$$L_1 = |\{1 \leq i \leq k : b_i = 1\}|, L_0 = |\{1 \leq i \leq k : b_i = 0\}|.$$

Output:

$$C'_m(u) = \begin{cases} 1 & \text{if } L_1 > L_0 \\ 0 & \text{if } L_0 > L_1 \end{cases}$$

To show that the above circuit C' works notice that if $B_m(u) = 1$ then the above experiment is expected to output $L_1 = (k/2) + k/P(m)$ many 1's and $L_0 = (k/2) - k/P(m)$ many 0's. Thus $L_1 > L_0$. Similarly, if $B_m(u) = 1$ then $L_1 < L_0$. Now, the theorem follows from the Weak Law of Large Numbers.•

To give the proofs of the XOR theorems stated below a further property of the predicates $B = \{B_m : m \geq 0\}$ considered will be needed, called the

Random Generation Hypothesis, abbreviated RGH. A predicate $B = \{B_m : m \geq 0\}$, where $B_m : X_m \longrightarrow \{0,1\}$, satisfies the RGH if there exists an algorithm running in time polynomial in m which on input m it will output a random pair $< x, y >$, where $x \in X_m$ and $y \in \{0,1\}$ such that $B_m(x) = y$.

Example 5.3 *The predicates in examples 5.1 and 5.2 satisfy RGH.*

From now on and for the rest of this section whenever Yao's XOR theorem is applied to a predicate B it will be assumed that B satisfies RGH. The necessity of this assumption will be become apparent in the course of the formal proof of the XOR lemma.

The most interesting result on xoring is the following

Theorem 5.5 *(**Yao's XOR Theorem, A. Yao**) Let M be an infinite set of integers, let g, h be polynomial time computable functions such that $g(m) \geq 2^{h(m)} \geq (\log_2 m)^{1+\epsilon}$, for some $\epsilon > 0$. Let $B = \{B_m : m \geq 0\}$, $B_m : X_m \longrightarrow \{0,1\}$ be a family of predicates on the family $X = \{X_m : m \geq 0\}$. If there exists a polynomial Q and a polynomial size circuit $C = \{C_m : m \geq 0\}$ such that for all $m \in M$,*

$$Pr\left[x \in (X_m)^{g(m)} : B_m^{(g)}(x) = C_m(u)\right] \geq \frac{1}{2} + \frac{1}{Q(m)},$$

then for any polynomial P there exists a polynomial size circuit $C' = \{C'_m : m \geq 0\}$ such that for all but a finite number of $m \in M$,

$$Pr\left[x \in X_m : B_m(x) = C'_m(x)\right] \geq 1 - \frac{1}{P(m)}. \bullet$$

The above theorem is in fact an immediate consequence of the following

Theorem 5.6 *(**Yao's XOR Lemma, A. Yao**) Let M be an infinite set of integers, and let $0 < \epsilon_m, \delta_m < 1$, for each $m \in M$. Let $B = \{B_m : m \geq 0\}$, $B_m : X_m \longrightarrow \{0,1\}$ be a family of predicates on the family $X = \{X_m : m \geq 0\}$. If there exists a polynomial size circuit $C = \{C_m : m \geq 0\}$ such that for all $m \in M$,*

$$Pr\left[< x, x' > \in X_m \times X_m : B_m^{(2)}(x, x') = C_m(x, x')\right] \geq \frac{1}{2} + \epsilon_m,$$

then there exists a polynomial size circuit $C' = \{C'_m : m \geq 0\}$ such that for all $m \in M$,

$$Pr\left[x \in X_m : B_m(x) = C'_m(x)\right] \geq \frac{1}{2} + (1 - \delta_m) \cdot \sqrt{\frac{\epsilon_m}{2}}. \bullet$$

Proof of the XOR theorem from the XOR lemma: Let B, C, g, Q satisfy the hypothesis of the XOR theorem. Let P be an arbitrary polynomial. Put $\delta_m = 1/P(m)$. Using the projection theorem, it can be assumed without loss of generality that for all m, $g(m) = 2^{h(m)}$. The idea is for each m to apply the XOR lemma a sufficient number of times, namely $h(m)$-times. Indeed, fix $m \in M$; define by induction $\epsilon_{i,m} > 0$ and circuits $C^i = \{C_m^i : m \geq 0\}$ as follows, for $i = 1, \ldots, h(m)$,

$$\epsilon_{1,m} = \frac{1}{Q(m)}, \ \epsilon_{i+1,m} = (1 - \delta_m) \cdot 2^{-\frac{1}{2}} \cdot \sqrt{\epsilon_{i,m}}.$$

Assume that the circuits $C^1 = C, C^2, \ldots, C^i$ have already been defined. Apply the XOR lemma to the circuit C^i and the xor

$$\left(B_m^{(2^{h(m)-i-1})}\right)^{(2)},$$

to find a polynomial size circuit C^{i+1} such that

$$Pr\left[B_m^{(2^{h(m)-i-1})}(x) = C_m^{i+1}(x)\right] \geq \frac{1}{2} + (1 - \delta_m) \cdot 2^{-\frac{1}{2}} \cdot \sqrt{\epsilon_{i,m}}.$$

It will be shown that the circuit $C' = C^{h(m)}$ satisfies the conclusion of the XOR theorem. It is clear that

$$Pr[x \in X_m : B_m(x) = C_m^{h(m)}(x)] \geq \frac{1}{2} + \epsilon_{h(m),m}.$$

Moreover,

$$\epsilon_{h(m),m} = \gamma_m \cdot \beta_m \cdot \alpha_m,$$

where

$$\gamma_m = 2^{-1/2-1/2^2-\cdots-1/2^{h(m)-1}},$$
$$\beta_m = (1 - \delta_m)^{1+1/2+1/2^2+\cdots+1/2^{h(m)-2}},$$
$$\alpha_m = \left(\frac{1}{Q(m)}\right)^{1/2^{h(m)}}. \tag{15}$$

It is now easy to show that

$$\lim_{m \to \infty} \gamma_m = \frac{1}{2}, \beta_m \approx (1 - \delta_m)^2, \lim_{m \to \infty} \alpha_m = 1.$$

Hence,

$$\epsilon_{h(m),m} \approx \frac{1}{2}(1 - \delta_m)^2.$$

This determines the values of the sequence $\epsilon_{h(m),m}$, and completes the proof of the reduction of the XOR theorem to the XOR lemma.•

An immediate corollary of the XOR theorem and the projection theorem is the following

Theorem 5.7 (**Multiple XOR Theorem**) *Let M be an infinite set of integers, let g, h be polynomial time computable functions such that $g(m) \geq 2^{h(m)} \geq (\log_2 m)^{1+\epsilon}$, for some $\epsilon > 0$ and let k be a function from the positive integers to the positive integers. For each i, let $B^i = \{B^i_m : m \geq 0\}$, be a family of predicates. Put $B = B^{(g)}$. If there exists a polynomial Q and a polynomial size circuit $C = \{C_m : m \geq 0\}$ such that for all $m \in M$,*

$$Pr\left[x \in X^1_m \times \cdots \times X^{g(m)}_m : B^{(g)}_m(x) = C_m(u)\right] \geq \frac{1}{2} + \frac{1}{Q(m)},$$

then for any polynomial P there exists polynomial size circuits $C^1, C^2, \ldots$ such that for all i, and all but a finite number of $m \in M$,

$$Pr\left[x_i : B^i_m(x_i) = C^i_m(x_i)\right] \geq 1 - \frac{1}{P(m)}.\bullet$$

EXERCISES

1: Show that the circuits defined via (14) satisfy the conclusion of the projection theorem.

2: Show that the general case of the projection theorem follows from the case: for all m, $g(m) = 2$.

3: Extend theorem 5.4 to xors of more than one predicate.

4: RGH for a predicate B should not be confused with predicting B. Show that the predicates defined in examples 5.1, 5.2 satisfy RGH.

5: Show that $\lim_{m \to \infty} \alpha_m = 1$, where α_m was defined in equation (15). **Hint:** Use the hypothesis $2^{h(m)} \geq (\log_2 m)^{1+\epsilon}$.

5.4 PROOF OF THE XOR LEMMA

This subsection will be divided into two parts. The first part will be concerned with an intuitive, geometric discussion of the proof of the XOR lemma. The formal aspects of the proof will be discussed in part two.

PART 1: INTUITIVE PROOF OF THE XOR LEMMA

Assume that the hypothesis of the XOR lemma is true. i.e. M is an infinite set of integers, $0 < \epsilon_m, \delta_m < 1$, for each $m \in M$. Let $B = \{B_m : m \geq 0\}$, $B_m : X_m \longrightarrow \{0, 1\}$ be a family of predicates on the family $X = \{X_m : m \geq 0\}$. Assume there exists a polynomial size circuit $C = \{C_m : m \geq 0\}$ such that for all $m \in M$,

$$Pr\left[< x, x' > \in X_m \times X_m : B^{(2)}_m(x, x') = C_m(x, x')\right] \geq \frac{1}{2} + \epsilon_m.$$

Put

$$\eta_m = \left(1 - \frac{\delta_m}{2}\right) \cdot \sqrt{\frac{\epsilon_m}{2}}.$$

It is required to find a polynomial size circuit $C' = \{C'_m : m \geq 0\}$ such that for all $m \in M$,

$$Pr[x \in X_m : B_m(x) = C'_m(x)] \geq \frac{1}{2} + \eta_m. \tag{16}$$

For each $x \in X_m$ let

$$k(x) = |\{x' \in X_m : B_m^{(2)}(x, x') = C_m(x, x')\}|, p(x) = \frac{k(x)}{|X_m|}.$$

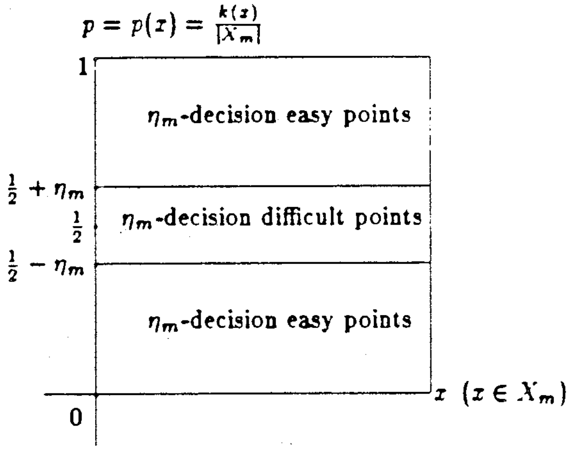

$p = p(x) = \frac{k(x)}{|X_m|}$
1
η_m-decision easy points
$\frac{1}{2} + \eta_m$
$\frac{1}{2}$
η_m-decision difficult points
$\frac{1}{2} - \eta_m$
η_m-decision easy points
x $(x \in X_m)$
0

Figure 2: Distribution of the η_m-decision points

Figure 5.4 pictures the difficulty involved in deciding the values of the predicate $B = \{B_m : m \geq 0\}$. One can distinguish the following two cases.

Case 1: $(\exists x \in X_m)(|p(x) - 1/2| \geq \eta_m)$.

In other words, in the language of Figure 5.4, in this case there is an η_m-decision easy point. Call such a point x_0. Define a polynomial size circuit as follows:

$$C'_m(x') = \begin{cases} C_m(x_0, x') \oplus B_m(x_0) & \text{if } p(x_0) \geq 1/2 + \eta_m \\ C_m(x_0, x') \oplus B_m(x_0) \oplus 1 & \text{if } p(x_0) \leq 1/2 - \eta_m \end{cases}$$

It is easy to see that C' satisfies the requirements of inequality 16. Indeed, on the one hand if $p(x_0) \geq 1/2 + \eta_m$ then

$$p(x_0) = Pr[x' \in X_m : B_m(x') = C'_m(x')] \geq \frac{1}{2} + \eta_m,$$

while on the other hand if $p(x_0) \leq 1/2 - \eta_m$ then

$$1 - p(x_0) = Pr[x' \in X_m : B_m(x') = C'_m(x')] \geq \frac{1}{2} + \eta_m.$$

Case 2: $(\forall x)(|p(x) - 1/2| < \eta_m)$.

In this case it will be assumed that t_m is an odd integer. This restriction however, is only technical and will present no difficulties for the formal proof. Figure 5.4 describes the distribution of points in the present case.

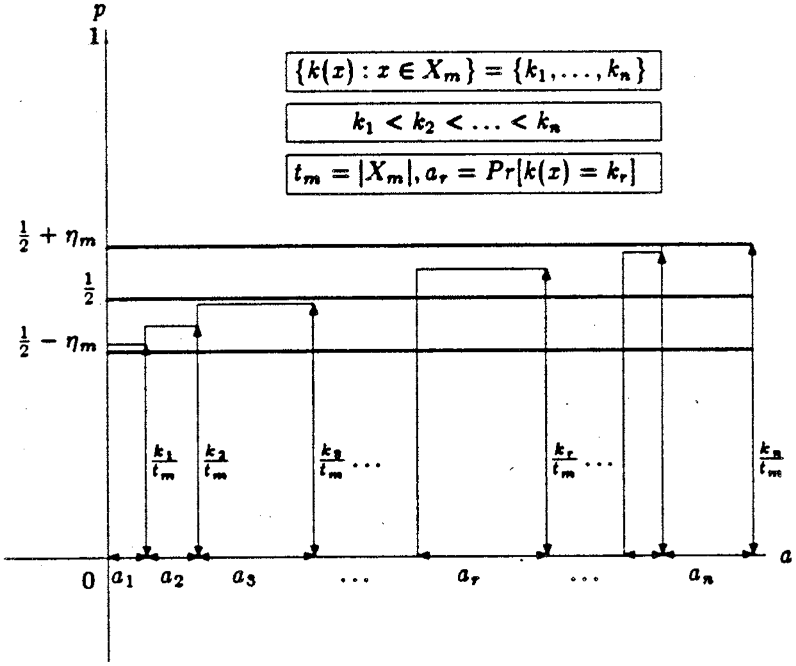


Figure 3: Graph of $p = \frac{k(x)}{t_m}, a = Pr[x \in X_m : k(x) = k]$

Put $t_m = |X_m|$. For each $x \in X_m$, $b \in \{0, 1\}$, let

$$K(x) = \{x' \in X_m : C_m(x, x') = B_m^{(2)}(x, x')\},$$

$$K_b(x) = \{x' \in X_m : C_m(x, x') = B_m^{(2)}(x, x') \oplus b\},$$

$$k(x) = |K(x)|, k_b(x) = |K_b(x)|.$$

It is not difficult to show that the following properties must hold for all $x \in X_m$,

1. $X_m = K_0(x) \cup K_1(x)$ and $t_m = k_0(x) + k_1(x)$.
2. $K(x) = K_{B_m(x)}(x)$ and $k(x) = k_{B_m(x)}(x)$.

For each $k \leq t_m$ define the bucket $\Sigma(k) = \{x \in X_m : k(x) = k\}$ and put $\sigma(k) = |\Sigma(k)|$. Figure 5.4 pictures the situation in which the function $k(x)$ assumes only the values $k_1 < \cdots < k_n$, in increasing order. Notice that if $k \notin \{k_1, \ldots, k_n\}$ then $\sigma(k) = 0$. For each $k \leq t_m$, let

$$a(k) = Pr[x \in X_m : k(x) = k] = \frac{\sigma(k)}{t_m}$$

i.e. $a(k)$ is the ratio of points in X_m which lie in the bucket $\Sigma(k)$. Put $a_r = a(k_r)$, $\sigma_r = \sigma(k_r)$. The r-th rectangle pictured, has a base of length a_r, and height equal to $p(x) = k_r/t_m$, for each $x \in \Sigma(k_r)$. It is clear that

$$\sum_{k=1}^{t_m} a(k) = \sum_{k=1}^{t_m} Pr[x \in X_m : k(x) = k] = 1. \tag{17}$$

The polynomial size circuit C which predicts the predicate B is based on a comparison of sizes of buckets. To be more specific one defines C'_m as follows:

Input: x
Step 1: Compute $k = k_0(x)$
Output:

$$C'_m(x) = \begin{cases} 0 & \text{if } \sigma(k) \geq \sigma(t_m - k) \\ 1 & \text{if } \sigma(k) < \sigma(t_m - k) \end{cases}$$

It will now be shown that the circuit defined above satisfies the requirements of the XOR lemma. It is a consequence of the definition of C' that

$$Pr[x \in X_m : C'_m(x) = B_m(x)] \geq \sum_{k > t_m/2} \max\{a(k), a(t_m - k)\} \tag{18}$$

Indeed, for $i = 0, 1$ put

$$p_i = Pr[C_m(x) = B_m(x) = i],$$

and observe that

$$Pr[x \in X_m : C'_m(x) = B_m(x)] = p_0 + p_1. \tag{19}$$

However,

$$p_0 = \sum_{\sigma(k) \geq \sigma(t_m - k)} Pr[k_0(x) = k \text{ and } B_m(x) = 0],$$

$$p_1 = \sum_{\sigma(k) < \sigma(t_m - k)} Pr[k_0(x) = k \text{ and } B_m(x) = 1].$$

It follows that,

$$p_0 + p_1 \geq$$

$$\sum_{k>t_m/2, \sigma(k)>\sigma(t_m-k)} Pr[k_0(x) = k, B_m(x) = 0]+$$

$$\sum_{k<t_m/2, \sigma(k)>\sigma(t_m-k)} Pr[k_0(x) = k, B_m(x) = 0]+$$

$$\sum_{k>t_m/2, \sigma(k)<\sigma(t_m-k)} Pr[k_0(x) = k,, B_m(x) = 1]+$$

$$\sum_{k<t_m/2, \sigma(k)<\sigma(t_m-k)} Pr[k_0(x) = k,, B_m(x) = 1].$$

Using the fact

$$k(x) = \begin{cases} k_0(x) & \text{if } B_m(x) = 0 \\ t_m - k_0(x) & \text{if } B_m(x) = 1, \end{cases}$$

it follows easily that

$$Pr[x \in X_m : C'_m(x) = B_m(x)] \geq$$

$$\sum_{k>t_m/2, \sigma(k)>\sigma(t_m-k)} Pr[k(x) = k] + \sum_{k<t_m/2, \sigma(k)>\sigma(t_m-k)} Pr[k(x) = k] =$$

$$\sum_{k>t_m/2} \max\{a(k), a(t_m - k)\},$$

which completes the proof of (18).

Let Ω denote the area under the pictured graph. The area of the r-th rectangle is equal to $a_r \cdot (k_r/t_m)$. Hence, Ω is equal to the sum of the areas of the n regtangles. It follows that on the one hand

$$\Omega = \sum_k \frac{k}{t_m} \cdot a(k), \tag{20}$$

and on the other hand

$$\Omega = Pr\left[< x, x' > \in X_m \times X_m : B_m^{(2)}(x, x') = C_m(x, x')\right] \geq \frac{1}{2} + \epsilon_m. \tag{21}$$

Let $d(k) = k/t_m - 1/2$ i.e. if the r-th rectangle lies above (respectively below) the horizontal line of height $1/2$ then $d(k_r)$ is the positive (respectively negative) distance of the heighest point of the rectangle from the horizontal line drawn at height $1/2$. It follows from (20) and (17) that

$$\Omega = \frac{1}{2} + \sum_k d(k) \cdot a(k) =$$

$$\frac{1}{2}+\sum_{k>t_m/2} d(k)\cdot a(k)+\sum_{k<t_m/2} d(k)\cdot a(k)=$$

$$\frac{1}{2}+\sum_{k>t_m/2} d(k)\cdot a(k)+\sum_{k>t_m/2} -d(k)\cdot a(t_m-k)=$$

$$\frac{1}{2}+\sum_{1/2+\eta_m>k/t_m>1/2} d(k)\cdot(a(k)-a(t_m-k))\leq$$

$$\frac{1}{2}+\eta_m\cdot\sum_{1/2+\eta_m>k/t_m>1/2} |a(k)-a(t_m-k)|. \qquad (22)$$

It follows from (21), (22) that

$$\sum_{k/t_m>1/2} |a(k)-a(t_m-k)|= \qquad (23)$$

$$\sum_{1/2+\eta_m>k/t_m>1/2} |a(k)-a(t_m-k)|\geq\frac{\epsilon_m}{\eta_m}=\frac{\sqrt{2\epsilon_m}}{1-\delta_m/2}.$$

Using the fact that

$$|a-a'|=\max\{a,a'\}-\min\{a,a'\},$$

$$a+a'=\max\{a,a'\}+\min\{a,a'\},$$

$$1=\sum_k a(k)=\sum_{k>t_m/2}(a(k)+a(t_m-k)), \qquad (24)$$

it is easy to show that

$$\sum_{k>t_m/2}\max\{a(k),a(t_m-k)\}=\frac{1}{2}+\sum_{k>t_m/2}|a(k)-a(t_m-k)|.$$

Using this, as well as inequalities (18), (23) one obtains that

$$Pr[x\in X_m : C'_m(x)=B_m(x)]\geq\frac{1}{2}+\frac{1}{2}\cdot\frac{\sqrt{2\epsilon_m}}{1-\delta_m/2}\geq\frac{1}{2}+\eta_m.$$

This completes the intuitive proof.

PART 2: FORMAL PROOF OF THE XOR LEMMA

Assume that the hypothesis of the XOR lemma is true. i.e. M is an infinite set of integers, $0<\epsilon_m,\delta_m<1$, for each $m\in M$. Let $B=\{B_m : m\geq 0\}$, $B_m : X_m \longrightarrow X_m$ be a family of predicates on the family $X=\{X_m : m\geq 0\}$.

Assume there exists a polynomial size circuit $C = \{C_m : m \geq 0\}$ such that for all $m \in M$,

$$\beta = Pr\left[< x, x' > \in X_m \times X_m : B_m^{(2)}(x, x') = C_m(x, x')\right] \geq \frac{1}{2} + \epsilon_m. \quad (25)$$

Fix $m \in M$ and put $\epsilon_m = \epsilon, \delta_m = \delta$, $\eta_m = \eta$. Also, define

$$s = 2 \cdot \left(\left\lceil \frac{\log_2 m}{\epsilon \cdot \delta^2} \right\rceil\right)^5 + 1, \ell = s^9.$$

i.e. s is odd. Throughout the proof below x (respectively y) with subscripts or superscripts will range over elements of X_m (respectively of $\{0, 1\}$). The circuit C'_m which predicts the predicate B_m with an η advantage is defined as follows:

Input: x

Step 1: Let $< x_1, y_1 >, \ldots, < x_\ell, y_\ell >$ be a random sample such that $B_m(x_i) = y_i$, for all $i = 1, \ldots, \ell$.

Step 2: Let $< x'_1, y'_1 >, \ldots, < x'_s, y'_s >$ be a random sample such that $B_m(x'_j) = y'_j$, for all $j = 1, \ldots, s$.

Step 3: Compute

$$K'(x_i, y_i) = \{j \leq s : C_m(x_i, x'_j) = y_i \oplus y'_j\}, k'(x_i, y_i) = |K'(x_i, y_i)|,$$

$$\Sigma'(k) = \{i \leq \ell : k'(x_i, y_i) = k\}, \sigma'(k) = |\Sigma'(k)|.$$

Step 4:

$$\text{Case 1}: \ (\exists i) \left| \frac{k'(x_i, y_i)}{s} - \frac{1}{2} \right| \geq \left(1 - \frac{\delta}{2}\right) \cdot \sqrt{\frac{\epsilon}{2}}.$$

In this case compute

$$k'_0 = \min\{k'(x_i, y_i) \text{ as in case 1 } : i = 1, \ldots, \ell\},$$

$$i_0 = \min\{i \leq \ell : i \in \Sigma'(k'_0)\}, \text{ and}$$

$$a = \begin{cases} C_m(x_{i_0}, x) \oplus y_{i_0} & \text{if } k'_0/s > 1/2 \\ C_m(x_{i_0}, x) \oplus y_{i_0} \oplus 1 & \text{if } k'_0/s < 1/2 \end{cases}$$

$$\text{Case 2}: \ (\forall i) \left| \frac{k'(x_i, y_i)}{s} - \frac{1}{2} \right| < \left(1 - \frac{\delta}{2}\right) \cdot \sqrt{\frac{\epsilon}{2}}.$$

In this case compute

$$k' = k'(x, 0) \text{ and}$$

$$a = \begin{cases} 0 & \text{if } \sigma(k') \geq \sigma(s - k') \\ 1 & \text{if } \sigma(k') < \sigma(s - k') \end{cases}$$

Output: $C_m(x) = a$

The rest of this section will be devoted to a proof of

$$\gamma = Pr\left[B_m(x) = C'_m(x)\right] \geq \frac{1}{2} + \eta. \tag{26}$$

Let G denote the event $C'_m(x) = B_m(x)$, A the event that after execution of step 3, case 1 occurs in step 4, and let $\overline{A}$ be the event that after execution of step 3, case 2 occurs in step 4. It is then clear that

$$\gamma = \alpha \cdot Pr[A] + \overline{\alpha} \cdot Pr[\overline{A}],$$

where $\alpha = Pr[G|A], \overline{\alpha} = Pr[G|\overline{A}]$.

The proof of the theorem will be divided into two claims. In CLAIM 1 a lower bound on $\alpha Pr[A]$ will be determined and in CLAIM 2 a lower bound on $\overline{\alpha} Pr[\overline{A}]$.

CLAIM 1: $\alpha Pr[A] \geq (1/2 + (1 - \frac{3}{4}\delta)\sqrt{\epsilon/2})Pr[A] - \exp[-\sqrt{s}]$

The proof is in several steps. For each x, y put

$$g(x,y) = Pr\left[C_m(x,x') = y \oplus y'\right]$$

and consider the events

$$F_1 : \left(\frac{k'(x,y)}{s} - \frac{1}{2}\right) \geq \left(1 - \frac{\delta}{2}\right) \cdot \sqrt{\frac{\epsilon}{2}} \text{ and } g(x,y) < \frac{1}{2} + \left(1 - \frac{3\delta}{4}\right) \cdot \sqrt{\frac{\epsilon}{2}},$$

$$F_2 : \left(\frac{k'(x,y)}{s} - \frac{1}{2}\right) \leq -\left(1 - \frac{\delta}{2}\right) \cdot \sqrt{\frac{\epsilon}{2}} \text{ and } g(x,y) > \frac{1}{2} - \left(1 - \frac{3\delta}{4}\right) \cdot \sqrt{\frac{\epsilon}{2}}.$$

It will be shown that

Subclaim 1: $Pr[F_1 \cup F_2] \leq \exp(-2s/3)$.

Since, $k'(x, y \oplus 1) = s - k'(x,y), g(x, y \oplus 1) = 1 - g(x,y)$ it is clear that for all x, y,

$$< x, y > \in F_1 \Leftrightarrow < x, y \oplus 1 > \in F_2,$$

and hence

$$Pr[F_1] = Pr[F_2].$$

Hence, it is enough to find an upper bound for $Pr[F_1]$. The idea is to think of $k'(x,y)$ as a Monte-Carlo computation of $g(x,y)$. Let $\theta = (\delta/4) \cdot \sqrt{\epsilon/2}$. Apply Bernshtein's law of large numbers to obtain

$$Pr\left[\left|\frac{k'(x,y)}{s} - g(x,y)\right| > \theta\right] \leq 2\exp(-s\theta^2)$$

Put

$$F_1^0 = \{< x, y > \in F_1 : g(x,y) > 1/4\}, F_1^1 = \{< x, y > \in F_1 : g(x,y) \leq 1/4\}.$$

and notice that $Pr[F_1] = Pr[F_1^0] + Pr[F_1^1]$. Then, using the definition of s,

$$Pr[F_1^0] \leq Pr\left[\frac{k'(x,y)}{s} - g(x,y) > \theta\right] \leq \tag{27}$$

$$2\exp(-s\theta^2) \leq 2\exp\left(-\frac{s\delta^2\epsilon}{32}\right) \leq 2\exp(-s^{3/4}).$$

In addition, it is true that

$$Pr[F_1^1] \leq Pr\left[\frac{k'(x,y)}{s} - \frac{1}{4} \geq \theta\right].$$

Thus, applying the previous argument to the right side of the above inequality, with $g(x,y) = 1/4$ it follows that

$$Pr[F_1^1] \leq 2\exp(-s^{3/4}). \tag{28}$$

Subclaim 1 now follows from inequalities (27), (28).

Consider the events

$$A_1 : \frac{k_0'}{s} > \frac{1}{2} \text{ and } (\forall i \in \Sigma(k_0'))\left[g(x_i,y_i) \geq \frac{1}{2} + \left(1 - \frac{3\delta}{4}\right)\sqrt{\frac{\epsilon}{2}}\right],$$

$$A_2 : \frac{k_0'}{s} < \frac{1}{2} \text{ and } (\forall i \in \Sigma(k_0'))\left[g(x_i,y_i) < \frac{1}{2} - \left(1 - \frac{3\delta}{4}\right)\sqrt{\frac{\epsilon}{2}}\right],$$

$$L = A \cap (A_1 \cup A_2), \text{ and } \overline{L} = \text{ the complement of } L.$$

Now, it will be shown that

Subclaim 2: $Pr[A \cap \overline{L}] \leq \exp(-\sqrt{s})$.

Indeed, $Pr[A \cap \overline{L}] \leq Pr[A \cap \overline{A_1}] + Pr[A \cap \overline{A_2}] \leq$

$$Pr\left[\frac{k_0'}{s} > \frac{1}{2} \text{ and } (\exists i \leq \ell)\left(g(x_i,y_i) < \frac{1}{2} + \left(1 - \frac{3\delta}{4}\right)\sqrt{\frac{\epsilon}{2}}\right)\right] +$$

$$Pr\left[\frac{k_0'}{s} < \frac{1}{2} \text{ and } (\exists i \leq \ell)\left(g(x_i,y_i) > \frac{1}{2} - \left(1 - \frac{3\delta}{4}\right)\sqrt{\frac{\epsilon}{2}}\right)\right] \leq$$

$$\ell Pr[F_1] + \ell Pr[F_2] \leq$$

$$2 \cdot s^9 \cdot \exp\left(-s^{2/3}\right) \leq \exp(-\sqrt{s}),$$

which completes the proof of subclaim 2. Consequently,

$$Pr[G|L] \cdot Pr[A \cap \overline{L}] \leq \exp(-\sqrt{s}). \tag{29}$$

Now to finish the proof of claim 1, notice that from the definition of C'_m on the one hand,

$$Pr[G|L, A_1] \geq E[g(x_i, y_i)|L, A_1] \geq \frac{1}{2} + \left(1 - \frac{3\delta}{4}\right)\sqrt{\frac{\epsilon}{2}},$$

$$Pr[G|L, A_2] \geq E[1 - g(x_i, y_i)|L, A_2] \geq \frac{1}{2} + \left(1 - \frac{3\delta}{4}\right)\sqrt{\frac{\epsilon}{2}},$$

and thus using

$$Pr[G|L] = Pr[G|L, A_1] \cdot Pr[A_1|L] + Pr[G|L, A_2] \cdot Pr[A_2|L]$$

it follows that

$$Pr[G|L] \geq \frac{1}{2} + \left(1 - \frac{3\delta}{4}\right)\sqrt{\frac{\epsilon}{2}}. \tag{30}$$

On the other hand it is clear that

$$\alpha Pr[A] = Pr[G|A]Pr[A] \geq Pr[G|L]Pr[L] =$$

$$Pr[G|L](Pr[A] - Pr[A \cap \overline{L}]) = Pr[G|L]Pr[A] - Pr[G|L]Pr[A \cap \overline{L}]). \tag{31}$$

Claim 1 is now a consequence of (30), (31) and (29).

CLAIM 2: $\overline{\alpha} Pr[\overline{A}] \geq (1/2 + (1 - \frac{3}{4}\delta)\sqrt{\epsilon/2})Pr[\overline{A}] - 1/(4s^2)$.

For each i, j consider the $\ell \cdot s$ independent random variables $X_{i,j}$ defined by

$$X_{i,j}(x, x') = \begin{cases} 1 & \text{if } C_m(x, x') = B_m^{(2)}(x, x') \\ 0 & \text{if } C_m(x, x') \neq B_m^{(2)}(x, x'). \end{cases}$$

and let

$$\xi(x, x') = \frac{1}{\ell \cdot s} \cdot \sum_{i,j} X_{i,j}(x, x'),$$

$$q_k(x, x') = \frac{1}{\ell} \cdot \left| \left\{ 1 \leq i \leq \ell : \sum_{j=1}^{s} X_{i,j}(x, x') = k \right\} \right|.$$

It is clear that q_k is the random variable corresponding to the quantity $\sigma(k)$. Setting $d_k = k/s - 1/2$, and using the fact that s is odd, it ia easy to see that

$$\xi(x, x') = \frac{1}{2} + \sum_{k > s/2} d_k \cdot (q_k(x, x') - q_{s-k}(x, x')).$$

Let a'_k be the expectation of the random variable q_k i.e. $a'_k = E[q_k]$. a'_k can be regarded as a Monte-Carlo computation of q_k. It is clear from (25) that for all

i, j, $\beta = E[X_{i,j}]$ and hence $\beta = E[\xi]$, using the expectation theorem. It follows that,

$$\beta = E[\xi] = \frac{1}{2} + \sum_{k>s/2} d_k \cdot (a'_k - a'_{s-k}) \leq \frac{1}{2} + \sum_{k>s/2} d_k \cdot |a'_k - a'_{s-k}|. \qquad (32)$$

Next the following two cases will be considered

Case 1: There exists $k \leq s$ such that

$$a'_k \geq \frac{1}{s^3} \text{ and } \left|\frac{k}{s} - \frac{1}{2}\right| \geq \eta.$$

Let k be as in case 1. Using the Weak Law of Large Numbers one obtains that

$$Pr\left[|a'_k - q_k| > \frac{s-1}{s^4}\right] \leq \frac{s^8}{4\ell(s-1)^2} \leq \frac{1}{4s^3}.$$

Consequently,

$$Pr\left[q_k < \frac{1}{s^4}\right] = Pr\left[q_k < \frac{1}{s^4} \text{ and } |a'_k - q_k| > \frac{s-1}{s^4}\right] \leq \frac{1}{4s^3}.$$

It follows that,

$$Pr[A] \geq Pr\left[q_k \geq \frac{1}{s^4}\right] \geq 1 - \frac{1}{4s^3}.$$

Hence, the desired inequality (26) follows easily using the result of CLAIM 1.

Case 2: For all $k \leq s$,

$$\left|\frac{k}{s} - \frac{1}{2}\right| \geq \eta \Rightarrow a'_k \leq \frac{1}{s^3}.$$

Consider the event J defined by

$$(\forall k)\left(|a'_k - a'_{s-k}| \geq \frac{1}{s^2} \Rightarrow a'_k - a'_{s-k}, q_k - q_{s-k} \text{ have the same sign}\right).$$

It is an immediate application of the Weak Law of Large Numbers that

$$Pr[J] \geq 1 - \frac{1}{4s^2}, \qquad (33)$$

(see exercise 1.) Thus,

$$\overline{\alpha} Pr[\overline{A}] = Pr[G|\overline{A}]Pr[\overline{A}] = Pr[J]Pr[G|\overline{A}, J]Pr[\overline{A}] \geq$$

$$Pr[G|\overline{A}, J]Pr[\overline{A}] - \frac{1}{4s^2}.$$

However, (32) and the assumption in Case 2 imply that

$$\frac{1}{2}+\epsilon \leq \beta \leq \sum_{1/2<k/s<1/2+\eta} d_k \cdot |a'_k - a'_{s-k}| + \frac{s}{s^3}. \tag{34}$$

It follows from (34) that

$$\sum_{1/2<k/s<1/2+\eta} |a'_k - a'_{s-k}| \geq \frac{\epsilon - 1/s^2}{\eta}. \tag{35}$$

Using the definition of J, and (24), one obtains

$$Pr[G|\overline{A}, J] \geq \tag{36}$$

$$\sum_{1/2<k/s, |a'_k - a'_{s-k}| \geq 1/s^2} \max\{a'_k, a'_{s-k}\} +$$

$$\sum_{1/2<k/s, |a'_k - a'_{s-k}| < 1/s^2} \left\{\min\{a'_k, a'_{s-k}\} - \frac{1}{s^2}\right\}$$

$$\geq \sum_{1/2<k/s} \max\{a'_k, a'_{s-k}\} - \frac{1}{2s} =$$

$$\frac{1}{2} + \frac{1}{2} \sum_{1/2<k/s} |a'_k - a'_{s-k}| - \frac{1}{2s}.$$

This and (35) imply the result in CLAIM 2.

Now the XOR lemma follows easily from CLAIM 1, CLAIM 2.•

EXERCISES

1: Prove inequality (33).

5.5 APPLICATIONS OF THE XOR LEMMA

There are three main applications of the XOR theorem. The first two concern unapproximable predicates and pseudo-random generators, and will be presented in the present subsection. The third one is the notion of strong (respectively weak) one way functions and will be presented in the next subsection (5.6).

APPLICATION 1: UNAPPROXIMABLE PREDICATES.

Definition 5.14 *Let P be any polynomial. A predicate $B = \{B_m : m \geq 0\}$ defined on the family $X = \{X_m : m \geq 0\}$ is $1/P$ unapproximable, and this will be abbreviated $UPR(X, B, 1/P)$, if there is no polynomial size circuit $C = \{C_m : m \geq 0\}$ such that for infinitely many m,*

$$Pr[x \in X_m : B_m(x) = C_m(x)] \geq \frac{1}{2} + \frac{1}{P(m)}.$$

Definition 5.15 *Let P be any polynomial. A predicate $B = \{B_m : m \geq 0\}$ defined on the family $X = \{X_m : m \geq 0\}$ is $(1/2 - 1/P)$-unapproximable, and this will be abbreviated $UPR(X, B, 1/2 - 1/P)$, if there is no polynomial size circuit $C = \{C_m : m \geq 0\}$ such that for infinitely many m,*

$$Pr[x \in X_m : B_m(x) = C_m(x)] \geq 1 - \frac{1}{P(m)}.$$

Remark: Notice that the above definition of B is $1/P$ unapproximable is equivalent to $(\forall C)(C$ does not P-predict $B)$ in definition 5.2.

An immediate application of the XOR theorem is the following

Theorem 5.8 *Let g be a positive integer valued function such that for some $\epsilon > 0$, $g(m) \geq [\log_2 m]^{1+\epsilon}$. Then for all X, B the following hold:*

1. $(\exists P)UPR(X, B, 1/P) \Rightarrow (\forall P)UPR(X, B, 1/2 - 1/P)$.
2. $(\exists P)UPR(X, B, 1/2 - 1/P) \Rightarrow (\forall P)UPR(X, B^{(g)}, 1/P)$. •

APPLICATION 2: PSEUDOM-RANDOM GENERATORS.

Recall the definition of pseudo-random generator on the family $X = \{X_m : m \geq 0\}$ given in definition 5.4. To any such pseudo-random generator G associate the sequence $b^G_{m,0}(x), \ldots, b^G_{m,Q(m)-1}(x)$ of bits generated by G, where for each m, $b^G_{m,i}$ is the $i-$th bit generated by G_m on input $x \in X_m$.

Definition 5.16 *Let P be a polynomial and $G = \{G_m : m \geq 0\}$ a pseudo-random generator on the family $X = \{X_m : m \geq 0\}$. The generator G, P-passes the $1/P$ Blum-Micali Test, and this will be abbreviated $BMT(X, G, 1/P)$, if for all polynomial size circuits $C = \{C_m : m \geq 0\}$ the following cannot hold for infinitely many m: there exists an $i < Q(m)$ such that*

$$Pr\left[G_m(x) : C_m(b^G_{m,0}(x), \ldots, b^G_{m,i-1}(x)) = b_{m,i}(x)\right] \geq \frac{1}{2} + \frac{1}{P(m)}.$$

Definition 5.17 *Let P be a polynomial and $G = \{G_m : m \geq 0\}$ a pseudo-random generator on the family $X = \{X_m : m \geq 0\}$. The generator G, P-passes the $1/2 - 1/P$ Blum-Micali Test, and this will be abbreviated $BMT(X, G, 1/2 -$

$1/P$), if for all polynomial size circuits $C = \{C_m : m \geq 0\}$ the following cannot hold for infinitely many m: there exists an $i < Q(m)$ such that

$$Pr\left[G_m(x) : C_m(b^G_{m,0}(x), \ldots, b^G_{m,i-1}(x)) = b_{m,i}(x)\right] \geq 1 - \frac{1}{P(m)}.$$

Remark: Notice that the above definition of G is $1/P$ unapproximable is equivalent to $(\forall C)(C$ does not P-predict $G)$ in definition 5.4.

Let G be any pseudo-random generator and let $b^G_{m,0}(x), \ldots, b^G_{m,Q(m)-1}(x)$ be the sequence of bits generated by G on input x. For any positive integer valued function g let $\{b^{G^{(g)}}_{m,i} : m \geq 0\}$ denote the g-xor of the predicate $\{b^G_{m,i} : m \geq 0\}$. The g-xor $G^{(g)}$ of the generator G is such that

1. For all m, $G^{(g)}_m : (X_m)^{g(m)} \longrightarrow S_{Q(m)}$ and
2. $G^{(g)}_m(u) =< b^{G^{(g)}}_{m,0}(u), \ldots, b^{G^{(g)}}_{m,Q(m)-1}(y) >$.

The following XOR theorem is proved exactly like theorem 5.5.

Theorem 5.9 (**XOR Theorem for Generators**) *Let M be an infinite set of integers, let g, h be polynomial time computable functions such that $g(m) \geq 2^{h(m)} \geq (\log_2 m)^{1+\epsilon}$, for some $\epsilon > 0$. Let $G = \{G_m : m \geq 0\}$, $G_m : X_m \longrightarrow S_{Q(m)}$ be a pseudo-reandom generator on the family $X = \{X_m : m \geq 0\}$. If there exists a polynomial P and a polynomial size circuit $C = \{C_m : m \geq 0\}$ such that for all $m \in M$ there exists an $i < Q(m)$ such that*

$$Pr\left[C_m(b^{G^{(g)}}_{m,0}(u), \ldots, b^{G^{(g)}}_{m,i-1}(u)) = b^{G^{(g)}}_{m,i}(u)\right] \geq \frac{1}{2} + \frac{1}{P(m)},$$

then for any polynomial P' there exists a polynomial size circuit $C' = \{C'_m : m \geq 0\}$ such that for all but a finite number of $m \in M$, there exists an $i < Q(m)$ such that

$$Pr\left[C'_m(b^G_{m,0}(u), \ldots, b^G_{m,i-1}(u)) = b^G_{m,i}(u)\right] \geq 1 - \frac{1}{P'(m)}. \bullet$$

Now, an immediate consequence of the XOR theorem for pseudo-random generators is the following

Theorem 5.10 *Let g be a positive integer valued function such that for some $\epsilon > 0$, $g(m) \geq \lfloor \log_2 m \rfloor^{1+\epsilon}$. Then for all X, G the following hold:*

1. $(\exists P) BMT(X, G, 1/P) \Rightarrow (\forall P) BMT(X, G, 1/2 - 1/P)$.
2. $(\exists P) BMT(X, G, 1/2 - 1/P) \Rightarrow (\forall P) BMT(X, G^{(g)}, 1/P)$. $\bullet$

5.6 ONE TO ONE, ONE WAY FUNCTIONS

The present subsection includes the third application of the XOR theorem.

Definition 5.18 **A polynomial size circuit** $C = \{C_m : m \geq 0\}$ **weakly** *(respectively* **strongly***)* P**-inverts** *the family* f *if for infinitely many* m,

$$Pr[x \in X_m : C_m(f_m(x)) = x] \geq \frac{1}{P(m)}$$

(respectively

$$Pr[x \in X_m : C_m(f_m(x)) = x] \geq 1 - \frac{1}{P(m)}.$$

Definition 5.19 $f = \{f_m : m \geq 0\}$ *is* **weak** *(respectively* **strong***)* **1 − 1, one way,** *if the following holds:*

$$(\forall P, C)(C \text{ does not weakly (respectively strongly) } P - \text{invert } f).$$

Theorem 5.11 *If the function* $f = \{f_m : m \geq 0\}$ *is a friendship function for the unapproximable predicate* $B = \{B_m : m \geq 0\}$, *then* $f = \{f_m : m \geq 0\}$ *is weak,1 − 1, one-way.*

Proof: Assume that the hypothesis of the theorem is true for the unapproximable predicate B and its friendship function f, but that the conclusion fails. Let $C = \{C_m : m \geq 0\}$ be a polynomial size circuit such that the following statement holds for infinitely many m,

$$Pr[x \in X_m : C_m(f_m(x)) = x] \geq \frac{1}{P(m)}.$$

Let M be the set of integers m which satisfy the above inequality. For each bit $b \in \{0,1\}$ let C_m^b be the following polynomial size circuit (due to Mike Fischer),

Input: x $(x \in X_m)$.

Step 1: Compute $y = C_m(x)$.

Output:

$$C_m^b(x) = \begin{cases} B_m(x) & \text{if } f_m(y) = x \\ b & \text{if } f_m(y) \neq x \end{cases}$$

Then the theorem will follow from the following

Claim: For all $m \in M$ there exists $b \in \{0,1\}$ such that

$$Pr[x \in X_m : B_m(x) = C_m^b(x)] \geq \frac{1}{2} + \frac{1}{2P(m)}.$$

Proof of the Claim: Let $m \in M$, and choose a bit $b \in \{0,1\}$ such that

$$Pr[x \in X_m : B_m(x) = b | f_m(C_m(x)) \neq x] \geq \frac{1}{2}.$$

Put

$$p = Pr[x \in X_m : f_m(C_m(x)) = x].$$

Then it can be shown that

$$Pr[x \in X_m : B_m(x) = C_m^b(x)] =$$

$$Pr[B_m(x) = C_m^b(x) \text{ and } f_m(C_m(x)) = x] +$$

$$Pr[B_m(x) = C_m^b(x) \text{ and } f_m(C_m(x)) \neq x] =$$

$$Pr[f_m(C_m(x)) = x] +$$

$$Pr[B_m(x) = C_m^b(x) | f_m(C_m(x)) \neq x] \cdot Pr[f_m(C_m(x)) \neq x] =$$

$$p + Pr[B_m(x) = C_m^b(x) | f_m(C_m(x)) \neq x] \cdot (1-p) \geq$$

$$p + \frac{1}{2}(1-p) = \frac{1}{2} + \frac{p}{2}.$$

Since by assumption

$$p = Pr[f_m(C_m(x)) \neq x] \geq \frac{1}{P(m)},$$

it follows that

$$Pr\left[x \in X_m : B_m(x) = C_m^b(x)\right] \geq \frac{1}{2} + \frac{1}{2P(m)},$$

and the proof of the claim is complete.

Since the set M is infinite, it follows from the claim that there exists a bit $b \in \{0,1\}$ and an infinite subset M' of M such that for each $m \in M'$,

$$Pr\left[x \in X_m : B_m(x) = C_m^b(x)\right] \geq \frac{1}{2} + \frac{1}{2P(m)}.$$

Then the circuit

$$C' = \{C_m^b : m \geq 0\},$$

$(2 \cdot P)$-predicts the predicate B, which is a contradiction.•

The following theorem is very important because it can be used in conjunction with theorem 5.1 to construct secure pseudo-random generators.

Theorem 5.12 (**Yao's One Way Function Theorem, A. Yao**) *The following three statements are equivalent:*

1. *There is a strong,* $1-1$, *one-way function.*

2. *There is a weak,* $1-1$, *one-way function.*

3. There is an unapproximable predicate $B = \{B_m : m \geq 0\}$ and a friendship function $f = \{f_m : m \geq 0\}$ corresponding to it.

Proof: (3) ⇒ (2)
This was proved in detail in theorem 5.11.
(2) ⇒ (1)
This is immediate from definition 5.18.
(1) ⇒ (3)
Let $f = \{f_m : m \geq 0\}$ be a strong one to one, one-way function such that $f_m : X_m \longrightarrow X_m$ is one to one, and onto. Let $B_{i,m} : X_m \longrightarrow \{0,1\}$ be the function defined by

$$x \longrightarrow B_{i,m}(x) = \text{the } i - \text{th bit of } f^{-1}(x).$$

Further, let $B_m : (X_m)^m \longrightarrow \{0,1\}$ be the following predicate

$$(x_1,\ldots,x_m) \longrightarrow B_m(x_1,\ldots,x_m),$$

where

$$B_m(x_1,\ldots,x_m) = B_{i,m}(x_1) \oplus \cdots \oplus B_{m,m}(x_m).$$

It will be shown that the function $g = \{g_m : m \geq 0\}$, where $g_m : (X_m)^m \longrightarrow (X_m)^m$ is defined through

$$g_m(x_1,\ldots,x_m) =< f_m(x_1),\ldots,g_m(x_m) >,$$

is a friendship function for the predicate B. Indeed, both conditions of definition 5.3 are easy to verify e.g. to prove condition 2. notice that

$$B_m(g_m(x_1,\ldots,x_m)) =$$

$$(1 - \text{st bit of } x_1) \oplus (2 - \text{nd bit of } x_2) \oplus \cdots \oplus (m - \text{th bit of } x_m).$$

It remains to show that the predicate B is unapproximable. Assume on the contrary that there exists a polynomial size circuit $C' = \{C'_m : m \geq 0\}$ and a polynomial P such that the following property holds for infinitely many m,

$$Pr[x \in (X_m)^m : B_m(x) = C'_m(x)] \geq \frac{1}{2} + \frac{1}{P(m)}.$$

Let M' be the set of indices $m \in M$ that satisfy the above inequality. The multiple XOR theorem implies that there exist polynomial size circuits $\{C_{i,m} : m \geq i \geq 1\}$ such that the following property holds for infinitely many m,

$$(\forall i < m)\left(Pr[u \in X_m : B_{i,m}(u) = C_{i,m}(u)] \geq 1 - \frac{1}{mP(m)}\right).$$

It will be shown that the circuit

$$C_m(u) = < C_{1,m}(u), \ldots, C_{m,m}(u) >$$

strongly P-inverts the function f. Indeed, for each $m \in M'$ it can be shown that

$$Pr\left[u \in X_m : C_m(u) \neq f_m^{-1}(u)\right] \leq$$

$$\sum_{i=1}^{m} Pr\left[u \in X_m : C_{i,m}(u) \neq B_{i,m}(u)\right] \leq \sum_{i=1}^{m} \frac{1}{mP(m)} = \frac{1}{P(m)}.$$

It follows that for all $m \in M'$,

$$Pr\left[u \in X_m : C_m(u) = f_m^{-1}(u)\right] \geq 1 - \frac{1}{P(m)},$$

which is a contradiction. This completes the proof of the theorem.•

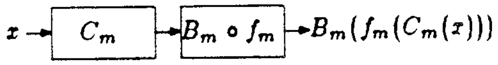
Recall that in theorem 5.12 the passage from a strong one way function to a friendship function and its associated unapproximable predicate was accomplished by passing to a space of higher dimension, namely $(X_m)^m$. However, the answer to the following question seems to be open.

Question: Is every weak or strong one way function the friendship function of an unapproximable predicate?

EXERCISES

1: Show that the predicate B defined in the course of the proof of (1) ⇒ (3) of theorem 5.12 satisfies RGH.

2: The circuit C' considered in the course of the proof of theorem 5.11 was defined nondeterministically. Show that if the function $f = \{f_m : m \geq 0\}$ is a friendship function for the unapproximable predicate $B = \{B_m : m \geq 0\}$ and satisfies $Pr[C_m(f_m(x)) = x] \geq 1/2 + 1/P(m)$ for infinitely many m, then the deterministic circuit pictured below satisfies the conclusion of the claim in theorem 5.11:



5.7 BIBLIOGRAPHICAL REMARKS

Most of the results of this section are the work of A. Yao and are outlined in cite [Y1]. The formal proof of the XOR lemma is partly based on [Y2]. Theorem 5.1 can be found in [BM]. Additional information on the security of public key cryptosystems as well as a different approach to the proof of the xor lemma can also be found in the unpublished [Rac].

Bibliography

[AMM] Adleman, L., Manders, K. and Miller, G., On Taking Roots in Finite Fields, 20-th IEEE FOCS, Vol. 20, 1977, pp. 175-178.

[AB] Ajtai, M., Ben-Or, M., A Theorem on Probabilistic Constant Depth Computations, 26th IEEE Symposium on Foundations of Computer Science, pp. 471 - 474, IEEE, 1984.

[An] Angluin, Dana, Lecture Notes on the Complexity of Some Problems in Number Theory, Yale University, Department of Computer Science, August,1982, 243.

[AL] Angluin, Dana and Lichtenstein David, Provable Security of Cryptosystems: a Survey, Yale University, Department of Computer Science, August,1982.

[Ber] Berlekamp, E. R., Factoring Polynomials over Large Finite Fields, Mathematics of Computation, Vol. 24, 1970, pp. 713-735.

[Bet] Beth, T., Introduction to Cryptology, in: Arbeitstagung über Kryptographie in Burg Feuerstein, T. Beth editor, pp. 1 - 28, Springer Verlag Lecture Notes in Computer Science, Vol. 149, 1983.

[Br] Brown, G. W., Monte Carlo Methods, in Modern Mathematics for the Engineer, Edwin F. Beckenbach editor, pp. 279 - 303, McGraw-Hill, 1956.

[BBS] Blum, L., Blum, M., and Shub, M., A Simple Secure Pseudo-Random Generator, IEEE CRYPTO 82, 1982.

[BM] Blum, M. and Micali, S., How to Generate Cryptographically Strong Sequences of Pseudo-Random Bits, in 23rd IEEE Symposium on Foundations of Computer Science, pp. 112 - 117, IEEE, 1982.

[C] Carmichael, R. D., On Composite Numbers Which Satisfy the Fermat Congruence, American Mathematical Monthly, Vol. 19, 1912, pp. 22-27.

[DDDHL] Demillo, R., Davida, G., Dobking, D., Harrison, A., and Lipton, R., Applied Cryptology, Cryptographic Protocols, and Computer Security Models, Proceedings of Symposia in Applied Mathematics, Vol. 29, American Mathematical Society, 1983.

[D] Denning, D., Cryptography and Data Security, Addison Wesley 1983.

[DH] Diffie, W., and Hellman, M., New Directions in Cryptography, IEEE Transactions on Information Theory, IT 22, pp. 644 - 654, 1976.

[EL] Eier, R. and Lagger H., Trapdoors in Knapsack Cryptosystems, in: Arbeitstagung über Kryptographie in Burg Feuerstein, T. Beth editor, pp. 316 - 322, Springer Verlag Lecture Notes in Computer Science, Vol. 149, 1983.

[E] Ellison, W. J., Les Nombres Premiers, Hermann, Paris, 1975.

[F] Feller, W., An Introduction to Probability Theory and its Applications , Wiley, 1966, New York.

[Ga] Gauss, C. F., Disquisitiones Arithmeticae, Yale University Press, 1966.

[Ge] Gesternhaber, M., The 152-nd Proof of the Law of Quadratic Reciprocity, American Mathematical Monthly, Vol.70, 1963, pp. 397-398.

[Gn] Gnedenko, B. V., The Theory of Probability, Mir Publishers, 1976, Fifth Printing 1982, Moscow.

[GK] Gnedenko, B. V., and Khinchin, A. Ya., An Elementary Introduction to the Theory of Probability, Dover Publications, 1962, New York.

[GGM] Goldreich, O., Goldwasser, S. and Micali, S., How to Construct Random Functions, to appear.

[GM] Goldwasser, S. and Micali, S., Probabilistic Encryption and How to Play Mental Poker Keeping Secret All Partial Inforation, 14th STOC, pp. 365 - 377, 1982.

[GMT] Goldwasser, S., Micali, S., and Tong, P., Why and How to Establish a Private Code on a Public Network, In 23rd IEEE Symposium on Foundations of Computer Science, pp. 134 - 144, IEEE, 1982.

[Hal] Halton, J. H., A Retrospective and Prospective Survey of the Monte Carlo Method, SIAM Review, pp. 1-63, Vol. 12, No. 1, January 1970.

[Has] Hasse, H., Vorlesungen über Zahlentheorie, Springer Verlag, zweite Auflage, 1964.

[Hoo] Hooley, C., On Artin's Conjecture, Journal für die reine und angewandte Mathematik, pp. 208 - 220, Band 225, 1967.

[Hou] Householder, A. S.,Editor, Monte Carlo Method, U.S. Department of Commerce, National Bureau of Standards, Applied Mathematics Series, Volume 12, 1951, Washington D.C.

[KP] Koch H. and Pieper H., Zahlentheorie, VEB Deutscher Verlag der Wissenschaften, 1976, Berlin.

[Kol] Kolmogorov, A. N., The Theory of Probability, In: Mathematics: its content, methods and meaning, Aleksandrov, Kolmogorov and Lavrent'ev eds., MIT press, 1963.

[Kon] Konnheim, A., Cryptography: A Primer, John Wiley and Sons, 1981.

[Kn] Knuth, D. E., The Art of Computer Programming: Seminumerical Algorithms, Addison-Wesley, Vol. II, 1981, Reading Mass.

[Kr] Kraetzel, E., Zahlentheorie, VEB Deutscher Verlag der Wiss., 1981, Berlin.

[La] Landau, E., Handbuch der Lehre von der Verteilung der Primzahlen, Band 1, 1953, Chelsea, New York.

[Lem] Lempel, A., Cryptology in Transition, Computing Survey 11, 1979, pp. 285 - 303.

[Len] Lenstra, H. W., Integer Programming with a Fixed Number of Variables, University of Amsterdam, Department of Mathematics TR 81-03, April, 1981.

[Lev] LeVeque, W., Fundamentals of Number Theory, Addison-Wesley, 1977, Reading Mass.

[Ma] Mardzanisvili, K. K. and Postnikov, A. B., Prime Numbers, In: Mathematics: its content, methods and meaning, Aleksandrov, Kolmogorov and Lavrent'ev eds., MIT press, 1963.

[MH] Merkle, R, and Hellman M., Hiding Information and Signatures in Trapdoor Knapsacks, IEEE Transactions on Information Theory, IT 24-5, Sept. 1978.

[Mi] Mignotte, M., How to Share a Secret, in: Arbeitstagung über Kryptographie in Burg Feuerstein, T. Beth editor, pp. 371 - 375, Springer Verlag Lecture Notes in Computer Science, Vol. 149, 1983.

[N] Niederreiter, H., Quasi-Monte Carlo Methods and Pseudo-Random Numbers, Bulletin of American Mathematical Society, pp. 957 - 1042, Vol. 84, 1978.

[NZ] Niven, I. and Zuckerman, H. S., An Introduction to the Theory of Numbers, John Wiley and Sons, 1960, New York.

[Pe] Pekelis, V., Key to the Cipher, in: Cybernetics A to Z, pp. 169-174, 1974, Mir Publishers, Moscow.

[Pi] Pieper, H., Variationen über ein zahlentheorisches Thema von C. F. Gauss, VEB Deutscher Verlag der Wissenschaften, 1978, Berlin.

[Pl] Plumstead, J., Inferring a Sequence Generated by a Linear Congruence, in 23rd IEEE Symposium on Foundaticns of Computer Science, pp. 153 - 159, IEEE, 1982.

[PH] Pohlig, S. C., Hellman, M. E., An Improved Algorithm for Computing Logarithms over $GF(p)$ and its Cryptographic Significance, IEEE Transactions on Information Theory, Vol. IT-24, 1978, pp. 106-110.

[Prac] Prachar, K., Primzahlverteilung, 1957, Springer Verlag, Heidelberg.

[Prat] Pratt, V., Every Prime has a succinct certificate, SIAM Journal of Computing, pp.214 - 220, 1975.

[Rab] Rabin, M. O., Digitalized Signatures and Public Key Functions as Intractable as factorization, MIT Laboratory for Computer Science, January, 1979, 212.

[Rac] Rackoff, C., Lecture Notes on Cryptographic Protocolls, University of Toronto, 1984.

[Re1] Rényi, A., Wahrscheinlichkeitsrechnung mit einem Anhang über die Informationstheorie , VEB Deutscher Verlag der Wissenschaften, 1962, Berlin.

[Re2] Rényi, A., Foundations of Probability Theory, Holden Day, 1970, San Francisco.

[RSA] Rivest, R., Shamir, A., and Adelman, L., A Method for Obtaining digital Signatures and Public Key Cryptosystems, Comm. ACM, Vol. 21, pp. 120 - 126, 1978.

[Ro] Ross, Sheldon M., Introduction to Probability Models, Academic Press, 1980, New York.

[SS] Sattler, J., and Scnorr, C. P., Ein Effizienzvergleich der Faktorisierungsverfahren von Morrison-Brillhart und Schroeppel, in: Arbeitstagung über Kryptographie in Burg Feuerstein, T. Beth editor, pp. 331 - 351, Springer Verlag Lecture Notes in Computer Science, Vol. 149, 1983.

[Scha] Schanks, D., Solved and Unsolved Problems in Number Theory, Chelsea Publ. Co., Second Edition, 1978, New York.

[Schn] Scnorr, C. P., Is the RSA Scheme Safe?, in : Arbeitstagung über Kryptographie in Burg Feuerstein, T. Beth editor, pp. 325 - 329, Springer Verlag Lecture Notes in Computer Science, Vol. 149, 1983.

[Sham1] Shamir, A., A Polynomial Time Algorithm for Breaking the Basic Merkle-Hellman Cryptosystem, 23rd IEEE Symposium on Foundations of Computer Science, 1982, pp. 145 - 152.

[Sham2] Shamir, A., How to Share a Secret, Com. ACM, Vol 22, No. 11, pp. 612-613, Nov. 1979.

[Shan1] Shannon, C.E., Communication Theory of Secrecy Systems, Bell System Technical Journal, 1949, Vol. 28, pp. 656 - 715.

[Shr] Shreider, Yu. A., The Monte Carlo Method, Pergamom Press, 1966.

[So] Sobol, I. M., The Monte Carlo Method, University of Chicago Press, 2nd edition, 1974.

[Vi] Vinogradov, I. M., Elements of Number Theory, Dover Publications, Inc., 1954, New York.

[W] Weil, A., Number Theory for Beginners, with the Collaboration of M. Rosenlicht, Springer Verlag, 1979, Heidelberg.

[Y1] Yao, A., C., Theory and Applications of Trapdoors functions, in 23rd IEEE Symposium on Foundations of Computer Science, pp. 80-91, IEEE, 1982.

[Y2] Yao, A. C., Lectures on the XOR Theorem, Handwritten unpublished notes of four lectures delivered in Spring 82, Yale University.

[Z] Zagier, D., Die ersten 50 Millionen Primzahlen, in: Lebendige Zahlen, Fünf Exkursionen, Mathematische Miniaturen, by: Borho, W., Jantzen, J. C., Kraft H., Rohlfs, J. and Zagier, D., 1, pp. 39 - 73, Birkhäuser Verlag, 1981.