IEICE TRANS. FUNDAMENTALS, VOL.E92–A, NO.1 JANUARY 2009

PAPER *Special Section on Cryptography and Information Security*

A New Randomness Test Based on Linear Complexity Profile

Kenji HAMANO[†a)], ***Member***, **Fumio SATO**[††b)], ***Nonmember***, ***and*** **Hirosuke YAMAMOTO**[†c)], ***Fellow***

SUMMARY Linear complexity can be used to detect predictable nonrandom sequences, and hence it is included in the NIST randomness test suite. But, as shown in this paper, the NIST test suite cannot detect nonrandom sequences that are generated, for instance, by concatenating two different M-sequences with low linear complexity. This defect comes from the fact that the NIST linear complexity test uses deviation from the ideal value only in the last part of the whole linear complexity profile. In this paper, a new faithful linear complexity test is proposed, which uses deviations in all parts of the linear complexity profile and hence can detect even the above nonrandom sequences. An efficient formula is derived to compute the exact area distribution needed for the proposed test. Furthermore, a simple procedure is given to compute the proposed test statistic from linear complexity profile, which requires only $O(M)$ time complexity for a sequence of length M.

key words: *randomness test, linear complexity profile, NIST SP800-22*

1. Introduction

Random sequences play an essential role in secure cryptographic systems. But if pseudo-random sequences used in such systems show an evidence for nonrandomness, it might give adversaries an important clue to attack the systems.

The National Institute of Standards and Technology (NIST) has released NIST SP800-22 [12] in 2001, which is the standard test suite for randomness in the field of data security. However, some problems have been found in the NIST test suite. It was reported in [2], [3], [8], [10] that the DFT (spectral) test and the Lempel-Ziv compression test included in the NIST test suite need to be corrected. Furthermore, it was found in [9] that the recommended input size of the approximate entropy test should also be modified. Hence, the NIST updated some values of parameters for the DFT test and removed the Lempel-Ziv compression test from the NIST test suite in 2004. Okutomi et al. [13] evaluated the randomness of sequences generated by DES and SHA-1 based on the NIST test suite, and they showed that the overlapping template matching test and the Maurer's "Universal Statistical" test in the NIST test suite did not follow the theoretical binomial distribution, when DES or SHA-1 is used as a pseudo-random number generator. In the case of the overlapping template matching test, the problem was caused from inaccurate probabilities derived for the occurrence of the template in the NIST test suite [5]. Moreover, Kaneda et al. [7] reported that revising the Maurer's "Universal Statistical" test based on the model proposed by Coron [1], the empirical distribution of pass rate follows the theoretical binomial distribution. Recently, it is also reported in [6], [14] that the probabilities used in the test for the longest run of ones in the NIST test suite need to be corrected.

This paper treats the linear complexity test in the NIST test suite. The linear complexity of a sequence is defined by the length of the shortest linear feedback shift-register (LFSR) that can generate the sequence. In general, complexity tests evaluate the difficulty of predicting postfix bits from prefix bits in a sequence. Since the Lempel-Ziv compression test, which is based on Lempel-Ziv complexity, was removed from the NIST test suite, the linear complexity test is the only one test that can check the prediction difficulty and hence, it is important. But the NIST linear complexity test uses deviation from the ideal value only in the last part of the whole linear complexity profile and does not use deviations in all parts of the linear complexity profile. So, it cannot detect a lower linear complexity in all parts other than the last part of the whole linear complexity profile. Actually, in this paper, we will show that a sequence generated by concatenating two M-sequences with lower linear complexity can pass all the NIST tests including the linear complexity test. In order to improve this defect, we will propose a new faithful linear complexity test, which is based on the linear complexity profile, i.e. all the linear complexities of prefix sequences of a given sequence. This new linear complexity test can detect the above nonrandom sequences which can pass the NIST test suite.

In Sect. 2, we review the NIST linear complexity test. In Sect. 3, we derive the detailed property of the linear complexity profile. In Sect. 4, we give a recursive formula to calculate the exact distribution used in the linear complexity profile, and we propose a new faithful linear complexity test in Sect. 5. Furthermore, in Sect. 6, we show an example of a nonrandom sequence that can pass all the NIST tests, but cannot pass the new linear complexity test.

2. NIST Linear Complexity Test

Let H_0 be the hypothesis that a given binary sequence $\varepsilon =$

Manuscript received March 19, 2008.
Manuscript revised July 28, 2008.

[†]The authors are with the Department of Complexity Science and Engineering, the University of Tokyo, Kashiwa-shi, 277-8561 Japan.
[††]The author is with the Technical Research & Development Institute, Ministry of Defense, Tokyo, 154-8511 Japan.
a) E-mail: hamano@it.k.u-tokyo.ac.jp
b) E-mail: satoh@cs.trdi.mod.go.jp
c) E-mail: Hirosuke@ieee.org

DOI: 10.1587/transfun.E92.A.166

Table 1 Breakdown of the 188 statistical tests.

Test Name	#*P*-value	Test ID	Parameter
frequency	1	1	-
block-frequency	1	2	128
cumulative-sums	2	3-4	-
runs	1	5	-
longest-run	1	6	-
rank	1	7	-
dft	1	8	-
nonperiodic-templates	148	9-156	9
overlapping-templates	1	157	9
universal	1	158	7, 1280
apen	1	159	10
random-excursions	8	160-167	-
random-excursions-variant	18	168-185	-
serial	2	186-187	16
linear-complexity	1	188	500

$\varepsilon_0\varepsilon_1\cdots\varepsilon_{n-1}$ of length n is an ideal random sequence satisfying $\Pr(\varepsilon_i = 0) = \Pr(\varepsilon_i = 1) = \frac{1}{2}$ and $\Pr(\varepsilon_i \mid \varepsilon_0, \ldots, \varepsilon_{i-1}) = \Pr(\varepsilon_i)$, $0 \le i \le n-1$. A statistical test of randomness is a procedure that can judge the acceptance or rejection of the hypothesis H_0 for a given sequence based on its statistical properties. The significance level α is the probability of rejecting H_0 when it is true. Let x be an observed value of a test statistic X obtained by applying a statistical test to a random sample, then the P-value of x is the probability such that X is larger than x in the case that H_0 is true. A small P-value gives an evidence that a given sequence is nonrandom. Hence, we treat that if P-value $< \alpha$, then the given sequence fails the statistical test and H_0 is rejected. Otherwise, the given sequence can be regarded as a random sequence and H_0 is accepted.

The NIST test suite currently consists of 15 core statistical tests that can be viewed as 188 statistical tests as shown in Table 1 [12], [16]. Default input parameters in the NIST test suite are also shown in Table 1. The procedure of the NIST linear complexity test is given as follows [12].

Procedure of NIST Linear Complexity Test

S1 Partition a given sequence of length n into N disjoint subsequences of length M, where $n = MN$.

S2 Compute the linear complexity $\mathcal{L}_k$ of the k-th subsequence for $k = 1, \ldots, N$ by using the Berlekamp-Massey algorithm [11].

S3 Calculate the value of μ by

$$\mu = \frac{M}{2} + \frac{9 + (-1)^{M+1}}{36} - \frac{\frac{M}{3} + \frac{2}{9}}{2^M}.$$

S4 Calculate the value of $T_k = (-1)^M(\mathcal{L}_k - \mu) + \frac{2}{9}$ for $k = 1, \ldots, N$.

S5 For $k = 1$ to N, update $\nu_0, \ldots, \nu_6$ based on the value of T_k as follows:

$T_k \le -2.5$	Increment ν_0 by one
$-2.5 < T_k \le -1.5$	Increment ν_1 by one
$-1.5 < T_k \le -0.5$	Increment ν_2 by one
$-0.5 < T_k \le 0.5$	Increment ν_3 by one
$0.5 < T_k \le 1.5$	Increment ν_4 by one
$1.5 < T_k \le 2.5$	Increment ν_5 by one
$T_k > 2.5$	Increment ν_6 by one

S6 Calculate the test statistic $\chi^2 = \sum_{i=0}^{6} \frac{(\nu_i - N\pi_i)^2}{N\pi_i}$, where $\pi_0 = 0.01047, \pi_1 = 0.03125, \pi_2 = 0.125, \pi_3 = 0.5, \pi_4 = 0.25, \pi_5 = 0.0625, \pi_6 = 0.02078$.

S7 Calculate a P-value on the basis of the fact that the distribution of the test statistic χ^2 asymptotically follows a χ^2 distribution with six degrees of freedom under H_0.

S8 If P-value $< \alpha$, reject H_0.

The P-value of a statistical test distributes uniformly in $[0, 1]$ if H_0 is true. So, in order to evaluate test results when s P-values are obtained by applying the statistical test to s sequences, we evaluate $\mathcal{P}$ and $\mathcal{U}$: the former is defined by the ratio of sequences that pass the statistical test at a significance level α and the latter is defined by the P-value of a χ^2 statistic given by $\chi^2 = \left(\sum_{i=1}^{10}\left(f_i - \frac{s}{10}\right)^2\right) / \frac{s}{10}$, where f_i is the number of P-values included in sub-interval $C_i = [0.1(i-1), 0.1i)$, $i = 1, 2, \ldots, 10$, to check the uniformity of P-values. If $\mathcal{P}$ falls outside of the range

$$\left[\hat{p} - 3\sqrt{\frac{\hat{p}(1-\hat{p})}{s}},\ \hat{p} + 3\sqrt{\frac{\hat{p}(1-\hat{p})}{s}}\right], \tag{1}$$

where $\hat{p} = 1 - \alpha$, then we can consider that the sequences are nonrandom. If $\mathcal{U} \ge 0.0001$, it is treated in the NIST test that the P-values distribute uniformly. However, for $\alpha = 0.01$ and $s = 10^3$, which are the most commonly used values and also used in this paper, the probability of type I error is relatively large because $\Pr\left\{\mathcal{P} \le \hat{p} - 3\sqrt{\frac{\hat{p}(1-\hat{p})}{s}}\right\} \approx 0.0033$, and hence, the probability that one or more $\mathcal{P}$s become below the threshold $\hat{p} - 3\sqrt{\frac{\hat{p}(1-\hat{p})}{s}}$ is given by $\sum_{i=1}^{188} {}_{188}\mathrm{C}_i (1 - 0.0033)^{188-i} 0.0033^i = 0.463$ in the case of all the 188 statistical tests.

In the NIST report [16] on the evaluation of AES finalists as random number generators, the P-value of $\mathcal{P}$ is used rather than the range given by Eq. (1). If the P-value of $\mathcal{P}$ is not less than 0.0001, the sequences are considered to be random. The minimum acceptable criterion of the test pass ratio $\mathcal{P}$ is given by 0.976 for $\alpha = 0.01$ and $s = 10^3$. In this paper, we also use the above criteria for $\mathcal{P}$ and $\mathcal{U}$ in the same way as [16].

3. Mathematical Background

For a given random binary sequence $\varepsilon^M = \varepsilon_0\varepsilon_1\cdots\varepsilon_{M-1}$ of length M, let L_i be the linear complexity of $\varepsilon^i = \varepsilon_0\varepsilon_1\cdots\varepsilon_{i-1}$, which is the first i bits of ε^M. For simplicity, we assume that $L_0 = 0$. The linear complexity profile of ε^M is a line graph connecting the following points:

$$\{(0, L_0), (1, L_0), (1, L_1), (2, L_1), (2, L_2), (3, L_2), (3, L_3), \ldots, (M, L_{M-1}), (M, L_M)\}.$$

A typical linear complexity profile is shown in Fig. 1, where line $l: y = \frac{1}{2}x$ is also drawn for reference.

In order to derive two theorems used in our test, we consider ε^M that satisfies the following two conditions.

Condition 1: M is even.

Condition 2: Point (M, L_M) is on the line $l: y = \frac{1}{2}x$.

Under these conditions, the linear complexity profile and the line l construct many pairs of congruent triangles as shown in Figs. 1 and 2. To describe each pair of congruent triangles (PCT), we use the following notations.

Definition 1: Let T_m be a PCT which has a horizontal edge of length m and let $a(T_m)$ be the area of T_m.

Definition 2: Let $a^{(M)}$ be the total area of all PCTs constructed by the linear complexity profile of ε^M and the line l, i.e. $a^{(M)}$ is the sum of all $a(T_m)$.

The linear complexity L_i and PCT T_m satisfy the following properties and theorems.

Property 1: The equation $L_i + L_{i-1} = i$ holds if $L_i > L_{i-1}$.

Proof See Theorem 2 of [11]. □

Property 2: All the triangles constructed by the linear complexity profile and the line l are similar.

Proof All the triangles are right-angled triangles and have the same acute angles $\theta = \arctan \frac{1}{2}$. □

Property 3: Two adjacent triangles shown in Fig. 2 are congruent.

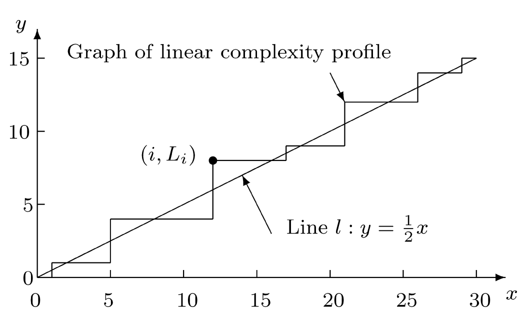

Fig. 1 Linear complexity profile and line l: $y = x/2$.

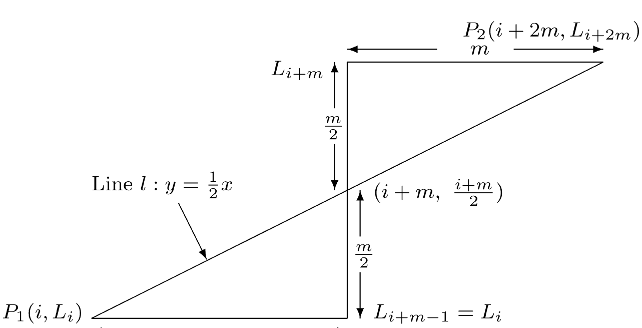

Fig. 2 A pair of congruent triangles (PCT).

Proof It is clear from Properties 1 and 2. □

Property 4: The horizontal distance from point P_1 to point P_2 shown in Fig. 2 is even.

Proof It is clear from Property 3. □

Property 5: If ε^M satisfies Conditions 1 and 2, the x-coordinates of points P_1 and P_2 of every PCT shown in Fig. 1 are even.

Proof It is clear from Property 4. □

The following theorems can be derived from the above properties.

Theorem 3.1: When (i, L_i) is on the line l, a PCT T_m shown in Fig. 2 occurs with probability $\frac{1}{2^m}$.

Proof For all $j \geq 0$, when $L_j \geq \frac{j}{2}+1$, the equation $L_{j+1} = L_j$ holds with probability 1. Otherwise the equation $L_{j+1} = L_j$ holds with probability $\frac{1}{2}$ and the equation $L_{j+1} = j+1-L_j$ holds with probability $\frac{1}{2}$ [11]. The occurrence of T_m means that the linear complexity keeps the same value during m bits. Therefore, $\Pr(T_m) = \frac{1}{2^m}$. □

Theorem 3.2: Let $a^{(M)}$ be the total area of PCTs that are constructed by ε^M satisfying Conditions 1 and 2, and the line l. Then, $a^{(M)}$ is given by

$$a^{(M)} = \sum_{j=1}^{M} \left| \frac{j}{2} - L_j \right|. \tag{2}$$

Proof The area of a PCT is given by the sum of rectangles whose width is one as shown in Fig. 3 and the area of each rectangle is equal to the vertical distance between the point (j, L_j) and the point $\left(j, \frac{j}{2}\right)$ on the line l. □

It is worth noting from Eq. (2) that the time complexity of computing $a^{(M)}$ from the linear complexity profile is $O(M)$. Note that $a^{(M)}$ becomes greater as the linear complexity profile deviates from the line l. Therefore, the large value of $a^{(M)}$ implies that H_0 is much unreliable.

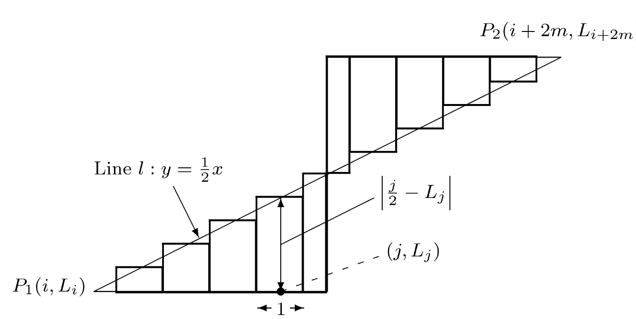

Fig. 3 Area calculation of a pair of congruent triangles (PCT).

4. Recursive Calculation for the Exact Area Distribution

We first note that for even i, $2 \le i \le M$,

$$\Pr\{\text{Point } (i, L_i) \text{ is on the line } l\} = \frac{1}{2}. \tag{3}$$

A brief proof of Eq. (3) is given in Appendix B for self-completeness although it is shown in [15]. In the same way as Definition 2, let $a^{(i)}$ denote the total area of all PCTs for ε^i such that i is even and (i, L_i) is on the line l. Furthermore, let A_i and F_i be the random variable of $a^{(i)}$ and its probability distribution, respectively. Then, since A_i takes finite discrete values, F_i can be represented by a finite set of pairs $[a^{(i)}, \Pr(a^{(i)})]$, namely

$$F_i = \{[a_1^{(i)}, \Pr(a_1^{(i)})],\ [a_2^{(i)}, \Pr(a_2^{(i)})],\ \ldots\}.$$

Some examples of F_i are shown in Appendix A. Since Eq. (3) holds for even $i > 0$, we have that

$$\sum_{a^{(i)}} \Pr(a^{(i)}) = \frac{1}{2},$$

where the summation takes all the possible values of $a^{(i)}$. Hence, doubling all the probabilities in F_i yields the conditional probability distribution under the condition that i is even and (i, L_i) is on the line l.

For every ε^M, we can divide the set of PCTs into the last PCT and the remaining PCTs as illustrated in Fig. 4. Then, it is possible for every $T_{\frac{M}{2}-k}$, $0 \le k \le \frac{M}{2} - 1$, to become the last PCT of ε^M if ε^M is randomly generated. The discrete distribution of the remaining PCTs is F_{2k} when the last PCT is $T_{\frac{M}{2}-k}$. Therefore, the discrete distribution F_M can be represented by

$$F_M = \bigcup_{k=0}^{\frac{M}{2}-1} \Big\{[a_1^{(2k)} + a(T_{\frac{M}{2}-k}), \Pr(a_1^{(2k)}) \times \Pr(T_{\frac{M}{2}-k})], \\ [a_2^{(2k)} + a(T_{\frac{M}{2}-k}), \Pr(a_2^{(2k)}) \times \Pr(T_{\frac{M}{2}-k})],\ \cdots\Big\}, \tag{4}$$

where $\bigcup$ represents the union of sets.

In order to represent F_M recursively, we introduce an operator $\circ$ as follows:

$$[x_1, x_2] \circ [y_1, y_2] = [x_1 + y_1, x_2 \times y_2],$$
$$\{[x_1, x_2],\ [x_3, x_4],\ \ldots\} \circ [y_1, y_2] \\ = \{[x_1, x_2] \circ [y_1, y_2],\ [x_3, x_4] \circ [y_1, y_2],\ \ldots\}.$$

Then, F_M can be represented by

$$F_M = \bigcup_{k=0}^{\frac{M}{2}-1} \Big\{[a_1^{(2k)}, \Pr(a_1^{(2k)})],\ [a_2^{(2k)}, \Pr(a_2^{(2k)})],\ \cdots\Big\} \\ \circ [a(T_{\frac{M}{2}-k}), \Pr(T_{\frac{M}{2}-k})] \\ = \bigcup_{k=0}^{\frac{M}{2}-1} F_{2k} \circ [a(T_{\frac{M}{2}-k}), \Pr(T_{\frac{M}{2}-k})]. \tag{5}$$

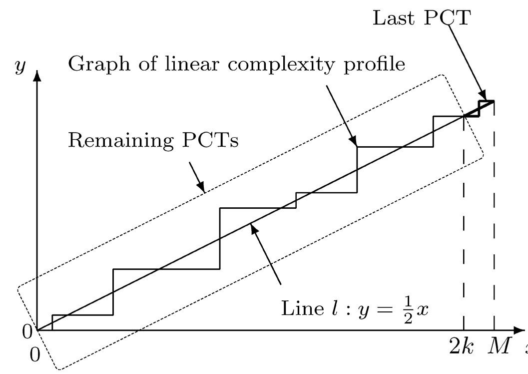

Fig. 4 Last PCT and remaining PCTs.

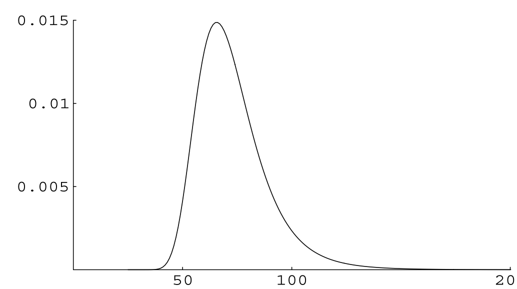

Fig. 5 Distribution of A_{100}.

Table 2 Statistics of the conditional probability distribution of A_M.

M	50	100	150	200	500
expectation	35.5	73	111	148	373
mode	30.5	66	102.5	139	363
median	33	69.5	107	144.5	368.5
variance	120	270	12630	22474	140594
upper 5%	55.5	103	147.5	191	441
upper 1%	72.5	126	173.5	220	480
upper 0.1%	103.5	164	215.5	264	534

Therefore, assuming $F_0 = \{[0, 1]\}$ for simplicity, we can calculate F_M recursively by Eq. (5). See examples shown in Appendix A.

Equation (5) means that the calculation of F_M does not require exponential time in M. Figure 5 shows F_{100}, i.e. the distribution of A_{100}, obtained by Eq. (5). Some statistics of the conditional probability distribution of A_M including the upper percentage points are shown for $M = 50, 100, 150, 200, 500$ in Table 2.

5. New Faithful Linear Complexity Test

The procedure of a new faithful randomness test based on A_M is given as follows.

Procedure of the New Linear Complexity Test

S1 Set the significance level α to 1%.

S2 Partition given sequence ε of length n into N disjoint subsequences of even length M, say $\varepsilon = \varepsilon_1^M \varepsilon_2^M \cdots \varepsilon_N^M$, where $n = MN$. Let T be the upper 5% point of A_M given in Table 2.

S3 Compute $a_k^{(M)}$, which is the observed value of A_M for each ε_k^M, $k = 1, \ldots, N$, by using Eq. (2).

S4 Let $\mathcal{A}$ be the set $\left\{a_k^{(M)} \middle| \mathcal{L}_k = \frac{M}{2}, 1 \le k \le N.\right\}$, where $\mathcal{L}_k$ is the linear complexity of ε_k^M, and let N' be the cardinality of $\mathcal{A}$. (Note that N' is expected to be approximately $\frac{N}{2}$ when H_0 is true.)

S5 If $\left|\frac{N'}{N} - \frac{1}{2}\right| > \frac{3}{2\sqrt{N}}$, reject H_0 on the basis of 3σ method.

S6 If the inequality $N'\phi > 5$ does not hold for $\phi = \Pr(\mathrm{A}_M > T)$, increase N and return to S2.

S7 Calculate $p = \dfrac{\#\{a_k^{(M)} \mid a_k^{(M)} \in \mathcal{A},\ a_k^{(M)} > T\}}{N'}$.

S8 Calculate $z = \dfrac{p - \phi}{\sqrt{\frac{\phi(1-\phi)}{N'}}}$.

S9 Compute P-value $= \mathrm{erfc}\left(\frac{|z|}{\sqrt{2}}\right)$, where $\mathrm{erf}(x) = \frac{2}{\sqrt{\pi}} \int_x^\infty \exp(-u^2)\mathrm{d}u$.

For the upper 5% point of $\mathrm{A}_{\mathbf{500}}$, we have $T = 441$ and $\phi = 0.049600$. The value of ϕ is not exactly 5% because of the discreteness of A_M. Note that the output form of our test is the same as that of the NIST test suite because our test outputs a P-value in the same way as the NIST test suite.

6. Advantages of the New Linear Complexity Test

We compare our test with the NIST linear complexity test to show an advantage of our test.

Since NIST recommends that $n \ge 10^6$, $500 \le M \le 5000$, where n is the length of a sequence and M is the length of each subsequence, we made a comparison for the case of $n = 10^6$ and $M = 500$. Our test and the NIST linear complexity test were applied to sequences generated by SHA-1, which is known as a good random number generator. Test results are summarized in Table 3, where f_i is the number of P-values included in sub-interval $C_i = [0.1(i-1), 0.1i)$, $i = 1, 2, \ldots, 10$. As shown in Table 3, both tests concluded that the sequences appear to be random.

Now, let us show that the NIST linear complexity test has a problem arising from the fact that it checks only the last part of the whole linear complexity profile. Consider sequences ε' in which the first 244 bits of each subsequence are all "0"s, the 245th bit to the 488th bit of each subsequence are all "1"s, and the remaining 12 bits of each subsequence are random numbers generated by SHA-1. For each subsequence, the set of PCTs includes T_{244} corresponding to the first 488 bits. Test results are summarized in Table 4, which show that the NIST linear complexity test failed to reject H_0. However, our test rejected H_0 because the observed value of $\mathrm{A}_{\mathbf{500}}$ of each subsequence is greater than $a(T_{244}) = 29768$, and thereby "P-value < 0.01" always holds.

Although the NIST linear complexity test fails to detect the nonrandomness of ε', the frequency test in the NIST test suite can detect it. But there exist nonrandom sequences such that all the NIST tests fail to detect the nonrandomness but our test can detect it. Consider sequences ε'' in which the first $2 \times 89 + 30 = 208$ bits of each subsequence are generated by an M-sequence whose primitive polynomial is degree 89 polynomial of three terms, and the remaining $500 - 208 = 292$ bits of each subsequence are generated by another M-sequence whose primitive polynomial is degree 521 polynomial of 279 terms. For each subsequence, the set of PCTs includes a PCT which is larger than T_{30}. Our test can determine the nonrandomness of sequences ε'' because all the observed values of $\mathrm{A}_{\mathbf{500}}$ are larger than $a(T_{30}) = 450$, and thereby "P-value < 0.01" always holds. Figures 6 and 7 depict the test results of the NIST test suite in the case where default input parameters were used in the NIST test suite. For each statistical test in the NIST test suite, the calculated $\mathcal{P}$ and $\log_{10} \mathcal{U}$ are plotted in Figs. 6 and 7, respectively. From the figures, we can see that the NIST test suite

Table 3 Test results for sequences generated by SHA-1.

	NIST's test	Our test
f_1	94	92
f_2	97	110
f_3	98	107
f_4	92	98
f_5	106	105
f_6	104	100
f_7	109	101
f_8	110	106
f_9	100	84
f_{10}	90	97
$\mathcal{U}$	0.878618	0.794391
$\mathcal{P}$	0.9900	0.9880
Result	Pass	Pass

Table 4 Test results of the NIST linear complexity test and the new linear complexity test for ε'.

	NIST's test	Our test
f_1	99	1000
f_2	100	0
f_3	98	0
f_4	101	0
f_5	101	0
f_6	87	0
f_7	110	0
f_8	112	0
f_9	101	0
f_{10}	91	0
$\mathcal{U}$	0.832561	0.000000
$\mathcal{P}$	0.9910	0.0000
Result	Pass	Reject

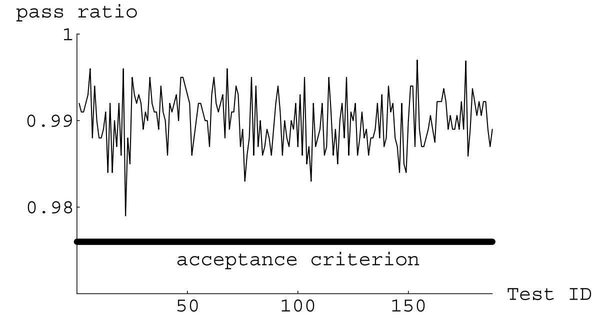

Fig. 6 $\mathcal{P}$s of the NIST test suite for ε''.

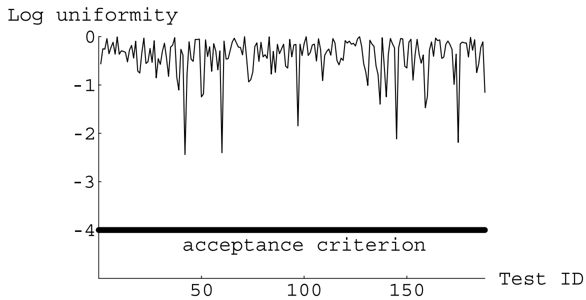

Fig. 7 $\mathcal{U}$s of the NIST test suite for ε''.

failed to reject H_0 because any $\mathcal{P}$s and $\mathcal{U}$s cannot become smaller than their acceptance criteria. It is conjectured that if sequences have similar properties to ε'', the same result is obtained.

7. Conclusions

We proposed a new randomness test based on the whole linear complexity profile. We also showed that any tests in the NIST test suite cannot detect the nonrandom sequences ε'' defined in Sect. 6, but the new linear complexity test proposed in this paper can reject the sequences ε''. These results come from the following fact. The NIST linear complexity test can detect deviations from the line $l : y = \frac{1}{2}x$ only in the last part of the whole linear complexity profile, but our new test uses deviations in all parts of the linear complexity profile, and hence can realize a faithful test.

We also derived an efficient formula to compute the exact area distribution needed for the new linear complexity test. Furthermore, we gave a simple procedure to compute the test statistic of the new test. Since the output form of the new test is the same as that of the NIST test suite, it can be easily added into the NIST test suite.

References

[1] J.S. Coron, "On the security of random sources," Public-Key Cryptography, vol.1560, LNCS, pp.29–42, Springer-Verlag, 1999.
[2] K. Hamano, F. Satoh, and M. Ishikawa, "Randomness test using discrete Fourier transform," Technical Report 6841, Technical Research and Development Institute, Japan Defense Agency, Sept. 2003.
[3] K. Hamano, "The distribution of the spectrum for the discrete Fourier transform test included in SP800-22," IEICE Trans. Fundamentals, vol.E88-A, no.1, pp.67–73, Jan. 2005.
[4] K. Hamano, "Correction of overlapping template matching test included in NIST randomness test suite," Technical Report 6944, Technical Research and Development Institute, Japan Defense Agency, Oct. 2006.
[5] K. Hamano and T. Kaneko, "Correction of overlapping template matching test included in NIST randomness test suite," IEICE Trans. Fundamentals, vol.E90-A, no.9, pp.1788–1792, Sept. 2007.
[6] K. Hamano, "Correction of "test for the longest run of ones in a block" included in NIST randomness test suite," IEICE Technical Report, ISEC2007-3, May 2007.
[7] M. Kaneda, H. Okutomi, and K. Nakamura, "A study on Maurer's "Universal statistical" test included in NIST randomness test suite," Abstracts of the 2007 Symposium on Cryptography and Information Security, p.202, Jan. 2007.
[8] M. Kaneda, H. Okutomi, and K. Nakamura, "A study on discrete Fourier transform test included in NIST randomness test suite," IEICE Technical Report, ISEC2006-124, March 2007.
[9] Kaneko Lab., http://www.ipa.go.jp/security/enc/CRYPTREC/fy16/documents/rep_ID0211_000.pdf, Dec. 2004.
[10] S. Kim, K. Umeno, and A. Hasegawa, "On the NIST statistical test suite for randomness," IEICE Technical Report, ISEC2003-87, Dec. 2003.
[11] J.L. Massey, "Shift-register synthesis and BCH decoding," IEEE Trans. Inf. Theory, vol.IT-15, no.1, pp.122–127, Jan. 1969.
[12] National Institute of Standards and Technology, "A statistical test suite for random and pseudorandom number generators for cryptographic applications," NIST special publication 800-22, 2001.
[13] H. Okutomi, M. Kaneda, K. Yamaguchi, and K. Nakamura, "A study on the randomness evaluation method using NIST randomness test," Abstracts of the 2006 Symposium on Cryptography and Information Security, p.10, Jan. 2006.
[14] H. Okutomi, M. Kaneda, and K. Nakamura, "A study on the NIST randomness test — Especially evaluating "The test for the longest run of ones in a block"," Abstracts of the 2008 Symposium on Cryptography and Information Security, p.308, Jan. 2008.
[15] R.A. Rueppel, Analysis and Design of Stream Ciphers, Communications and Control Engineering, Springer-Verlag, Berlin, 1986.
[16] J. Soto, "Randomness testing of the AES candidate algorithms," NIST AES report, available at http://citeseer.ist.psu.edu/250886.html

Appendix A: Examples of Discrete Distribution F_i

Discrete distribution F_i can be calculated as follows.

$$\begin{aligned}
F_0 &= \{[0,1]\},\\
F_2 &= F_0 \circ \left[\frac{1}{2}, \frac{1}{2}\right] = \left\{\left[\frac{1}{2}, \frac{1}{2}\right]\right\},\\
F_4 &= F_2 \circ \left[\frac{1}{2}, \frac{1}{2}\right] \cup F_0 \circ \left[2, \frac{1}{4}\right]\\
&= \left\{\left[1, \frac{1}{4}\right]\right\} \cup \left\{\left[2, \frac{1}{4}\right]\right\} = \left\{\left[1, \frac{1}{4}\right], \left[2, \frac{1}{4}\right]\right\},\\
F_6 &= F_4 \circ \left[\frac{1}{2}, \frac{1}{2}\right] \cup F_2 \circ \left[2, \frac{1}{4}\right] \cup F_0 \circ \left[\frac{9}{2}, \frac{1}{8}\right]\\
&= \left\{\left[1, \frac{1}{4}\right], \left[2, \frac{1}{4}\right]\right\} \circ \left[\frac{1}{2}, \frac{1}{2}\right] \cup \left\{\left[\frac{1}{2}, \frac{1}{2}\right]\right\} \circ \left[2, \frac{1}{4}\right]\\
&\quad \cup \{[0,1]\} \circ \left[\frac{9}{2}, \frac{1}{8}\right]\\
&= \left\{\left[\frac{3}{2}, \frac{1}{8}\right], \left[\frac{5}{2}, \frac{1}{8}\right]\right\} \cup \left\{\left[\frac{5}{2}, \frac{1}{8}\right]\right\} \cup \left\{\left[\frac{9}{2}, \frac{1}{8}\right]\right\} \qquad \text{(A$\cdot$ 1)}\\
&= \left\{\left[\frac{3}{2}, \frac{1}{8}\right], \left[\frac{5}{2}, \frac{1}{4}\right], \left[\frac{9}{2}, \frac{1}{8}\right]\right\}. \qquad \text{(A$\cdot$ 2)}
\end{aligned}$$

The elements with the same first component $\frac{5}{2}$ in Eq. (A· 1) are merged into $\left[\frac{5}{2}, \frac{1}{4}\right]$ in Eq. (A· 2) by summing up the second components $\frac{1}{8}$ and $\frac{1}{8}$.

In a similar way, F_8 can be obtained as below.

$$F_8 = \left\{\left[2, \frac{1}{16}\right], \left[3, \frac{3}{16}\right], \left[4, \frac{1}{16}\right], \left[5, \frac{1}{8}\right], \left[8, \frac{1}{16}\right]\right\}.$$

Appendix B: Proof of Eq. (3)

Let $N_i(L)$ be the number of binary sequences of length i with linear complexity L. A recursive formula of $N_i(L)$ can be obtained from the proof of Theorem 3.1 as follows:

$$N_i(L) = \begin{cases} 2N_{i-1}(L) + N_{i-1}(i-L), & \frac{i}{2} < L \le i, \\ 2N_{i-1}(L), & L = \frac{i}{2}, \\ N_{i-1}(L), & 0 \le L < \frac{i}{2}, \end{cases}$$

where $N_1(0) = N_1(1) = 1$. Hence, we obtain that

$$N_i(L) = \begin{cases} 2^{\min\{2i-2L, 2L-1\}}, & 0 < L \le i, \\ 1, & L = 0. \end{cases}$$

Since $N_i(\frac{i}{2}) = 2^{i-1}$ and the number of sequences of length i is 2^i, the probability that point (i, L_i) of the linear complexity profile is on the line $l: \; y = \frac{1}{2}x$ is $\frac{1}{2}$ for even i.

Kenji Hamano received the B.E. and M.E. degrees in Mathematical Engineering from the University of Tokyo in 1999 and 2001, respectively. In 2001, he joined the Technical Research & Development Institute, Ministry of Defense. He is currently a doctoral student at the University of Tokyo. His research interests are randomness, statistics and cryptography.

Fumio Sato received the M.E. degree from the University of Tokyo, Japan, in 1993 and the Ph.D. degree from the Tokyo Institute of Technology, Japan, in 2000. He is currently a senior research scientist in the Technical Research & Development Institute, Ministry of Defense, Japan. His current research interest is information technology security evaluation.

Hirosuke Yamamoto was born in Wakayama, Japan, on November 15, 1952, He received the B.E. degree from Shizuoka University, in 1975 and the M.E. and Ph.D. degrees from the University of Tokyo, in 1977 and 1980, respectively, all in electrical engineering. In 1980, he joined Tokushima University. He was an Associate Professor at Tokushima University, the University of Electro-Communications, and the University of Tokyo, from 1983 to 1987, from 1987 to 1993, and from 1993 to 1999, respectively. Since 1999, he has been a Professor at the University of Tokyo. He was with the School of Engineering and the School of Information Science and Technology from 1993 to 1999 and from 1999 to 2004, respectively, and is currently with the School of Frontier Sciences in the University of Tokyo. In 1989 and 1990, he was a Visiting Scholar at the Information Systems Laboratory, Stanford University. His research interests are in Shannon theory, data compression algorithms, and cryptology. Dr. Yamamoto is a member of the IEEE and the SITA (Society of Information Theory and its Applications). He served as the Chair of the IEEE Information Theory Society Japan Chapter in 2002 and 2003, and the TPC (Technical Program Committee) Co-Chair of ISITA2004 (the 2004 International Symposium on Information Theory and its Applications). He is currently the President of the SITA, an Associate Editor for Shannon Theory, IEEE Transactions on Information Theory, and the TPC Chair of ISITA2008.