AN IMPROVED METHOD FOR IDENTIFICATION OF PATTERNS IN THE NON-OVERLAPPING TEMPLATE MATCHING TEST

Yuichi Takeda,† Mituaki Huzii,† Norio Watanabe† and Toshinari Kamakura†

ABSTRACT

We proposed a modified non-overlapping template matching test and a method for specifying a pattern that appears too many times, in Takeda et al. (2014). The method that we proposed was effective for the most part, but had some difficulties. Our new contribution in this paper is to propose an improved method of identifying a pattern that appears too many times, in the non-overlapping template matching test for resolving these difficulties and to show how this identification test works effectively by some simulation studies.

1. Introduction

We are interested in tests for randomness in the field of cryptography. A statistical test suite has been proposed in Rukhin et al. (2010), which has been used in the field of cryptography. The non-overlapping template matching test was introduced in the test suite and would be effective for finding a pattern that appears many times. Here a pattern means a string consisting of 0’s and 1’s with a certain length. As in Rukhin et al. (2010), we call a pattern that is used for carrying out the non-overlapping template matching test, a template. Our purpose is to specify a pattern which appears too many times. We deal with the case when we have no information on B (not even the size m of B). In general, the non-overlapping template matching test depends on the setting of the template. It is difficult to try the test for all possible lengths and patterns as templates. Therefore, it is important to set a template in the non-overlapping template matching test. However, no method for setting of the template is proposed in Rukhin et al. (2010). In Takeda et al. (2014), we proposed a test which is a modification of the non-overlapping template matching test, and a method for identifying a pattern which appears too many times.

Let \( \{Z_i\} \) be a binary sequence whose size is n. We assume that the null hypothesis is \( \{Z_i\} \) is a sequence of mutually independent random variables and \( Z_i \) takes 0 and 1 with probability 1/2. We set a short length \( m^* \), which, we suppose, is shorter than the length m of the pattern B which appears too many times. By considering each of all binary patterns of length \( m^* \) to be the template \( B^* \), we carry out our modification method of the non-overlapping template matching test. By this method, we can find the existence of a pattern which appears too many times. In addition, we need to specify the pattern B which appears too many times. Therefore we proposed an identification method of the pattern B in Takeda et al. (2014). Our method succeeded in identification for many cases. However, our method

*Center for Basic Education and Integrated Learning, Kanagawa Institute of Technology, 1030, Shimo-ogino, Atsugi, Kanagawa 243-0292, Japan E-mail: y-takeda@ctr.kanagawa-it.ac.jp
†Faculty of Science and Engineering, Chuo University, 1-13-27, Kasuga, Bunkyo, Tokyo 112-8551, Japan

Key words: Testing randomness; Non-overlapping template matching test; Kolmogorov-Smirnov test; Identification method; Cryptographic application; NIST 800-22
2. Proposed procedure of identifying the template (Takeda et al. (2014))

One of the important points in the non-overlapping template matching test proposed in Rukhin et al. (2010) is how to choose the template. No method is shown in Rukhin et al. (2010). In Takeda et al. (2014), we discussed the case when we do not have any information on the pattern $B$, which appears too many times, and its length $m$. Our method (Short-Bit-NO-test) proposed in Takeda et al. (2014) is as follows. Instead of choosing the pattern $B$ as the template, first we set a short length $m^*$ (a positive integer) about which we can expect $m^* < m$, take each $B^*$ of $2^{m^*}$ binary patterns as the template, count the number of occurrences of $B^*$ in the binary sequence of size $n$ by non-overlapping template matching method, and repeat this operation $N$ times in the same line as in Rukhin et al. (2010). We obtain numbers of times of size $N$ that $B^*$ occurs is the binary sequence of size $n$ and construct the frequency distribution function $F_N(k)$, $k = 0, 1, 2, \cdots$ of the numbers of times that $B^*$ occurs in the binary sequence of size $n$. Let $F^0(k)$, $k = 0, 1, 2, \cdots$ be the probability distribution function of the number of times that $B^*$ occurs in the binary sequence of size $n$ under the null hypothesis. We use the one-sided Kolmogorov-Smirnov test. Put
\[
D^-_N = \max_k \left( \max \left( F^0(k) - F_N(k) \right) , 0 \right)
\]
and use the evaluation $P(\sqrt{N}D^-_N \geq d) \leq \exp(-2d^2)$ when $N$ is sufficiently large, and construct our test statistic by using $\exp(-2d^2)$, when $d$ is the sample value of $\sqrt{N}D^-_N$.

In Takeda et al. (2014), we discussed two cases, Case (C-1) and Case (C-2). Case (C-1) is that we have to find the existence of the pattern $B$ which appears too many times, of a certain length in the binary sequence, but we do not need to identify the pattern $B$. Case (C-2) is that we need to identify the pattern $B$ which appears too many times.

In Case (C-2) we proposed a method for identifying the pattern $B$ in the sequence. Our method is the following:

In the first step, we construct the set $B^{1*}$. $B^{1*}$ is the set of templates, for which the null hypothesis is rejected in the Short-Bit-NO-Test. In the second step, by making the combination $B^*_{12} = B^*_1 \oplus B^*_2$ of two templates $B^*_1$ and $B^*_2$ included in $B^{1*}$, we take each $B^*_{12}$ as a template and carry out the above Short-Bit-NO-Test. Here a combination $B^*_{12} = B^*_1 \oplus B^*_2$ of two templates means the first $m^*$ bits of $B^*_{12}$ are the same as $B^*_1$ (or $B^*_2$), the last $m^*$ bits of $B^*_{12}$ are the same as $B^*_2$ (or $B^*_1$) and the length $m^*_{12}$ of $B^*_{12}$ satisfies $m^*_{12} \leq m^* + m^*$. In the second step, by using the templates $B^*_{12}$’s which give rejection of the null hypothesis, we carry out the Short-Bit NO-Test. The third, fourth, \cdots steps are repetitions of the same operations. We stop the operation when we do not have any template which gives rejection of the null hypothesis or the repetition comes to the pre-specified step. The pattern $B$ is the template for which the $p$-value is minimum in the last set.

We calculated success ratios of this identification method by simulation studies. In our simulation studies, we set $m^* = 3, n = 10^3, N = 10^3$ and the level of significance
Improved Non-overlapping Template Matching Test

\( \alpha = 0.05 \). We generate original binary sequences, each of which is supposed to satisfy the null hypothesis. We modify each of these sequences so that a pattern \( \tilde{B} \) of length \( \tilde{m} \) appears many times and construct modified binary sequences to use in our simulation studies in the following way. We want to find the pattern \( \tilde{B} \) as the pattern \( B \) which appears too many times. We call the pattern \( \tilde{B} \) the \textit{embedded pattern}. Let \( \tilde{B} \) be an embedded pattern and let \( \tilde{m} \) be its length in bits. We divide every original sequence of length \( nN \) by \( \tilde{m} \) and obtain \( \lfloor nN/\tilde{m} \rfloor \) segments. Among these segments, we choose randomly \( \Gamma \) segments and replace each of these \( \Gamma \) segments by \( \tilde{B} \). Thus we can have a modified binary sequence. We define \( \Gamma.ratio \) as \( \Gamma/\lfloor nN/\tilde{m} \rfloor \). For each case, we repeated our simulation study \( 10^4 \) times. We tried to see how often we could find \( \tilde{B} \) successfully. The results are shown in Takeda et al. (2014). In many cases, the success.ratio becomes large when the \( \Gamma.ratio \) becomes large. However, in the case of \( \tilde{B} = 000001 \), the success.ratio is small. In this paper, we try to improve these weak points of our identification method.

3. Improved identification method

Before improving our identification method, we investigate the reason why the success.ratio’s are low in the case of \( \tilde{B} = 000001 \). When a pattern \( B \) is expressed as \( B = b_1b_2 \cdots b_m \), we put \( C = b_ib_{i+1} \cdots b_{i+j} \), where \( i \) and \( j \) are positive integers and \( i + j \leq m \). We call the pattern \( C \) a component pattern of \( B \). Generally, the distribution of the number of occurrences of a template depends on both the embedded pattern \( \tilde{B} \) and its length \( \tilde{m} \). In particular, we are interested in the distribution of the number of occurrences of the template which is a component pattern of \( \tilde{B} \). We derived the frequency distribution functions of the number of occurrences of these templates by simulation studies.

In the case of \( \tilde{B} = 010001 \), we expect the Short-Bit-NO-test to reject the templates 010 and 001. The frequency distribution is shifted to the right when the \( \Gamma.ratio \) becomes large. In this case we would be able to reject the templates 010 and 001 and construct the template 010001 by combining 010 and 001.

Figures 1 and 2 show frequency distributions of the numbers of occurrences of templates \( B_1 = 000 \) and \( B_2 = 001 \), in the case of \( \tilde{B} = 000001 \). The x-axis expresses the number of occurrences of the template, and the y-axis expresses the frequency. In this case, we expect the Short-Bit-NO-test to reject the templates 000 and 001. We can see that the frequency distribution is shifted to the right in the case of the template 000 when the \( \Gamma.ratio \) becomes large. But the frequency distributions are almost the same in the case of the template 001. In this case we would not be able to reject the template 001 and to construct the template 010001 by combining 010 and 001.

From these results, in some cases because the frequency distributions of the numbers of the template that is a component pattern of \( \tilde{B} \), are not shifted to the right when the \( \Gamma.ratio \) becomes large, we will not be able to construct the embedded pattern \( \tilde{B} \) by combining two
templates that are rejected under the null hypothesis. Even if both templates $B_1$ and $B_2$, which are component patterns of $\tilde{B}$, are not rejected, we should use the embedded pattern as template. It is necessary to change the rule for constructing the embedded pattern $\tilde{B}$ by combining two templates. We compare the $p$-value as follows. Let $B_1$ and $B_2$ be two templates and let $B_3 = B_1 \oplus B_2$. If the $p$-value of the template $B_3$ is smaller than $p$-values of both templates $B_1$ and $B_2$, we use $B_3$ in the next step. From the above consideration, we revise the identification method we previously proposed. The improved identification method is as follows.
Improved Non-overlapping Template Matching Test

Fig. 3: Frequency distribution functions of the number of occurrences of $B^* = 001$, $\tilde{B} = 0000001 \{0\%\text{(line)} 3\%\text{(dash)} 5\%\text{(dot)}\}$

First step. Let $\alpha$ be the level of significance. We set a short length $m^*$, which is supposed to be $m^* < m$. Let $B^{1*} = \{B^{1*}(1), B^{1*}(2), \ldots, B^{1*}(2^{m^*})\}$ be the set of the all $m^*$-bit patterns. Considering each element of $B^{1*}$ to be the template $B^*$, we carry out the Short-Bit-NO-Test. Let $p^{1*}(t)$ be a $p$-value of the $B^{1*}(t)$ for all $m^*$-bit patterns ($1 \leq t \leq T_1$). If we do not have any $m^*$-bit pattern $B^*$ where $p^{1*}(t) < \alpha$, we accept the null hypothesis. Otherwise, we proceed to the Second step.

$\kappa$-th step ($\kappa \geq 2$). Let $B^{\kappa*} = B^{(\kappa-1)*} \oplus B^{(\kappa-1)*} = \{B^{\kappa*}(1), B^{\kappa*}(2), \ldots, B^{\kappa*}(T_{1\kappa})\}$, where $T_{1\kappa}$ is the number of elements of $B^{\kappa*}$. Considering each element of $B^{i*}$ to be the template, we carry out the Short-Bit-NO-Test. Let $p^{i*}(t)$ be a $p$-value of the any element of $B^{i*}$. Let $B^{1\kappa*}$ be the set of templates for which the null hypothesis is rejected in the $\kappa$-th step. For any $B^{i*}(t)$, there exist $B^{(\kappa-1)*}(u)$ and $B^{(\kappa-1)*}(v)$, $1 \leq u, v \leq T_{1\kappa-1}$, in $B^{(\kappa-1)*}$ such that $B^{i*}(t) = B^{(\kappa-1)*}(u) \oplus B^{(\kappa-1)*}(v)$. When $p^{1\kappa*}(t) < \min(p^{(\kappa-1)*}(u), p^{(\kappa-1)*}(v))$, we use the $B^{i\kappa*}(t)$ in the next step. Let $B^{2\kappa*}$ be the set of templates for which $p^{1\kappa*}(t) < \min(p^{(\kappa-1)*}(u), p^{(\kappa-1)*}(v))$. We put

$$B^{\kappa*} = B^{2\kappa*} \cup B^{3\kappa*} = \{B^{\kappa*}(t); 1 \leq t \leq T_{\kappa}\},$$

where $T_{\kappa}$ is the number of elements of $B^{\kappa*}$. Let $p^{\kappa*}(t)$ be a $p$-value of the $B^{\kappa*}(t)$ for $1 \leq t \leq T_{\kappa}$. If $B^{\kappa*}$ is empty, we stop this process.

When $B^{(\kappa-2)*}$ includes the embedded pattern $\tilde{B}$, $B^{(\kappa-1)*}$ may include templates whose lengths are a few bits longer than one of the embedded patterns $\tilde{B}$. And the embedded pattern $\tilde{B}$ is a component pattern of these templates. These templates may be rejected in the $(\kappa-1)$-th step. In this case, because we should choose $B^{(\kappa-2)*}$, we compare the $p$-values of the elements of $B^{(\kappa-2)*}$ and $B^{(\kappa-1)*}$. If

$$\min_{1 \leq t \leq T_{(\kappa-2)}} p^{(\kappa-2)*}(t) \leq \min_{1 \leq t \leq T_{(\kappa-1)}} p^{(\kappa-1)*}(t),$$

we conclude that the embedded pattern $\tilde{B}$ is included in the set of templates $B^{(\kappa-2)*}$. 
Otherwise, we conclude that the embedded pattern \( \tilde{B} \) is included in the set of templates \( B^{(\kappa-1)^*} \). If \( B^{\kappa^*} \) is not empty, we proceed to the \((\kappa+1)\)-th step.

We continue this process until the \( \kappa^0 \)-th step, where \( \kappa^0 \) is the pre-assigned value. When \( B^{(\kappa_0-1)^*} \) includes the embedded pattern \( \tilde{B} \), \( B^{\kappa_0^*} \) may include templates whose lengths are a few longer than one of the embedded patterns \( \tilde{B} \). And the embedded pattern \( \tilde{B} \) is a component pattern of these templates. These templates may be rejected in the \( \kappa_0 \)-th step. In this case, because we should choose \( B^{(\kappa_0-1)^*} \), we compare the \( p \)-values of the elements of \( B^{(\kappa_0-1)^*} \) and \( B^{\kappa_0^*} \). If

\[
\min_{1 \leq t \leq T(\kappa_0-1)^*} p^{(\kappa_0-1)^*(t)} \leq \min_{1 \leq t \leq T_{\kappa_0}} p^{\kappa_0^*(t)},
\]

we conclude that the embedded pattern \( \tilde{B} \) is included in the set of the templates \( B^{(\kappa_0-1)^*} \). Otherwise we conclude that the embedded pattern \( \tilde{B} \) is included in the set of the templates \( B^{\kappa_0^*} \). We can show our Specifying Template Procedure as the flowchart in the Appendix.

4. Simulation studies

We examined the effectiveness of our improved identification method by simulations. We constructed the modified binary sequence for examining the effectiveness of our improved method in the same way as in Takeda et al. (2014). In these simulation studies, we set the same condition of Takeda et al. (2014). We generated original binary sequences, each of which was supposed to satisfy the null hypothesis. We modified each of these sequences so that a pattern \( \tilde{B} \) of length \( \tilde{m} \) bits appeared many times and constructed modified binary sequences to use in our simulation studies. We divided every original sequence of length \( nN \) by \( \tilde{m} \) and obtained \( \lfloor nN/\tilde{m} \rfloor \) segments. Among those segments, we chose randomly \( \Gamma \) segments and replaced each of these \( \Gamma \) segments by \( \tilde{B} \). We defined the \( \Gamma \)-ratio as \( \Gamma/\lfloor nN/\tilde{m} \rfloor \). We tried to see how we could find \( \tilde{B} \) successfully. We set \( m^* = 3, n = 10^3, N = 10^3 \) and the level of significance \( \alpha = 0.05 \). For each case, we repeated \( 10^4 \) times the identification of the embedded pattern in simulation study. Because we intend to reduce the number of elements in the set, which includes the pattern we should identify, the \textit{success.ratio} was defined as

\[
\text{success.ratio} = \frac{\text{times that } \tilde{B} \text{ is included in the set of templates obtained finally}}{10^4}.
\]

Table 1 shows the \textit{success.ratio}'s.

| \( \tilde{m} \) | \( \tilde{B} \) | \textit{success.ratio} |
|---|---|---|
| | \( 0.01 \) | \( 0.03 \) | \( 0.05 \) | \( 0.06 \) | \( 0.07 \) | \( 0.08 \) | \( 0.09 \) | \( 0.10 \) |
| 4 | 00001 | 0.9943 | 1.0000 | * | * | * | * | * | * |
| | 0101 | 0.9925 | 1.0000 | * | * | * | * | * | * |
| | 1111 | 0.9667 | 1.0000 | * | * | * | * | * | * |
| 5 | 000001 | 0.9984 | 1.0000 | * | * | * | * | * | * |
| | 01001 | 0.9958 | 1.0000 | * | * | * | * | * | * |
| | 11111 | 0.8469 | 0.9975 | 0.9998 | 1.0000 | * | * | * | * |
| 6 | 0000001 | 0.9988 | 1.0000 | * | * | * | * | * | * |
| | 010001 | 0.9971 | 1.0000 | * | * | * | * | * | * |
| | 111111 | 0.3363 | 0.4506 | 0.8050 | 0.9417 | 0.9364 | 0.9667 | 0.9710 | 0.9641 |
We can see the improved identification method is especially effective in the cases when \( B = 00001 \) and \( 000001 \). In the case of \( B = 111111 \), the success ratio's are as the same as those in Takeda et al. (2014). When we succeed in identification, we are interested in how many templates the set includes. We hope that the numbers of templates are small. Table 2 shows the numerical mean of the elements of the set of templates, when the set of the templates include the embedded pattern \( \tilde{B} \).

Table 2: mean of number of the elements

| \( \hat{m} \) | \( B \) | 0.01 | 0.03 | 0.05 | 0.06 | 0.07 | 0.08 | 0.09 | 0.10 |
|---|---|---|---|---|---|---|---|---|---|
| 4 | 0001 | 37.2 | 45.7 | 43.2 | 42.3 | 41.6 | 41.2 | 40.9 | 40.7 |
| | 0101 | 54.4 | 60.3 | 52.2 | 48.6 | 45.7 | 43.6 | 41.9 | 40.7 |
| | 1111 | 56.5 | 47.0 | 38.6 | 36.9 | 35.8 | 35.1 | 34.8 | 34.5 |
| 5 | 00001 | 39.5 | 39.8 | 35.3 | 33.9 | 32.9 | 32.2 | 31.8 | 31.4 |
| | 01001 | 39.0 | 41.5 | 37.7 | 36.5 | 35.6 | 34.9 | 34.5 | 34.1 |
| | 11111 | 61.9 | 56.2 | 45.0 | 41.2 | 38.4 | 36.4 | 34.9 | 33.7 |
| 6 | 000001 | 42.9 | 41.9 | 36.6 | 34.2 | 32.5 | 31.4 | 30.6 | 30.1 |
| | 010001 | 29.9 | 35.6 | 34.3 | 33.6 | 33.1 | 32.8 | 32.7 | 32.5 |
| | 111111 | 65.7 | 58.3 | 44.6 | 40.3 | 37.0 | 34.7 | 32.9 | 31.5 |

In all cases, the mean values are 30 \( \sim \) 70. Because the number of all templates from 3 bits to 9 bits is 1,016, we consider that we could reduce the number of candidates of the embedded pattern \( \tilde{B} \). As far as our simulation studies are concerned, the improved method shows satisfactory results for all cases.

5. Conclusion

In Takeda et al. (2014), we proposed a method of identifying patterns that appear too many times in a binary sequence. We showed that the method was effective in many cases, but had difficulty in some cases. In this paper, we investigated causes why our method had difficulty in some cases and, by taking account of the causes, we proposed an improved method. We examined the effectiveness of the improved method by simulation studies. By these studies, we have shown our improved method is effective for a wider class of cases as far as we examined. Our future work is to find a better method for identification, e.g., by using another method instead of the Kolmogorov-Smirnov test and by taking into consideration the concept of multiple comparisons.

Acknowledgments

The authors thank the editor and the referees for helpful comments and suggestions, which were very useful for improving the paper. This work was supported by “The Research on Security and Reliability in Electronic Society”, Chuo University 21st Century COE Program.
Huzii, M., Takeda, Y., Watanabe, N., Kamakura, T. and Sugiyama, T. (2006). Testing randomness for cryptographic applications. (in Japanese) Journal of the Japan Statistical Society 35, 181–199.

Rukhin, A., Soto, J., Nechvatal, J., Smid, M., Barker, E., Leigh, S., Levenson, M., Vangel, M., Banks, D., Heckert, A., Dray, J. and Vo, S. (2010). A statistical test suite for random and pseudorandom number generators for cryptographic applications. NIST Special Publication 800-22 Revision 1a, The National Institute of Standards and Technology, U.S.A.

Takeda, Y., Huzii, M., Watanabe, N. and Kamakura, T. (2014). Modified Non-overlapping template matching test and proposal on setting template. Journal of Japanese Society of Computational Statistics 27, 49–60.
Appendix

Specified Set (Sp.Set) means the set of templates in which the pattern we seek would be included.

start

Short-Bit-NO-Test length $m^*$
Short-Bit-NO-Test for $B^{1*}$
(all $m^*$-bit templates)
$p$-value $p^{1*(t)}$ of $t$-th element of $B^{1*}$

We have a rejected template.

No
Accept SNH

Short-Bit-NO-Test for $B^{2*} = B^{1*} \oplus B^{1*}$
$p^{2*(t)}$ for $t$-th element $B^{2*(t)} \in B^{2*}$
$B^{2*} = \{\text{set of all rejected templates among } B^{2*}\}$

$\exists B^{2*(t)} \in B^{2*}$
$p^{2*(t)} < \min(p^{1*(u)}, p^{1*(v)})$
when $B^{2*(t)} = B^{1*(u)} \oplus B^{1*(v)}$

No

Yes

$B^{2*} = B^{2*}$

$B^{2*} = \{B^{2*(t)}\}$
$B^{2*} = B^{2*} \cup B^{3*}$

Yes

$B^{2*} = \phi$

Yes

Sp.Set=$B^{1*}$

$\kappa = 3$

$\kappa$-th step
\[ \kappa \text{-th step} \]

Short-Bit-NO-Test for \( B_1^{\kappa*} = B^{(\kappa-1)*} \oplus B^{(\kappa-1)*} \)

\[ p^{\kappa*}(t) \] for \( t \)-th element \( B^{\kappa*}(t) \in B^{\kappa*} \)

\( B_2^{\kappa*} = \text{set of all rejected templates among } B_1^{\kappa*} \)

\[ \exists B^{\kappa*}(t) \in B^{\kappa*} \]
\[ p^{\kappa*}(t) < \min(p^{(\kappa-1)*}(u), p^{(\kappa-1)*}(v)) \]
when \( B^{\kappa*}(t) = B^{(\kappa-1)*}(u) \oplus B^{(\kappa-1)*}(v) \)

\[ B_2^{\kappa*} = \{ B^{\kappa*}(t) \} \]
\[ B^{\kappa*} = B_2^{\kappa*} \cup B_3^{\kappa*} \]

\[ B^{\kappa*} = \phi \]

Determining Sp. Set #1

\[ \kappa < \kappa_0 \]

Determining Sp. Set #2

\[ \kappa = \kappa + 1 \]
Improved Non-overlapping Template Matching Test

Determining Sp. Set #1

\[ \exists B^{(\kappa-2)\ast}(t) \in B^{(\kappa-2)\ast} \]
\[ p^{(\kappa-2)\ast}(t) \leq \min p^{(\kappa-1)\ast}(t) \]

No \quad \text{Sp.Set} = B^{(\kappa-1)\ast}

Yes \quad \text{Sp.Set} = B^{(\kappa-2)\ast}

\{ \text{Stop} \}

Determining Sp. Set #2

\[ \exists B^{(\kappa_0-1)\ast}(t) \in B^{(\kappa_0-1)\ast} \]
\[ p^{(\kappa_0-1)\ast}(t) \leq \min p^{(\kappa_0)\ast}(t) \]

No \quad \text{Sp.Set} = B^{\kappa_0\ast}

Yes \quad \text{Sp.Set} = B^{(\kappa_0-1)\ast}

\{ \text{Stop} \}

(Received: November 30, 2016, Accepted: August 9, 2017)
