Một số cải tiến cho các kiểm tra thống kê tính ngẫu nhiên sử dụng so khớp mẫu

Hoàng Đình Linh, Trần Thị Lượng

***Tóm tắt*—Các kiểm tra liên quan đến so khớp mẫu chồng lấp đã được đề xuất trong NIST SP 800-22 [1], tuy nhiên các xác suất trong các kiểm tra này chỉ đúng cho các mẫu đặc biệt và cần được tính lại cho các mẫu khác. Trong [2], các tác giả đã đề xuất các tiêu chuẩn thống kê so khớp mẫu mới cho tất cả các mẫu 4 bit. Các kiểm tra mới này áp dụng cho chuỗi bất kỳ có độ dài tối thiểu là 5504 bit, trong khi theo NIST độ dài tối thiểu 10^6 bit. Trong bài báo này, chúng tôi đã cải tiến và đề xuất các kiểm tra so khớp mẫu 4 bit mới mà có thể áp dụng cho các chuỗi bất kỳ có độ dài nhỏ nhất chỉ là 3726 bit. Hơn nữa, chúng tôi đưa ra 3 kiểm tra thống kê so khớp mẫu 5 bit mới. Kết quả lý thuyết và thực hành cho thấy các đề xuất cải tiến của chúng tôi là rất hiệu quả trong việc đánh giá tính ngẫu nhiên cho các bộ tạo số giả ngẫu nhiên.**

***Abstract*—Randomness tests related to overlapping template matching have been proposed in NIST SP 800-22 [1], however the probabilities in these tests are only valid for specific samples and should be recalculated for other samples. In [2], the authors proposed new template matching tests for all 4-bit templates. The new tests can be applied to any sequence of minimum length of 5504 bits whereas the overlapping template matching test in the NIST test suite can only be applied to sequences of minimum length of 10^6 bits. In this paper, we have modified and proposed new 4-bit template matching tests that can be applied to any sequence of minimum length 3726 bits. Furthermore, we proposed three new 5-bit template matching tests. Our theoretical and practical results show that our new proposed tests are very efficient in psedorandom number generator testing.**

Từ khóa—Kiểm tra tính ngẫu nhiên, so khớp mẫu, 5 bit, 4 bit.

Keywords—randomness testing, template matching, 5-bit, 4-bit.

Bài báo được nhận ngày 22/10/2020. Bài báo được nhận xét bởi phản biện thứ nhất ngày 27/10/2020 và được chấp nhận đăng ngày 27/10/2020. Bài báo được nhận xét bởi phản biện thứ hai ngày 12/5/2021 và được chấp nhận đăng ngày 16/5/2021.

I. Giới thiệu

Trong mật mã học, các số ngẫu nhiên đóng vai trò quan trọng, tuy nhiên việc tạo ra các số ngẫu nhiên cho các mục đích mật mã là một nhiệm vụ khó. Các số ngẫu nhiên được tạo ra một cách lý tưởng từ các nguồn ngẫu nhiên thực sự, được gọi là các bộ tạo số ngẫu nhiên thực sự (TRNG), các bộ tạo này sử dụng các nguồn không tất định để tạo các số ngẫu nhiên. Tuy nhiên, việc tạo các số ngẫu nhiên sử dụng TRNG là không hiệu quả, mặt khác rất khó lưu trữ và truyền giao một số lượng lớn các bit ngẫu nhiên. Do đó, các thuật toán tất định được gọi là các bộ tạo số giả ngẫu nhiên (PRNG) có ưu thế hơn TRNG. PRNG sử dụng chuỗi nhị phân ngẫu nhiên thực sự có độ dài k (thường được gọi là mầm) và tạo ra chuỗi nhị phân tuần hoàn "trông có vẻ ngẫu nhiên" có độ dài $l > k$ [3]. Các đặc tính của PRNG là khác so với TRNG. Đầu tiên, PRNG là hiệu quả hơn khi so sánh với TRNG, vì mất ít thời gian hơn để tạo được cùng một số bit ở đầu ra. Các PRNG là các thuật toán tất định, tức là một chuỗi số cho trước là có thể lặp lại. Các PRNG có chu kỳ còn TRNG thì không.

Một kiểm tra tính ngẫu nhiên theo thống kê (gọi tắt là kiểm tra ngẫu nhiên thống kê) được đề xuất để kiểm tra giả thiết không (H_0) phát biểu rằng chuỗi đầu vào là ngẫu nhiên. Phép kiểm tra sẽ nhận một chuỗi đầu vào nhị phân và đưa ra kết luận "chấp nhận" hay "bác bỏ" giả thiết này. Các kiểm tra ngẫu nhiên thống kê có 2 loại xác suất sai lầm. Nếu dữ liệu là ngẫu nhiên và kết luận là bác bỏ H_0 thì đó là xác suất sai lầm loại I, nếu dữ liệu là không ngẫu nhiên và kết luận là chấp nhận H_0 thì đó là loại II. Xác xuất của loại I được gọi là mức ý nghĩa của kiểm tra và thường được ký hiệu là α. Một kiểm tra ngẫu nhiên thống kê sẽ đưa ra một giá trị là số thực nằm giữa 0 và 1, được gọi là *p-value*. Nếu *p-value* $> \alpha$ thì H_0 được chấp nhận, ngược lại là bác bỏ. Mức ý nghĩa phụ thuộc

vào các ứng dụng, và đối với các ứng dụng mật mã thường lấy bằng 0.01 [1].

Các chuỗi đầu ra của PRNG phải trông có vẻ ngẫu nhiên, do đó các phân tích thống kê PRNG là cần thiết. Quá trình này được thực hiện bằng cách sử dụng PRNG tạo ra một chuỗi mẫu và đánh giá nó bởi bộ các kiểm tra tính ngẫu nhiên bằng thống kê. Có nhiều bộ kiểm tra thống kê tính ngẫu nhiên trong các tài liệu [1], [4]-[7] chứa nhiều các kiểm tra thống kê tính ngẫu nhiên khác nhau. Cũng được coi như các PRNG đó là các đầu ra của các nguyên thủy mật mã chẳng hạn như mã khối và hàm băm. Do đó, các đầu ra này phải trông có vẻ ngẫu nhiên sao cho khi phân tích các đầu ra, thì việc dự đoán thuật toán là không thể. Do vậy, việc đánh giá các đầu ra của các thuật toán bằng các kiểm tra thống kê tính ngẫu nhiên có tầm quan trọng rất lớn.

Bộ kiểm tra NIST [8] là bộ kiểm tra thông dụng nhất cho các ứng dụng mật mã. Phân tích thống kê các thuật toán ứng cử AES cuối cùng được Soto và cộng sự thực hiện sử dụng bộ kiểm tra NIST [9]. Một số kiểm tra yêu cầu chuỗi có độ dài 10^6, trong khi đầu ra các thuật toán cuối AES là 128 bit. Soto và các cộng sự đã nối các đầu ra của các thuật toán để nhận được chuỗi dài mà áp dụng được tất cả các kiểm tra. Gần đây, Sulak và cộng sự đề xuất một phương pháp thay thế, khi họ tính và sử dụng các phân bố chính xác thay vì phân bố xấp xỉ hoặc phân bố tiệm cận [10]. Có xác suất chính xác, không cần chuỗi dài nữa, và họ áp dụng các kiểm tra ngẫu nhiên trực tiếp cho các đầu ra của các thuật toán thay vì ghép chúng.

Đã có một số nghiên cứu đối với các kiểm tra trong bộ kiểm tra NIST [11]-[15]. Okutomi và cộng sự đã áp dụng các kiểm tra trong bộ kiểm tra NIST cho các dữ liệu ngẫu nhiên được lấy từ các thuật toán mật mã DES và SHA-1 [14]. Họ quan sát thấy kiểm tra thống kê Maurer và kiểm tra so khớp mẫu chồng lấp có vấn đề với tỷ lệ của dữ liệu ngẫu nhiên vượt qua các kiểm tra. Hamano và cộng sự đã chỉnh sửa các xác suất cho kiểm tra so khớp mẫu chồng lấp, ở đây họ lấy mẫu B = 111111111 [15] và NIST đã cập nhật các xác suất này. Tuy nhiên, như đã lưu ý ở [28], xác suất của mỗi mẫu phụ thuộc vào chính nó. Trong [2], các tác giả xem xét $m = 4$ và phân loại 16 mẫu có thể thành bốn nhóm. Sau đó, đề xuất 4 kiểm tra thống kê ngẫu nhiên mới mà có thể áp dụng cho các chuỗi có độ dài tối thiểu là 5504 bit. Trong bài báo này, chúng tôi đã chỉnh sửa lại 4 kiểm tra thống kê trên cho phép áp dụng được cho các chuỗi có độ dài tối thiểu chỉ 3726 bit. Hơn nữa, chúng tôi đưa ra 3 kiểm tra thống kê so khớp mẫu 5 bit mới.

Bài báo này cấu trúc như sau: Trong Phần II trình bày một số kiến thức chuẩn bị. Phần III nhắc lại một số kết quả trong [2]. Phần IV trình bày một số kết quả chỉnh sửa đối với các kiểm tra trong [2] cho phép áp dụng cho các chuỗi có độ dài ngắn hơn và đề xuất 3 kiểm tra so khớp mẫu 5 bit mới. Trong phần V, chúng tôi áp dụng các kiểm tra mới cho dữ liệu ngẫu nhiên và không ngẫu nhiên, đồng thời quan sát sức mạnh các kiểm tra mới. Kết luận bài báo và đề xuất một số công việc tiếp theo được trình bày trong Phần VI.

II. MỘT SỐ KIẾN THỨC CHUẨN BỊ

Phân bố χ^2 được sử dụng để so sánh chuỗi các sự việc được quan sát tốt thế nào so khớp với chuỗi được kỳ vọng dưới phân bố giả thiết.

Định nghĩa 1 ([8]). Biến ngẫu nhiên có phân bố χ^2 với υ bậc tự do nếu hàm mật độ xác suất tương ứng $f(x) = 0$ với $x < 0$ và

$$f(x) = \frac{1}{\Gamma\left(\frac{\upsilon}{2}\right)2^{\upsilon/2}} x^{\frac{\upsilon}{2}-1} e^{-\frac{x}{2}}, x \geq 0,$$

ở đây υ là số nguyên dương và Γ là hàm *gamma*:

$$\Gamma(t) = \int_0^\infty x^{t-1} e^{-x} dx \text{ với } t > 0.$$

Kiểm tra so khớp χ^2 là kiểm tra thống kê ngẫu nhiên, ở đây kiểm tra thống kê có phân bố χ^2, coi H_0 là đúng. Nói cách khác, giả sử E_i là các giá trị kỳ vọng, F_i là các giá trị quan sát với $1 \leq i \leq k$. Khi đó:

$$\chi^2 = \sum_{i=1}^{k} \frac{(F_i - E_i)^2}{E_i} \text{ và } p\text{-}value = igamc\left(2, \frac{\chi^2}{2}\right),$$

trong đó *igamc* là hàm *gamma* không đầy đủ [16].

Tiếp theo, chúng ta nhắc lại một số kết quả tổ hợp dùng để tính các xác suất phân bố của các mẫu trong các phần tiếp theo.

Bổ đề 1 ([17]). Số nghiệm nguyên không âm của phương trình $x_1 + x_2 + ... + x_b = a$ là:

$$\binom{a+b-1}{b-1}.$$

Bổ đề 2 ([17]). Số nghiệm nguyên của phương trình $x_1 + x_2 + ... + x_b = a$ với $x_i \geq c, 1 \leq i \leq b$ là

$$\binom{a-b(c-1)-1}{b-1}.$$

Bổ đề 3 ([17]). [Nguyên lý loại trừ] Số nghiệm nguyên không âm của phương trình $x_1 + x_2 + ... + x_b = a$ với $x_i \leq c, 1 \leq i \leq b$ là

$$\sum_{j=0}^{b} \binom{a+b-1-j(c+1)}{b-1} \binom{b}{j} (-1)^j .$$

III. Các kiểm tra so khớp mẫu 4 bit đề xuất bởi F. Sulak

Đối tượng của các kiểm tra so khớp mẫu 4 bit đề xuất bởi F. Sulak là tần số của mẫu 4 bit cho trước trong chuỗi nhị phân. Các kiểm tra tương tự được định nghĩa trong bộ kiểm tra NIST, đó là kiểm tra so khớp mẫu không chồng lấp và mẫu chồng lấp. Trong cả hai kiểm tra, mẫu B có độ dài m bit được chọn, chuỗi đối tượng của kiểm tra được chia thành N chuỗi con có độ dài M. Cửa sổ m bit được sử dụng để tìm các khối chồng lấp m bit của mỗi chuỗi con. Sau đó với mỗi khối, số các mẫu B trong các chuỗi con được đếm. Ký hiệu W_i là số lượng mẫu B trong khối thứ i. Với kiểm tra so khớp chồng lấp $M = 1032$ bit. Ký hiệu π_j là xác suất để $W_i = j$, với $0 \leq j \leq 4$ và π_5 là xác suất để $W_i \geq 5$. Với $M = 1032$ và B = 111111111, các xác suất chính xác đã được tính trong [15] sử dụng phép đệ quy. Sử dụng các xác suất này trong kiểm tra so khớp χ^2 cho kết quả *p-value*.

Đối với kiểm tra so khớp không chồng lấp, mẫu cho trước được chọn theo cách nếu quan sát thấy mẫu ở đâu đó trong chuỗi thì không thể nhìn thấy mẫu này trước khi kết thúc mẫu đó. Như đã lưu ý trong bộ kiểm tra NIST, nếu đã thấy mẫu ở đâu đó trong chuỗi, không thể lại thấy nó ở $m-1$ khối tiếp theo, vì cửa sổ m bit trượt đi m bit. Điều này cho thấy rằng, phân bố là như nhau cho tất cả các mẫu không chồng lấp. Sử dụng ý tưởng đó, chúng tôi đưa ra định nghĩa mẫu chồng lấp k bit.

Định nghĩa 2. Cho B là một chuỗi nhị phân độ dài n. Chuỗi B' được tạo bằng cách nối B với $n-1$ bit bất kỳ. Khi đó, B được gọi là có k bit chồng lấp nếu $n-1-k$ khối tiếp theo của chuỗi B' có độ dài n thu được bằng cách lần lượt dịch sang phải 1 bit trong B' không thể trùng với B.

Từ định nghĩa trên, thì Mệnh đề 1 dưới đây đã được đưa ra trong [2].

Mệnh đề 1 (Proposition 3.1, [2]). Phân bố tần số của các mẫu chỉ định trước chỉ phụ thuộc vào số lượng các bit chồng lấp trong mẫu.

Sử dụng mệnh đề này, chúng ta phân loại các mẫu 4 bit theo số các bit chồng lấp. Có 4 dạng các khối:

1. Các khối không chồng lấp: 0001, 0011, 0111, 1000, 1100, 1110
2. Các khối chồng lấp 1 bit: 0010, 0100, 0110, 1001, 1011, 1101
3. Các khối chồng lấp 2 bit: 0101, 1010
4. Các khối chồng lấp 3 bit: 0000, 1111

Ví dụ, 0010 là mẫu 4 bit chồng lấp 1 bit theo Định nghĩa 1. Ta có $B' = 0010***$, trong đó $*$ có thể là bit 0 hoặc 1. Ta lần lượt dịch sang phải 1 bit đối với chuỗi B' bắt đầu từ khối 0010 ta thu được các khối sau: $010*, 10**, 0***$. Rõ ràng 2 khối đầu tiên không thể trùng với khối 0010, chỉ có khối thứ 3 có thể trùng với khối 0010. Do đó, theo Định nghĩa 1, khối 0010 là khối chồng lấp k bit và $4 - 1 - k = 2$ hay $k = 1$.

Để thực hiện kiểm tra, chúng ta chọn mỗi khối tương ứng từ mỗi dạng và tìm chính xác phân bố. Khác với các tiếp cận trước đây, trong [2] xét các bit vòng trong mỗi chuỗi con. Trong [2], tác giả đã đưa ra các kết quả lý thuyết cho 4 trường hợp của các mẫu 4 bit đồng thời đề xuất 4 tiêu chuẩn kiểm tra so khớp mẫu 4 bit mới được áp dụng cho các chuỗi nhị phân có độ dài tối thiểu là 5504 bit. Trong phần này, chúng tôi nhắc lại các kết quả lý

thuyết trong [2] và đề xuất chỉnh sửa lại 4 tiêu chuẩn so khớp mẫu 4 bit cho phép áp dụng được cho các chuỗi có độ dài tối thiểu là 3726 bit.

A. Nhắc lại một số kết quả lý thuyết trong [2]

Để xây dựng kiểm tra ngẫu nhiên, chúng ta cần tìm xác suất mà một mẫu cho trước B xuất hiện k lần trong chuỗi con. Đối với trường hợp không chồng lấp, chọn $B = 0001$ và tính xác suất tương ứng. Coi như đã biết trọng số W, và số lượng các loạt V của chuỗi.

Định lý 1 (Theorem 3.6, [2]). Giả sử $\{a_1, a_2, ..., a_n\}$ là chuỗi nhị phân và $b_i = a_i a_{i+1} a_{i+2} a_{i+3}$ là các khối có độ dài 4 với $1 \le i \le n$, và $a_{n+j} = a_j$ với $j = 1,2,3$, và giả sử K là số khối 0001 trong các khối $b_i, 1 \le i \le n$. Gọi w là trọng số của chuỗi và $2r$ là số các loạt trong chuỗi. Nếu chuỗi không phải toàn 0 hoặc toàn 1 thì:

$$\Pr(K=k) = \frac{n}{r \cdot 2^n} \binom{w-1}{r-1} \binom{r}{k} \times \times \sum_{a=0}^{r-k} \binom{r-k}{a} \binom{n-w-r-a-k-1}{k-1}.$$

Trong kiểm tra so khớp mẫu, cần tìm xác suất xuất hiện mẫu với bất kỳ trọng số và số lượng các loạt có thể của chuỗi. Từ đó có hệ quả sau.

Hệ quả 1 (Corollary 3.8, [2]). Giả sử $\{a_1, a_2, ..., a_n\}$ là chuỗi nhị phân và $b_i = a_i a_{i+1} a_{i+2} a_{i+3}$ là các khối có độ dài 4, với $1 \le i \le n$ và $a_{n+j} = a_j$ với $j = 1,2,3$, và giả sử K là số khối 0001 trong các khối $b_i, 1 \le i \le n$. Nếu chuỗi không phải toàn 0 hoặc toàn 1 thì:

$$\Pr(K=k) = \sum_{w=1}^{n-1} \sum_{r=1}^{\lfloor n/2 \rfloor} \frac{n}{r \cdot 2^n} \binom{w-1}{r-1} \binom{r}{k} \times \times \sum_{a=0}^{r-k} \binom{r-k}{a} \binom{n-w-r-a-k-1}{k-1}.$$

Đối với trường hợp trùng khớp 1 bit, chọn $B = 0110$ và nhận xác xuất tương ứng.

Định lý 2 (Theorem 3.9, [2]). Giả sử $\{a_1, a_2, ..., a_n\}$ là chuỗi nhị phân và $b_i = a_i a_{i+1} a_{i+2} a_{i+3}$ là các khối có độ dài 4 với $1 \le i \le n$ và $a_{n+j} = a_j$ với $j = 1,2,3$ và giả sử K là số khối 0110 trong các khối b_i với $1 \le i \le n$. Gọi w là trọng số của chuỗi và $2r$ là số các loạt trong chuỗi. Nếu chuỗi không phải toàn 0 hoặc toàn 1 thì:

$$\Pr(K=k) = \frac{n}{r \cdot 2^n} \binom{n-w-1}{r-1} \binom{r}{k} \times \sum_{a=0}^{r-k} \binom{r-k}{a} \binom{n-2r+a-1}{r-k-a-1}.$$

Hệ quả 2 (Corollary 3.10, [2]). Giả sử $\{a_1, a_2, ..., a_n\}$ là chuỗi nhị phân và $b_i = a_i a_{i+1} a_{i+2} a_{i+3}$ là các khối có độ dài 4 với $1 \le i \le n$ và $a_{n+j} = a_j$ với $j = 1,2,3$ và giả sử K là số khối 0110 trong các khối b_i với $1 \le i \le n$. Nếu chuỗi không phải toàn 0 hoặc toàn 1 thì:

$$\Pr(K=k) = \sum_{w=1}^{n-1} \sum_{r=1}^{\lfloor n/2 \rfloor} \frac{n}{r \cdot 2^n} \binom{n-w-1}{r-1} \binom{r}{k} \times \times \sum_{a=0}^{r-k} \binom{r-k}{a} \binom{w-2r+a-1}{r-k-a-1}.$$

Đối với trường hợp chồng lấp 2 bit, thì chọn trước khối 1010. Khác với định lý 1 và 2 để nhận được xác suất thì sử dụng mô hình khác, ở đó x_i là các hộp màu đỏ và y_i là các hộp màu trắng.

Định lý 3 (Theorem 3.11, [2]). Giả sử $\{a_1, a_2, ..., a_n\}$ là chuỗi nhị phân và $b_i = a_i a_{i+1} a_{i+2} a_{i+3}$ là các khối có độ dài 4 với $1 \le i \le n$ và $a_{n+j} = a_j$ với $j = 1,2,3$ và giả sử K là số các khối 1010 trong các chuỗi b_i với $1 \le i \le n$. Gọi w là trọng số của chuỗi và $2r$ là số các loạt trong chuỗi. Nếu chuỗi không phải toàn 0 hoặc toàn 1 thì:

$$\Pr(K=k) = \frac{n}{r \cdot 2^n} \sum_{a=k}^{r-1} \binom{r}{a} \binom{a}{k} \times \binom{n-w-r-1}{r-a-1} \binom{w-a-1}{r-k-1}.$$

Hệ quả 3 (Corollary 3.12, [2]). Giả sử $\{a_1, a_2, ..., a_n\}$ là chuỗi nhị phân và $b_i = a_i a_{i+1} a_{i+2} a_{i+3}$ là các khối có độ dài 4 với $1 \le i \le n$ và $a_{n+j} = a_j$ với $j = 1,2,3$ và giả sử K là số khối 1010 trong các chuỗi b_i với $1 \le i \le n$. Nếu chuỗi không phải toàn 0 hoặc toàn 1 thì:

$$\Pr(K = k) = \sum_{w=2}^{n-1} \sum_{r=2}^{\lfloor n/2 \rfloor} \frac{n}{r \cdot 2^n} \sum_{a=k}^{r-1} \binom{r}{a} \binom{a}{k} \times \times \binom{n-w-r-1}{r-a-1} \binom{w-a-1}{r-k-1}.$$

Chúng ta chọn trước khối 1111 cho trường hợp chồng lấp 3 bit. Áp dụng nguyên lý loại trừ để thu được xác suất.

Định lý 4 (Theorem 3.13, [2]). Giả sử $\{a_1, a_2, ..., a_n\}$ là chuỗi nhị phân và $b_i = a_i a_{i+1} a_{i+2} a_{i+3}$ là các khối có độ dài 4 bit với $1 \le i \le n$ và $a_{n+j} = a_j$ với $j = 1,2,3$ và giả sử K là số khối 1111 trong các khối b_i với $1 \le i \le n$. Gọi w là trọng số của chuỗi và $2r$ là số các loạt trong chuỗi. Nếu chuỗi không phải toàn 0 hoặc toàn 1 thì:

$$\Pr(K = k) = \frac{n}{r \cdot 2^n} \binom{n-w-1}{r-1} \sum_{t=0}^{r} \binom{r}{t} \binom{k-1}{t-1} \times \times \sum_{i=0}^{r} \binom{w-k-3t-3i-1}{r-t-1} \binom{r-t}{i} (-1)^i.$$

Hệ quả 4 (Corollary 3.14, [2]). Giả sử $\{a_1, a_2, ..., a_n\}$ là chuỗi nhị phân và $b_i = a_i a_{i+1} a_{i+2} a_{i+3}$ là các khối có độ dài 4 bit với $1 \le i \le n$ và $a_{n+j} = a_j$ với $j = 1,2,3$ và giả sử K là số khối 1111 trong các khối b_i với $1 \le i \le n$. Nếu chuỗi không phải toàn 0 hoặc toàn 1 thì:

$$\Pr(K = k) = \sum_{w=1}^{n-1} \sum_{r=1}^{\lfloor n/2 \rfloor} \frac{n}{r \cdot 2^n} \binom{n-w-1}{r-1} \times \sum_{t=0}^{r} \binom{r}{t} \binom{k-1}{t-1} \sum_{i=0}^{r} \binom{w-k-3t-3i-1}{r-t-1} \binom{r-t}{i} (-1)^i.$$

Chú ý 1: Công thức tính xác suất trong định lý 4 và hệ quả 4 sử dụng nguyên lý loại trừ chính xác hơn phải là:

$$\sum_{i=0}^{r-t} \binom{w-k-3t-3i-1}{r-t-1} \binom{r-t}{i} (-1)^i.$$

Do khi $i \ge r-t$ thì $\binom{r-t}{i} = 0$ nên các giá trị xác suất tính được vẫn chính xác.

B. Mô tả kiểm tra so khớp mẫu 4 bit đề xuất bởi F. Sulak

Đối tượng của các kiểm tra so khớp 4 bit là số lượng các mẫu cho trước có trong chuỗi. Chúng ta áp dụng kiểm tra so khớp χ^2 để đánh giá các giá trị quan sát tốt thế nào so với các giá trị kỳ vọng. Với mục đích này, chúng ta chia chuỗi thành các khối 128 bit và tìm số lần xuất hiện của các mẫu trong mỗi khối. Sau đó, áp dụng kiểm tra χ^2 với 5 khoảng và cho giá trị *p-value* sử dụng Bảng 1. Các xác suất trong Bảng 1 được tính sử dụng các Hệ quả 1, 2, 3, 4.

BẢNG 1 [2]. CÁC XÁC SUẤT KHOẢNG CHO KIỂM TRA SO KHỚP MẪU 4 BIT

0 bit		**1 bit**	
0-6	0.24205627	0-5	0.16353105
7	0.17082354	6-7	0.27433485
8	0.18629989	8	0.15466485
9	0.16401892	9-10	0.24482145
10-128	0.23680138	11-128	0.16264780
2 bit		**3 bit**	
0-5	0.19990529	0-4	0.21976555
6-7	0.25650023	5-6	0.18737326
8-9	0.25566342	7-8	0.18572664
10-11	0.16931656	9-10	0.15200493
12-128	0.11861450	11-128	0.25512962

Giả sử chúng ta muốn kiểm tra chuỗi nhị phân có độ dài n sử dụng kiểm tra so khớp mẫu. Chúng ta có thể tóm tắt thủ tục như sau:

- Chọn mẫu 4 bit B.
- Chia chuỗi thành M khối 128 bit:

$$(M = \left\lfloor \frac{n}{128} \right\rfloor).$$

- Với mỗi khối, viết 3 bit đầu tiên vào cuối khối.

- Tìm sự xuất hiện của B trong khối đầu tiên theo cách chồng lấp và tăng giá trị khoảng tương ứng thêm 1, ký hiệu là F_i, $1 \le i \le 5$. Lặp lại cho tất cả các khối.
- Áp dụng kiểm tra so khớp χ^2, tức là tính

$$\chi^2 = \sum_{i=1}^{5} \frac{(F_i - M.p_i)^2}{M.p_i} \text{ và}$$

$$p\text{-}value = igamc(2, \frac{\chi^2}{2})$$

trong đó p_i nhận được từ bảng 1 theo số các bit lặp trong mẫu B.

- Nếu $p\text{-}value < 0.01$ kết luận chuỗi không ngẫu nhiên, ngược lại kết luận chuỗi ngẫu nhiên.

Mã giả của kiểm tra được phát biểu ở Thuật toán 1. Vì giá trị mong muốn trong mỗi khoảng nhỏ nhất là 5, và xác suất bé nhất trong Bảng 2 là 0.1186145, đối tượng chuỗi để kiểm tra cần có ít nhất $128 \times \left\lfloor \frac{5}{0.1186145} \right\rfloor = 5504$ bit.

Thuật toán 1 (Algorithm 4.1, [2]). Kiểm tra so khớp mẫu $\left(\{a_1, a_2, ..., a_n\}, B\right)$

$F_1 = 0, F_2 = 0, F_3 = 0, F_4 = 0, F_5 = 0;$

$M = \left\lfloor \frac{n}{128} \right\rfloor;$

for $i \leftarrow 0$ **to** $M - 1$

do

for $j \leftarrow 1$ **to** 128

do

$b_j = a_{128i+j};$

$b_{129} = a_{128i+1}, b_{130} = a_{128i+2}, b_{131} = a_{128i+3};$

$count = 0;$

for $j \leftarrow 1$ **to** 128

do

if $b_j b_{j+1} b_{j+2} b_{j+3} = B;$

then count ++;

tăng F_i theo Bảng 1

Áp dụng kiểm tra so khớp χ^2 cho $F_1, F_2, F_3, F_4, F_5;$

return (*p-value*)

IV. MỘT SỐ ĐỀ XUẤT CẢI TIẾN KIỂM TRA SO KHỚP MẪU 4 BIT

A. Đề xuất chỉnh sửa kiểm tra so khớp mẫu 4 bit của F. Sulak cho phép áp dụng cho các chuỗi bit có độ dài ngắn hơn

Trong phần này, chúng tôi đề xuất các kiểm tra so khớp mẫu 4 bit mới bằng cách tính toán lại các giá trị xác suất mẫu 4 bit với chuỗi có độ dài 64 bit. Các giá trị xác suất thu được như sau:

BẢNG 2. CÁC XÁC SUẤT KHOẢNG CHO KIỂM TRA SO KHỚP MẪU 4 BIT MỚI

0 bit		**1 bit**	
0-2	0.1573780416	0-2	0.2085239984
3	0.2199712762	3	0.2040669059
4	0.2610253962	4	0.2161895238
5	0.2047024926	5	0.1735024332
2 bit		**3 bit**	
0-2	0.2475044083	0-2	0.2475044083
3	0.1911190740	3	0.1911190740
4	0.1888668388	4	0.1888668388
5-6	0.2551760741	5-6	0.2551760741

Giả sử chúng ta muốn kiểm tra chuỗi nhị phân có độ dài n sử dụng kiểm tra so khớp mẫu. Chúng ta có thể tóm tắt thủ tục như sau:

- Chọn mẫu 4 bit B.
- Chia chuỗi thành M khối 64 bit

$$(M = \left\lfloor \frac{n}{64} \right\rfloor).$$

- Với mỗi khối, viết 3 bit đầu tiên vào cuối khối.
- Tìm sự xuất hiện của B trong khối đầu tiên theo cách chồng lấp và tăng giá trị khoảng tương ứng thêm 1, ký hiệu là F_i, $1 \le i \le 5$. Lặp lại cho tất cả các khối.
- Áp dụng kiểm tra so khớp χ^2, tức là tính

$$\chi^2 = \sum_{i=1}^{5} \frac{(F_i - M.p_i)^2}{M.p_i} \text{ và}$$

$$p\text{-}value = igamc(2, \frac{\chi^2}{2})$$

trong đó p_i nhận được từ Bảng 2 theo số các bit lặp trong mẫu B.

- Nếu $p\text{-}value < 0.01$ kết luận chuỗi không ngẫu nhiên, ngược lại kết luận chuỗi ngẫu nhiên.
- Vì giá trị mong muốn trong mỗi khoảng nhỏ nhất là 5, và xác suất bé nhất trong bảng 3 là 0.0858776261, đối tượng chuỗi để kiểm tra cần có ít nhất $64 \times \left\lfloor \frac{5}{0.0858776261} \right\rfloor = 3726$ bit.

B. Một số kết quả đối với mẫu 5 bit

Sử dụng Định nghĩa 2, chúng tôi phân loại các mẫu 5 bit theo số các bit chồng lấp. Có 5 lớp như sau:

1. Các khối không chồng lấp: 00001(1), 00011(3), 00111(7), 01111(15), 10000(16), 11000(24), 11100(28), 11110(30), 01011(11), 10100(20), 11010(26), 00101(5).
2. Các khối chồng lấp 1 bit: 00010(2), 01000(8), 01110(14), 10001(17), 10111(23), 11101(29), 00110(6), 11001(25), 01100(12), 10011(19).
3. Các khối chồng lấp 2 bit: 10110(22), 01001(9), 11011(27), 00100(4), 01101(13), 10010(18).
4. Các khối chồng lấp 3 bit: 01010(10), 10101(21).
5. Các khối chồng lấp 4 bit: 00000(0), 11111(31).
6. Các giá trị trong dấu ngoặc đơn là giá trị số nguyên của chuỗi 5 bit tương ứng.

1. Trường hợp không chồng lấp

Để xây dựng kiểm tra ngẫu nhiên, chúng ta cần tìm xác suất mà một mẫu cho trước B xuất hiện k lần trong chuỗi con. Đối với trường hợp không chồng lấp, chúng ta chọn $B = 00001$ và tính xác suất phù hợp. Coi như đã biết trọng số W, và số lượng các loạt V của chuỗi.

Định lý 5. Giả sử $\{a_1, a_2, ..., a_n\}$ là chuỗi nhị phân và $b_i = a_i a_{i+1} a_{i+2} a_{i+3} a_{i+4}$ là các khối có độ dài 5 với $1 \le i \le n$, và $a_{n+j} = a_j$ với $j = 1,2,3,4$, và giả sử K là số khối 00001 trong các khối $b_i, 1 \le i \le n$. Gọi w là trọng số của chuỗi và $2r$ là số các loạt trong chuỗi. Nếu chuỗi không phải toàn 0 hoặc toàn 1 thì:

$$\Pr(K = k) = \frac{n}{r \cdot 2^n} \binom{w-1}{r-1} \binom{r}{k} \sum_{a=0}^{r-k} \binom{r-k}{a} \times \times \sum_{b=0}^{r-k-a} \binom{r-k-a}{b} \binom{n-w-r-2a-b-2k-1}{k-1}.$$

Chứng minh. Đầu tiên chú ý rằng, số các loạt là chẵn nếu chuỗi không phải toàn 0 hoặc toàn 1. Coi các bit sắp xếp trên một vòng tròn và chúng ta viết các số 1 và 0 liên liếp để có $2r$ loạt. Kết quả, có tất cả $w-r$ số 1 và $n-w-r$ số 0.

Vì tất cả các khối 00001 chứa các khối 01, nếu một loạt của các số 0 có nhiều hơn 3 số 0, nó tạo ra chính xác 1 khối 00001. Bây giờ, chúng ta tìm phân bố của $w-r$ số 1 và $n-w-r$ số 0 sao cho có k khối 00001. Số lượng sắp xếp như vậy bằng số nghiệm nguyên dương của hệ phương trình:

$$\begin{cases} x_1 + x_2 + ... + x_r = n - w - r \\ y_1 + y_2 + ... + y_r = w - r \end{cases}$$

với điều kiện bổ sung là có chính xác k số x_i thỏa mãn $x_i \ge 3$ với $1 \le i \le r$. Điều kiện này đảm bảo có chính xác k khối 00001. Phương trình thứ 2 có $\binom{w-1}{r-1}$ nghiệm theo Bổ đề 1.

$$\underbrace{x_1 + \cdots + x_k}_{\ge 3} + \underbrace{x_{k+1} + \cdots + x_{k+a}}_{=2} + \underbrace{x_{k+a+1} + \cdots + x_{k+a+b}}_{=1} + \underbrace{x_{k+a+b+1} + \cdots + x_r}_{=0} = n - w - r$$

Để tìm số các nghiệm của phương trình thứ nhất, chúng ta có thể giả sử rằng $x_i \ge 3$ với

$1 \le i \le k$ (với hệ số $\binom{r}{k}$), $x_j = 2$ với $k+1 \le j \le k+a$ (với hệ số $\binom{r-k}{a}$), $x_t = 1$ với $k+a+1 \le t \le k+a+b$ (với hệ số $\binom{r-k-a}{b}$), và $x_s = 0$ với $k+a+b+1 \le s \le r$; do đó số nghiệm không âm của phương trình thứ nhất phụ thuộc vào số nghiệm của phương trình:

$x_1 + x_2 + ... + x_k = n - w - r - 2a - b, x_i \ge 3, 1 \le i \le k$

là $\binom{n-w-r-2a-b-2k-1}{k-1}$ theo Bổ đề 2. Chú ý rằng nếu $k = 0$, chúng ta không thể áp dụng Bổ đề 2, trong trường hợp này coi như chỉ có 1 nghiệm. Chúng ta sử dụng điều này trong cả bài báo.

Hơn nữa, mỗi sắp xếp trên vòng tròn tạo ra n chuỗi. Tuy nhiên, vì có r khối 01, nên r chuỗi trong các chuỗi đó là trùng nhau. Do đó, xét tính đối xứng của vòng tròn, các chuỗi khác toàn số 0 hoặc là toàn số 1, chúng ta có:

$$\Pr(K=k) = \frac{n}{r \cdot 2^n}\binom{w-1}{r-1}\binom{r}{k}\sum_{a=0}^{r-k}\binom{r-k}{a} \times \sum_{b=0}^{r-k-a}\binom{r-k-a}{b}\binom{n-w-r-2a-b-2k-1}{k-1}.$$

■

Trong kiểm tra so khớp mẫu, chúng ta cần tìm xác suất không phụ thuộc vào trọng số và số lượng các loạt của chuỗi. Ta có hệ quả sau.

Hệ quả 5. Giả sử $\{a_1, a_2, ..., a_n\}$ là chuỗi nhị phân và $b_i = a_i a_{i+1} a_{i+2} a_{i+3} a_{i+4}$ là các khối có độ dài 5, với $1 \le i \le n$ với $a_{n+j} = a_j$ với $j = 1,2,3,4$, và giả sử K là số khối 00001 trong các khối $b_i, 1 \le i \le n$. Nếu chuỗi không phải toàn 0 hoặc toàn 1 thì:

$$\Pr(K=k) = \sum_{w=1}^{n-1}\sum_{r=1}^{\lfloor n/2 \rfloor}\frac{n}{r \cdot 2^n}\binom{w-1}{r-1}\binom{r}{k} \times \sum_{a=0}^{r-k}\binom{r-k}{a}\sum_{b=0}^{r-k-a}\binom{r-k-a}{b} \times \binom{n-w-r-2a-b-2k-1}{k-1}.$$

Chứng minh. Theo Định lý 5 chúng ta tính $Pr(K=k \mid W=w, V=2r)$, tính tổng cho tất cả các trọng số và các loạt, chúng ta nhận được $Pr(K=k)$. ■

2. Trường hợp chồng lấp 1 bit

Đối với trường hợp trùng khớp 1 bít, chúng ta chọn $B = 01110$ và nhận xác xuất tương ứng.

Định lý 6. Giả sử $\{a_1, a_2, ..., a_n\}$ là chuỗi nhị phân và $b_i = a_i a_{i+1} a_{i+2} a_{i+3} a_{i+4}$ là các khối có độ dài 5 với $1 \le i \le n$ và $a_{n+j} = a_j$ với $j = 1,2,3,4$ và giả sử K là số khối 01110 trong b_i với $1 \le i \le n$. Gọi w là trọng số của chuỗi và $2r$ là số các loạt trong chuỗi. Nếu chuỗi không phải toàn 0 hoặc toàn 1 thì:

$$\Pr(K=k) = \frac{n}{r \cdot 2^n}\binom{n-w-1}{r-1}\binom{r}{k}\sum_{a=0}^{r-k}\binom{r-k}{a} \times \sum_{b=0}^{r-k-a}\binom{r-k-a}{b}\binom{w-3r+a+2b-1}{r-k-a-b-1}.$$

Chứng minh. Sử dụng ý tưởng chứng minh của Định lý 1, chúng ta coi như các bít sắp xếp trên 1 hình tròn và chúng ta viết các số 1 và số 0 liên tiếp để có $2r$ loạt. Như vậy có tất cả $w-r$ số 1 và $n-w-r$ số 0.

Vì tất cả các khối 01110 chứa các khối 01, nếu loạt 1 chứa chính xác 3 số 1 thì nó có chính xác 1 khối 01110. Do đó, chúng ta cần tìm phân bố của $w-r$ số 1 và $n-w-r$ số 0 sao cho có k khối 01110. Số lượng các sắp xếp như vậy là số nghiệm nguyên dương của hệ:

$$\begin{cases} x_1 + x_2 + ... + x_r = n - w - r \\ y_1 + y_2 + ... + y_r = w - r \end{cases}$$

với điều kiện bổ sung là có chính xác k nghiệm $y_i = 2$ với $1 \le i \le r$. Điều kiện bổ sung này đảm

bảo có chính xác k khối 01110. Phương trình thứ nhất có $\binom{n-w-1}{r-1}$ nghiệm theo Bổ đề 1.

$$\underbrace{y_1+\cdots+y_k}_{=2}+\underbrace{y_{k+1}+\cdots+y_{k+a}}_{=1}+\underbrace{y_{k+a+1}+\cdots+y_{k+a+b}}_{=0}+\underbrace{y_{k+a+b+1}+\cdots+y_r}_{\geq 3}=w-r$$

Để tìm số các nghiệm của phương trình thứ hai, chúng ta giả sử rằng $y_i=2$ với $1\leq i\leq k$ (với hệ số $\binom{r}{k}$), $y_j=1$ với $k+1\leq j\leq k+a$ (với hệ số $\binom{r-k}{a}$), $y_t=0$ với $k+a+1\leq t\leq k+a+b$ (với hệ số $\binom{r-k-a}{b}$) $y_s\geq 3$ với $k+a+b+1\leq s\leq r$. Nói cách khác, chúng ta cần tìm nghiệm nguyên của phương trình:

$$y_{k+a+b+1}+y_{k+a+b+2}+\ldots+y_r=w-r-2k-a,\quad y_s\geq 3, k+a+b+1\leq s\leq r$$

mà là $\binom{w-r-2k-a-2(r-k-a-b)-1}{r-k-a-b-1}$ theo Bổ đề 2. Xét tính đối xứng của hình tròn, ta có:

$$\Pr(K=k)=\frac{n}{r\cdot 2^n}\binom{n-w-1}{r-1}\binom{r}{k}\sum_{a=0}^{r-k}\binom{r-k}{a}\times\sum_{b=0}^{r-k-a}\binom{r-k-a}{b}\binom{w-3r+a+2b-1}{r-k-a-b-1}.$$

■

Hệ quả 6. Giả sử $\{a_1,a_2,\ldots,a_n\}$ là chuỗi nhị phân và $b_i=a_ia_{i+1}a_{i+2}a_{i+3}a_{i+4}$ là các khối có độ dài 5 với $1\leq i\leq n$ và $a_{n+j}=a_j$ với $j=1,2,3,4$ và giả sử K là số khối 01110 trong các khối b_i với $1\leq i\leq n$. Nếu chuỗi không phải toàn 0 hoặc toàn 1 thì:

$$\Pr(K=k)=\sum_{w=1}^{n-1}\sum_{r=1}^{\lfloor n/2\rfloor}\frac{n}{r\cdot 2^n}\binom{n-w-1}{r-1}\binom{r}{k}\sum_{a=0}^{r-k}\binom{r-k}{a}\times\sum_{b=0}^{r-k-a}\binom{r-k-a}{b}\binom{w-3r+a+2b-1}{r-k-a-b-1}.$$

Chứng minh. Theo định lý 6 chúng ta tính $Pr(K=k\,|\,W=w,V=2r)$, tính tổng cho tất cả các trọng số và các loạt, chúng ta nhận được $Pr(K=k)$. ■

3. Trường hợp chồng lấp 4 bit

Chúng ta chọn trước khối 11111 cho trường hợp chồng lấp 4 bit. Áp dụng nguyên lý loại trừ để thu được xác suất.

Định lý 7. Giả sử $\{a_1,a_2,\ldots,a_n\}$ là chuỗi nhị phân và $b_i=a_ia_{i+1}a_{i+2}a_{i+3}a_{i+4}$ là các khối có độ dài 5 bit với $1\leq i\leq n$ và $a_{n+j}=a_j$ với $j=1,2,3,4$ và giả sử K là số khối 11111 trong các khối b_i với $1\leq i\leq n$. Gọi w là trọng số của chuỗi và $2r$ là số các loạt trong chuỗi. Nếu chuỗi không phải toàn 0 hoặc toàn 1 thì:

$$\Pr(K=k)=\frac{n}{r\cdot 2^n}\binom{n-w-1}{r-1}\sum_{t=0}^{r}\binom{r}{t}\binom{k-1}{t-1}\times\times\sum_{i=0}^{r-t}\binom{w-k-4t-4i-1}{r-t-1}\binom{r-t}{i}(-1)^i.$$

Chứng minh. Sử dụng ý tưởng chứng minh của Định lý 1, chúng ta coi như các bit sắp xếp trên 1 hình tròn và chúng ta viết các số 1 và số 0 liên tiếp để có $2r$ loạt. Như vậy có tất cả $w-r$ số 1 và $n-w-r$ số 0. Bây giờ, chúng ta tìm phân bố của $w-r$ số 1 và $n-w-r$ số 0 sao cho có k khối 11111 và $V=2r$. Hoàn toàn tương tự chúng ta cần tìm số các nghiệm nguyên của hệ

$$\begin{cases}x_1+x_2+\ldots+x_r=n-w-r, & x_i\geq 0, 1\leq i\leq r\\ y_1+y_2+\ldots+y_r=w-r, & y_j\geq 0, 1\leq j\leq r\end{cases}$$

thỏa mãn điều kiện có chính xác k khối 11111.

Phương trình thứ nhất có $\binom{n-w-1}{r-1}$ nghiệm theo Bổ đề 1. Không mất tính tổng quát (hoặc nhân với $\binom{r}{t}$), giả sử $y_j\geq 4, 1\leq j\leq t$ và $0\leq y_s\leq 3, t+1\leq s\leq r,$ chúng ta tìm số nghiệm của phương trình thứ 2 theo hai phần:

Mỗi loạt các số 1 có độ dài $l \geq 5$ cho $l-4$ khối 11111. Do đó để có k khối 11111, chúng ta cần có:

$$y_1 - 3 + y_2 - 3 + ... + y_t - 3 = k, y_j \geq 4$$
$$\Rightarrow y_1 + y_2 + ... + y_t = k + 3t, y_j \geq 4$$

do đó số nghiệm sẽ là $\binom{k-1}{t-1}$ (và $t = 0 \Leftrightarrow k = 0$) theo Bổ đề 2.

Vì trọng số của chuỗi là w, ta có:

$y_{t+1} + y_{t+2} + ... + y_r = w - r - k - 3t, 0 \leq y_j \leq 3, t+1 \leq j \leq r$, chúng ta áp dụng nguyên lý loại trừ để tìm số nghiệm và thu được

$$\sum_{i=0}^{r-t} \binom{w-k-4t-4i-1}{r-t-1} \binom{r-t}{i} (-1)^i.$$

Xét tính đối xứng của hình tròn, và chuỗi khác toàn 0 hoặc toàn 1, ta có:

$$\Pr(K=k) = \frac{n}{r \cdot 2^n} \binom{n-w-1}{r-1} \sum_{t=0}^{r} \binom{r}{t} \binom{k-1}{t-1} \times$$
$$\times \sum_{i=0}^{r-t} \binom{w-k-4t-4i-1}{r-t-1} \binom{r-t}{i} (-1)^i.$$

■

Hệ quả 7. Giả sử $\{a_1, a_2, ..., a_n\}$ là chuỗi nhị phân và $b_i = a_i a_{i+1} a_{i+2} a_{i+3} a_{i+4}$ là các khối có độ dài 5 với $1 \leq i \leq n$ và $a_{n+j} = a_j$ với $j = 1,2,3,4$ và giả sử K là số khối 11111 trong các khối b_i với $1 \leq i \leq n$. Nếu chuỗi không phải toàn 0 hoặc toàn 1 thì:

$$\Pr(K=k) = \sum_{w=1}^{n-1} \sum_{r=1}^{\lfloor n/2 \rfloor} \frac{n}{r \cdot 2^n} \binom{n-w-1}{r-1} \sum_{t=0}^{r} \binom{r}{t} \binom{k-1}{t-1}$$
$$\times \sum_{i=0}^{r-t} \binom{w-k-4t-4i-1}{r-t-1} \binom{r-t}{i} (-1)^i.$$

Chứng minh. Theo Định lý 7 chúng ta tính $Pr(K=k \mid W=w, V=2r)$, tính tổng cho tất cả các trọng số và các loạt, chúng ta nhận được $Pr(K=k)$. ■

Chú ý 2: Trong trường hợp chồng lấp 2 bit và 3 bit, các đánh giá đối với mẫu 5 bit là phức tạp hơn do đó chúng tôi chưa đưa ra được kết quả như các trường hợp trên.

Đối với mẫu 5 bit, khi xét chuỗi độ dài 128 bit chúng tôi thu được kết quả như sau:

BẢNG 3. CÁC XÁC SUẤT KHOẢNG CHO KIỂM TRA SO KHỚP MẪU 5 BIT VỚI CHUỖI 128 BIT

0 bit		**1 bit**	
0-2	0.1908228036	0-2	0.2135889951
3	0.2103926635	3	0.2031503314
4	0.2305286917	4	0.2125640475
5	0.1835219667	5	0.1700549649
6-128	0.1847338745	6-128	0.2006416611

2 bit	**3 bit**	**4 bit**	
Chưa xác định	Chưa xác định	0-1	0.2434158937
		2-3	0.2732075106
		4-5	0.2120313799
		6-8	0.1758805832
		9-128	0.0954646326

Giả sử chúng ta muốn kiểm tra chuỗi nhị phân có độ dài n sử dụng kiểm tra so khớp mẫu, có thể tóm tắt thủ tục như sau:

- Chọn mẫu 5 bit B (không thuộc 2 lớp chồng lấp 2, 3-bit).
- Chia chuỗi thành M khối 128 bit:

$$(M = \left\lfloor \frac{n}{128} \right\rfloor).$$

- Với mỗi khối, viết 3 bit đầu tiên vào cuối khối.
- Tìm sự xuất hiện của B trong khối đầu tiên theo cách chồng lấp và tăng giá trị khoảng tương ứng thêm 1, ký hiệu là F_i, $1 \leq i \leq 5$. Lặp lại cho tất cả các khối.
- Áp dụng kiểm tra so khớp χ^2; tức là tính

$$\chi^2 = \sum_{i=1}^{5} \frac{(F_i - M.p_i)^2}{M.p_i}$$

và

$$p - value = igamc(2, \frac{\chi^2}{2})$$

trong đó p_i nhận được từ bảng 3 theo số các bit lặp trong mẫu B.

- Nếu $p-value < 0.01$ kết luận chuỗi không ngẫu nhiên, ngược lại kết luận chuỗi ngẫu nhiên.

Vì giá trị mong muốn trong mỗi khoảng nhỏ nhất là 5, và xác suất bé nhất trong bảng 3 là 0.0954646326, đối tượng chuỗi để kiểm tra cần có ít nhất $128 \times \left\lfloor \frac{5}{0.0954646326} \right\rfloor = 6704$ bit.

V. MỘT SỐ KẾT QUẢ THỰC NGHIỆM

Chúng tôi đã thực hiện kiểm chứng lại các giá trị xác suất trong Bảng 1 sử dụng công cụ Magma. Kết quả các giá trị xác suất tính được trùng khớp với các giá trị xác suất đưa ra trong [2].

Thêm vào đó, chúng tôi đã lập trình thực thi kiểm tra so khớp mẫu 4 bit sử dụng ngôn ngữ C++. Đồng thời thực hiện kiểm tra đối với 10 file dữ liệu giả ngẫu nhiên. Mỗi file có kích cỡ 10^6 bytes. Mỗi file dữ liệu sẽ được chia thành *m* dãy con có độ dài *n* bit. Sau đó sử dụng kiểm tra tỷ lệ dãy vượt qua tiêu chuẩn để khẳng định dữ liệu là ngẫu nhiên hay không ngẫu nhiên. Cụ thể, chúng tôi số lượng các dãy trong mẫu mà có giá trị *p-value* $\geq \alpha$ và ký hiệu là m_p. Khi đó, dưới giả thiết về tính ngẫu nhiên, m_p tuân theo phân phối nhị thức $\mathcal{B}(m,1-\alpha)$ là xấp xỉ theo phân phối chuẩn $\mathcal{N}\left(m(1-\alpha), m\alpha(1-\alpha)\right)$ khi *n* đủ lớn. Do đó, tỷ lệ dãy vượt qua một kiểm tra $(= m_p / m)$ xấp xỉ theo $\mathcal{N}\left((1-\alpha), \frac{\alpha(1-\alpha)}{m}\right)$. Khoảng chấp nhận được của m_p / m được xác định sử dụng mức ý nghĩa như sau:

$$1-\alpha-3\sqrt{\frac{\alpha(1-\alpha)}{m}} < \frac{m_p}{m} < 1-\alpha+3\sqrt{\frac{\alpha(1-\alpha)}{m}}.$$

Nếu tỷ lệ dãy vượt qua nằm ngoài khoảng trên thì có bằng chứng xác định dữ liệu là không ngẫu nhiên.

Kết quả thu được khi chọn mẫu *B* chồng lấp 1 bit, ngưỡng $\alpha = 0.01$ và độ dài mỗi dãy là $n = 10000$ bit như sau:

FILE	TOTAL	PASS	RATE	GHI CHÚ
RAND01.bin	800	793	99.13%	ĐẠT
RAND02.bin	800	793	99.13%	ĐẠT
RAND03.bin	800	792	99.00%	ĐẠT
RAND04.bin	800	793	99.13%	ĐẠT
RAND05.bin	800	791	98.88%	ĐẠT
RAND06.bin	800	791	98.88%	ĐẠT
RAND07.bin	800	795	99.38%	ĐẠT
RAND08.bin	800	795	99.38%	ĐẠT
RAND09.bin	800	790	98.75%	ĐẠT
RAND10.bin	800	791	98.88%	ĐẠT

Khi chọn mẫu *B* chồng lấp 2 bit, ngưỡng $\alpha = 0.01$ và độ dài mỗi dãy là $n = 5504$ bit, kết quả thu được như sau:

FILE	TOTAL	PASS	RATE	GHI CHÚ
RAND01.bin	1453	1443	99.31%	ĐẠT
RAND02.bin	1453	1435	98.76%	ĐẠT
RAND03.bin	1453	1444	99.38%	ĐẠT
RAND04.bin	1453	1441	99.17%	ĐẠT
RAND05.bin	1453	1438	98.97%	ĐẠT
RAND06.bin	1453	1438	98.97%	ĐẠT
RAND07.bin	1453	1434	98.69%	ĐẠT
RAND08.bin	1453	1435	98.76%	ĐẠT
RAND09.bin	1453	1437	98.90%	ĐẠT
RAND10.bin	1453	1442	99.24%	ĐẠT

Chúng tôi cũng đã thực hiện kiểm tra các file dữ liệu trên sử dụng các kiểm tra 4 bit mới đề xuất. Khi chọn mẫu *B* chồng lấp 1 bit, ngưỡng $\alpha = 0.01$ và độ dài mỗi dãy là $n = 10000$ bit, kết quả thu được như sau:

FILE	TOTAL	PASS	RATE	GHI CHÚ
RAND01.bin	800	797	99.63%	ĐẠT
RAND02.bin	800	794	99.25%	ĐẠT
RAND03.bin	800	789	98.63%	ĐẠT
RAND04.bin	800	795	99.38%	ĐẠT
RAND05.bin	800	790	98.75%	ĐẠT
RAND06.bin	800	794	99.25%	ĐẠT
RAND07.bin	800	792	99.00%	ĐẠT
RAND08.bin	800	796	99.50%	ĐẠT
RAND09.bin	800	785	98.13%	ĐẠT
RAND10.bin	800	790	98.75%	ĐẠT

Khi chọn mẫu B chồng lấp 2 bit, ngưỡng $\alpha = 0.01$ và độ dài mỗi dãy là $n = 5504$ bit, kết quả thu được như sau:

FILE	TOTAL	PASS	RATE	GHI CHÚ
RAND01.bin	1453	1432	99.55%	ĐẠT
RAND02.bin	1453	1440	99.11%	ĐẠT
RAND03.bin	1453	1439	99.04%	ĐẠT
RAND04.bin	1453	1439	99.04%	ĐẠT
RAND05.bin	1453	1440	99.11%	ĐẠT
RAND06.bin	1453	1434	98.69%	ĐẠT
RAND07.bin	1453	1438	98.97%	ĐẠT
RAND08.bin	1453	1441	99.17%	ĐẠT
RAND09.bin	1453	1446	99.52%	ĐẠT
RAND10.bin	1453	1440	99.11%	ĐẠT

Khi chọn mẫu B chồng lấp 3 bit, ngưỡng $\alpha = 0.01$ và độ dài mỗi dãy là $n = 3726$ bit, kết quả thu được như sau:

FILE	TOTAL	PASS	RATE	GHI CHÚ
RAND01.bin	2150	2129	99.02%	ĐẠT
RAND02.bin	2150	2121	98.65%	ĐẠT
RAND03.bin	2150	2123	98.74%	ĐẠT
RAND04.bin	2150	2127	98.93%	ĐẠT
RAND05.bin	2150	2124	98.79%	ĐẠT
RAND06.bin	2150	2118	98.51%	ĐẠT
RAND07.bin	2150	2127	98.93%	ĐẠT
RAND08.bin	2150	2133	99.21%	ĐẠT
RAND09.bin	2150	2138	99.44%	ĐẠT
RAND10.bin	2150	2122	98.70%	ĐẠT

Để kiểm tra độ nhạy của kiểm tra so khớp mẫu 4 bit mới, chúng tôi sử dụng mô hình Markov bậc nhất để mô phỏng một nguồn sinh dữ liệu không ngẫu nhiên. Cụ thể, cho $\{s_n\}_{n\in\mathrm{N}}$ là một quá trình Markov bậc nhất của $\{0,1\}$, có ma trận chuyển là $\begin{pmatrix} p & 1-p \\ 1-p & p \end{pmatrix}$, trong đó $p > 0.5$.

Dễ thấy rằng các mẫu sinh bởi mô hình này không có phân bố đồng xác suất, do vậy đây là một nguồn không ngẫu nhiên. Chúng tôi đã sử dụng phần mềm Matlab để mô phỏng lại quá trình Markov trên và tạo ra các file dữ liệu có kích cỡ 10^6 byte.

Khi chọn mẫu B chồng lấp 3 bit, ngưỡng $\alpha = 0.01$ và độ dài mỗi dãy là $n = 3726$ bit, kết quả thu được như sau:

FILE	TOTAL	PASS	RATE	GHI CHÚ
data0.55.bin	2150	155	7.21%	KHÔNG ĐẠT
data0.6.bin	2150	2	0.09%	KHÔNG ĐẠT
data0.95.bin	2150	0	0.00%	KHÔNG ĐẠT

Từ kết quả trên cho thấy các kiểm tra mới có khả năng phát hiện dữ liệu không ngẫu nhiên tốt.

KẾT LUẬN

Trong bài báo này, dựa trên các kết quả về các tiêu chuẩn so khớp mẫu 4 bit trong [2], chúng tôi đã đề xuất một số kiểm tra so khớp mẫu 4 bit mới và đưa ra một số kết quả lý thuyết cho các mẫu 5 bit. Trong đó, chúng tôi đã đề xuất các tiêu chuẩn so khớp mẫu 4 bit mới. Ưu điểm của các tiêu chuẩn này là có thể áp dụng đối với các chuỗi chỉ cần độ dài 3726 bit. Hơn nữa, chúng tôi cũng xem xét đề xuất các tiêu chuẩn so khớp mẫu 5 bit, tuy nhiên trong trường hợp này chỉ dừng lại ở các kết quả lý thuyết cũng như thực hành cho đánh giá phân bố của 3 lớp: không chồng lấp, chồng lấp 1 bit và chồng lấp 4 bit. Công việc tiếp theo là nghiên cứu tiếp 2 lớp còn lại của các mẫu 5 bit: chồng lấp 2 bit và chồng lấp 3 bit.

TÀI LIỆU THAM KHẢO

[1] Bassham III, L.E., et al., *SP 800-22 Rev. 1a.* A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications, National Institute of Standards & Technology, Gaithersburg, MD, 2010.

[2] Sulak, F., *New statistical randomness tests: 4-bit template matching tests.* Turkish Journal of Mathematics, 41(1), 2017, p. 80-95.

[3] Oorschot, P.v., Vanstone, S.A., and MENEZES, A., *Handbook of applied cryptography*, CRC press, 1997.

[4] Kunuth, D., *The Art of Computer Programming vol. 2 Seminumerical Algorithms*, Reading, Massachusetts: Addison Wesley, 1998.

[5] Caelli, W., *Crypt x package documentation.* Information Security Research Centre School of Mathematics, Queensland University of Technology, 1992.

[6] Marsaglia, G., *The marsaglia random number cdrom including the diehard battery of tests of randomness,* 2008 http://www.stat.fsu.edu/pub/diehard (Accessed 19/8/2021).

[7] L'Ecuyer, P. and Simard, R., *TestU01: AC library for empirical testing of random number generators.* ACM Transactions on Mathematical Software (TOMS), 33(4), 2007, p. 22.

[8] Menezes, A.J., Van Oorschot, P., and Vanstone, S., *Handbook of Applied Cryptography, CRCPress,* Chapter. 5(7), 1996, p. 12.

[9] Soto, J. and Bassham, L., *Randomness testing of the advanced encryption standard finalist candidates*. 2000.

[10] Sulak, F., et al. *Evaluation of randomness test results for short sequences*. in *International Conference on Sequences and Their Applications*. Springer, 2010.

[11] Takeda, Y., *The problem of template matching test in the testing randomness by NIST.* IEICE Technical Report, 2005, p. ISEC2005-110.

[12] Doğanaksoy, A. and Göloğlu, F. *On Lempel-Ziv complexity of sequences*. in *International Conference on Sequences and Their Applications*. Springer, 2006.

[13] Doganaksoy, A. and Tezcan, C. *An alternative approach to Maurer's universal statistical test*. in *Information Security and Cryptology Conference ISC Turkey 2006 3rd International Conference Proceedings*. 2008.

[14] Okutomi, H., *A study on the randomness evaluation method using NIST randomness test.* Proc. SCIS, Jan., 2006.

[15] Hamano, K. and Kaneko, T., *Correction of overlapping template matching test included in NIST randomness test suite.* IEICE transactions on fundamentals of electronics, communications and computer sciences, 2007. 90(9): p. 1788-1792.

[16] Rukhin, A., et al., *Statistical test suite for random and pseudorandom number generators for cryptographic applications, NIST special publication.* 2010.

[17] Charalambides, C.A., *Enumerative combinatorics*., Chapman and Hall/CRC, 1996.

[18] Hoang Đinh Linh, "Some results on new statistical randomness tests based on length of runs", Journal of Science and Technology on Information security, ISSN 2615-9570, Vol. 08, No. 02, 2018, pp. 19-24.

[19] Adrián Alfonso Peñate, Daymé Almeida Echevarria, Laura Castro Argudín, "Statistical Assessment of two Rekeying Mechanisms applied to the Generation of Random Numbers", Journal of Science and Technology on Information Security, ISSN 2615-9570, Vol. 12, No. 02, 2020, pp. 38-44.

SƠ LƯỢC VỀ TÁC GIẢ

Hoàng Đình Linh

Đơn vị công tác: Viện Khoa học – Công nghệ mật mã, Ban Cơ yếu Chính phủ.

Email: hoangdinhlinh@bcy.gov.vn

Quá trình đào tạo: Tốt nghiệp chương trình Toán Tiên Tiến, Trường Đại học Khoa học Tự nhiên, Đại học Quốc gia Hà Nội năm 2014.

Hướng nghiên cứu hiện nay: Nghiên cứu đánh giá các bộ sinh số ngẫu nhiên, nghiên cứu các kiểm tra tính ngẫu nhiên theo thống kê.

Trần Thị Lượng

Đơn vị công tác: Học viện Kỹ thuật mật mã, Ban Cơ yếu Chính Phủ

Email: luongtranhong@gmail.com

Quá trình đào tạo: nhận bằng Cử nhân Toán tin của trường Đại học Khoa học Tự nhiên, Đại học Quốc gia Hà Nội vào năm 2006; nhận bằng Thạc sỹ (năm 2012) và Tiến sỹ (năm 2019) tại Học viện Kỹ thuật mật mã.

Hướng nghiên cứu hiện nay: Mật mã, an toàn cơ sở dữ liệu.