Indonesian Journal of Computer Science
ISSN 2302-4364 (*print*) dan 2549-7286 (*online*)
Jln. Khatib Sulaiman Dalam, No. 1, Padang, Indonesia, Telp. (0751) 7056199, 7058325
Website: ijcs.stmikindonesia.ac.id | E-mail: ijcs@stmikindonesia.ac.id

Identifikasi Nilai Acak Melalui Pemosisian Ulang Fungsi XOR Pada Blok Pertama LSFR A5/1

Ayub Susilo Wibowo[1], Alz Danny Wowo[2]
672019150@student.uksw.edu, alzdanny.wowor@uksw.edu
[1]Universitas Kristen Satya Wacana

Informasi Artikel

Diterima : 14 Des 2023
Direview : 19 Des 2023
Disetujui : 30 Des 2023

Kata Kunci

Linear Feedback Shift Register, Kriptografi, Skema A5/1

Abstrak

Skema A5/1 menghasilkan keacakan dengan menggunakan beberapa register pergeseran umpan balik linier. Fungsi linier pertama diposisikan ulang berdasarkan pemilihan 4 dari 19 bit menggunakan hukum komutatif, dengan paling banyak 60 kali pemosisian ulang. Bit-bit yang dipilih ini menjalani proses iterasi fungsi XOR yang menghasilkan output bit acak maksimum. Hasil akhir dihasilkan oleh XOR yang memproses output dari setiap fungsi linier. Kemampuan algoritma untuk menghasilkan bit keluaran acak diuji secara ekstensif dengan menggunakan metode perhitungan statistik seperti Runs Test, Block Bit, dan Mono Bit untuk mengukur keacakan. Hasilnya secara konsisten menunjukkan bahwa algoritma ini menghasilkan output acak untuk berbagai jenis input. Untuk mengevaluasi kemampuan enkripsi, sepuluh output dipilih dan diuji korelasinya. Sembilan dari keluaran menunjukkan tingkat korelasi yang 'sangat rendah', sementara satu keluaran memiliki tingkat korelasi yang 'rendah'. Hasil ini menunjukkan bahwa desain ini dapat diandalkan sebagai penghasil kunci untuk melindungi informasi.

Keywords

Linear Feedback Shift R Linear Feedback Shift Register, Kriptografi, Skema A5/1

Abstract

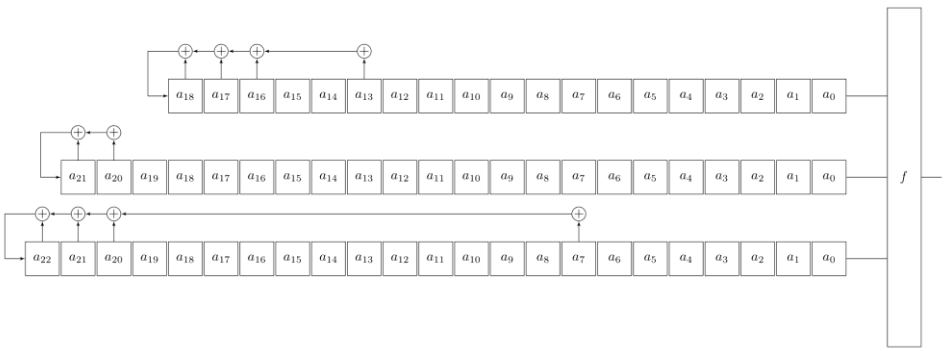
The A5/1 scheme generates randomness by means of multiple linear feedback shift registers. The first linear function is repositioned based on selecting 4 of the 19 bits using the commutative law, with at most 60 repositioning events. These selected bits undergo an XOR function iteration process that produces the maximum random bit output. The final result is generated by XOR processing the output of each linear function. The algorithm's ability to generate random output bits was extensively tested using statistical calculation methods such as Runs Test, Block Bit, and Mono Bit to measure randomness. The results consistently showed that the algorithm generates random outputs for different types of inputs. To evaluate the encryption capability, ten outputs were selected and tested for correlation. Nine of the outputs showed a 'very low' level of correlation, while one output had a 'low' level of correlation. These results suggest that the design is reliable as a key generator for protecting information.

A. Pendahuluan

Kualitas algoritma diukur berdasarkan kompleksitas ruang dan waktu [1]. Pengoptimalan ini memungkinkan proses *enkripsi* dan *deskripsi* yang efisien yang dapat menangani input acak dan menghasilkan output acak [2]. A5/1 diperkenalkan pada tahun 1991 sebagai sistem keamanan pada GMS (*sistem global untuk komunikasi seluler*) atau jaringan 2G [3][4]. A5/1 adalah kombinasi dari beberapa fungsi LFSR (*linear feedback shift register*). *Output* dari fungsi linear diinisialisasi dengan menggunakan *exclusive-or* (XOR) antara hasil bit input yang digeser, yang menentukan periode maksimum bit acak.

Penelitian [4][5][6][7][8][9][10] membahas tentang kelemahan proses *enkripsi* pada algoritma A5/1, kelemahan diantarnya pada kombinasi inisialisasi, jumlah linier, posisi bit pilihan, nilai korelasi, dan panjang linier. Terdapat beberapa juga penelitian yang melakukan modifikisi terhadap kelemahan [11][12][13][14][15][16]. Tujuan dari penelitian ini adalah untuk melakukan reposisi dari A5/1 sebelumnya melalui pengaturan ulang bit-bit input pada blok pertama. Pengujian keacakan digunakan sebagai tolak ukur untuk membedakan hasil keluaran yang diperoleh dari setiap susunan ulang dalam menghasilkan hasil keacakan yang optimal. Selain itu, pengujian enkripsi dilakukan pada sepuluh bit keluaran teratas yang dihasilkan dari proses penyusunan ulang pada A5/1, untuk membandingkannya dengan persamaan 1 dan menentukan metodologi apakah ada penyusunan ulang yang lebih unggul.

$$A1 = a13 \oplus a16 \oplus a17 \oplus a18 \quad \textbf{(1)}$$

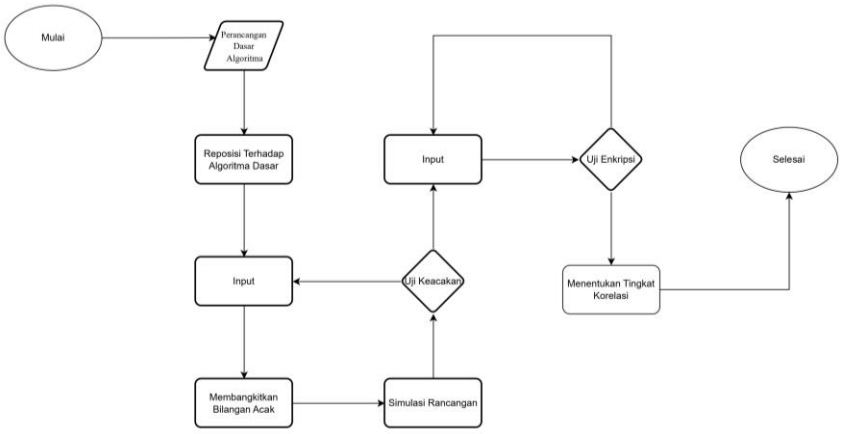


Gambar 1. Tiga Fungsi Linier

Secara keseluruhan, 19-bit terlibat dalam perhitungan, dan 4-bit yang dipilih berdasarkan sifat komutatif dari fungsi XOR, melakukan pergeseran pada semua kemungkinan kombinasi 4-bit dengan posisi yang berbeda. Penelitian ini membatasi hingga 60 kemungkinan meskipun terdapat banyak pilihan pemosisian ulang dalam persamaan ${}_4C_{19}$.

B. Metode Penelitian

Penelitian ini menggunakan metodologi eksperimental untuk memperoleh hasil pengujian. Desain algoritma dasar digunakan untuk mendapatkan algoritma baru melalui reposisi. *Microsoft Office Excel* digunakan dalam pengujian statistik dan *enkripsi*, dengan program yang telah dirancang.



Gambar 2. Rancangan Penelitian

Terdapat beberapa tahapan dalam perancangan penelitian. Tahap awal dengan, merancang algoritma LFSR tiga fungsi linear A5/1 berfungsi sebagai dasar reposisi. Pengubahan posisi terjadi pada fungsi linier pertama.

$$T(x, size) = ||x * 10count||, x\ 6 \neq 0 \quad \textbf{(2)}$$

Tahap perpotongan. Proses penghitungan nilai inisialisasi melibatkan pengubahan masukan plaintext menjadi kode ASCII, diikuti dengan pengubahannya menjadi bilangan biner. Bilangan biner yang dihasilkan dibagi dua menjadi bilangan *riil* dan *chaos*, yang kadang-kadang ditemukan dalam bilangan *riil* antara 0 dan 1. Setiap nilai *chaos* dikalikan hingga nilai keluaran yang diinginkan

tercapai, dan selanjutnya, hasilnya dipartisi, dan hanya komponen bilangan *riil* yang dipertahankan. Nilai *chaos* x yang dikonversi menggunakan persamaan 2, dimana count dimulai dari 1 dan bertambah 1 sampai $x \times 10^{count} > 10^{size-1}$. Hasilnya kemudian diambil bagian integer saja.

Tahap pengujian analisis statistik bertujuan untuk mendapatkan nilai keluaran acak. Tahap ini melibatkan penggunaan berbagai teknik seperti *Mono Bit*, *Block Bit*, dan *Runs Test*.

Mono Bit digunakan untuk menguji keacakan urutan biner. Untuk menerapkan tes ini, $Xi = 2\varepsilon_i - 1$.

$$s_n = x_1 + x_2 + \cdots + x_n \quad \textbf{(3)}$$

Uji statistik

$$S_{obs} = \frac{(|S_n|)}{\sqrt{n}} \quad \textbf{(4)}$$

P-value dihitung dengan menggunakan $P - value = erfc\left(\frac{S_{obs}}{\sqrt{2}}\right)$ digunakan dalam menghitung p-value, dimana erfc komponen error dengan definisi

$$p - value = \boldsymbol{erfc}\frac{2}{\sqrt{\pi}}\int_z^{\infty} e^{-u^2}\, du \quad \textbf{(5)}$$

Kriteria keacakan ditentukan oleh nilai P-value jika kurang dari 0,01, dalam hal ini baris dianggap tidak acak, sedangkan nilai P-value yang lebih besar dari 0,01 mengindikasikan bahwa baris tersebut acak.

Block Bit Test adalah teknik yang menilai keacakan dengan mengukur frekuensi satu blok M-Bit, yang terdiri dari $M/2$. bagian. Frekuensi blok (M, n) mengacu pada prosedur di mana M adalah panjang blok dan n adalah panjang string bit. Nilai ε mewakili frekuensi dari setiap hasil yang mungkin terjadi. Pengujian ini menghasilkan ukuran $x^{2}{}_{obs}$ yang merepresentasikan rasio kecocokan dengan M-Bit yang diberikan $(1/2)$, dengan distribusi x^2 yang digunakan sebagai distribusi referensi untuk statistik pengujian. Deskripsi pengujian melakukan pemisahkan urutan input menjadi $N = \left\lceil \frac{n}{M} \right\rceil$ pada blok yang tidak tumpang tindih dan membuang bit yang tidak digunakan. Gunakan persamaan untuk menentukan proposisi πi dalam setiap blok M-bit. Dengan $1 \leq i \leq N$.

$$\pi = \frac{\sum_{M}^{j=i} \varepsilon(i\ -\ 1)M\ +\ j}{M} \quad \textbf{(6)}$$

Uji statistik

$$\chi^{2}{}_{obs} = 4\sum_{m}^{i=1} \left(\pi_i - \frac{1}{2}\right) \quad \textbf{(7)}$$

$$P - value\ =\ \boldsymbol{igamc}\left(\frac{N}{2}, \frac{\chi 2\ (obs)}{2}\right) \quad \textbf{(8)}$$

Dan $igamc\ =\ \tau\ (z)\ =\ \int_0^{\infty} t^{\,z-1}\ \ e^{-1} dt$ berfungsi sebagai kriteria keacakan dalam menghitung p-value.

Runs Test adalah fungsi test dengan parameter n yang mewakili panjang string bit dan ε yang mewakili urutan bit dari hasil pengujian. Struktur fungsi umumnya adalah $\varepsilon\ =\ \varepsilon 1, \varepsilon 2, \ldots, \varepsilon n$.

Uji statistik dengan $Vn(obs)$ Jumlah total run, yang terdiri dari jumlah total angka nol dan satu dalam semua $n - bit$, dihitung. Distribusi referensi statistik uji adalah distribusi x^2. Untuk menghitung π dari urutan satu bit, gunakan rumus:

$$\pi = \frac{\sum_J \varepsilon_J}{n} \quad \textbf{(9)}$$

Jika nilai absolut dari perbedaan antara π dan $(1/2$ sama dengan atau lebih besar dari τ, tes Frekuensi "Mono Bit" tidak lulus, dan oleh karena itu tes tidak boleh dilakukan. Jika pengujian tidak dapat dilakukan, nilai P ditetapkan ke 0,0000. Penting untuk dicatat bahwa

$$\tau = \frac{2}{\sqrt{n}} \quad \textbf{(10)}$$

sudah didefinisikan dalam kode pengujian. Uji statistika

$$V_{n\,obs} = \sum_{k=1}^{n-1} r(k)\ +\ 1 \quad \textbf{(11)}$$

V_n(obs) dihitung dengan menjumlahkan nilai r(k) + 1 dari k = 1 hingga n-1, di mana r(k) sama dengan 0 jika εk = εk+1 dan 1 jika tidak. Untuk menentukan p-value menggunakan rumus.

$$= \boldsymbol{erfc}\frac{(|Vn(obs) - 2n\pi(1 - \pi)|)}{2\sqrt{2n\pi(1 - \pi)}} \quad \textbf{(12)}$$

Uji Korelasi digunakan untuk mengetahui hubungan antara variabel bebas x dan variabel terikat y. Uji korelasi menggunakan rumus sebagai berikut:

$$r = \frac{n\sum xy - \sum x \sum y}{\sqrt{(nx^2 - (\sum x)^2)) - (ny^2 - (\sum y)^2))}} \quad \textbf{(13)}$$

Koefisien korelasi referensi dapat dihitung sebagai

Tabel 1. Tingkat Korelasi

Interval	Tingkat Hubungan
0,00 – 0,19	Sangat Kecil
0,20 – 0,39	Kecil
0,40 – 0,59	Cukup
0,60 – 0,79	Kuat
0,80 – 1,00	Sangat Kuat

Tabel tersebut merupakan referensi untuk algoritma untuk data yang akan disembunyikan ”enkripsi”. Batas antar interval pada korelasi $-1 \leq r \leq 1$, nilai interval akan lebih baik jika mendekati 0, baik interval tersebut bernilai negatif maupun positif. Korelasi negatif dapat diabaikan, karena nilai negatif tidak memiliki dampak. Penilaian korelasi mengukur seberapa jauh nilai interval dari 0, sesuai dengan persamaan $-1 \leq r \leq 1$ Hasil dan pembahasan.

C. Hasil dan Pembahasan

Algoritma A5/1 menggunakan operasi XOR pada setiap fungsi umpan balik untuk menentukan nilai bit baru pada iterasi berikutnya. Setiap fungsi umpan balik menggunakan sejumlah masukan yang berbeda, dan ketiga fungsi tersebut merupakan input dalam fungsi utama f. A_i merepresentasikan setiap fungsi umpan balik, dan dinyatakan dalam Persamaan 13 untuk setiap $i = \{1,2,3\}$. Selain itu, bahasa yang digunakan adalah bahasa formal dan mengikuti kebenaran tata bahasa dan konvensi yang telah ditetapkan. Model ini menggunakan XOR untuk regularisasi dan menerapkan fungsi yang sama pada setiap iterasi. Keluaran dari iterasi awal f_i bertindak sebagai masukan untuk fungsi utama f.

$$\begin{gathered} A1 = a13 \oplus a16 \oplus a17 \oplus a18 \\ A2 = a20 \oplus a21 \\ A3 = a20 \oplus a21 \oplus a22 \end{gathered} \quad (13)$$

Hasil akhirnya dihitung sebagai

$$f = A1 \oplus A2 \oplus A3 \quad (14)$$

Fungsi utama yang menggabungkan setiap fungsi umpan balik fi disajikan pada Persamaan 13, yang berfungsi sebagai dasar untuk proses reposisi pada linear awal A_1. Tabel 2 merupakan hasil reposisi.

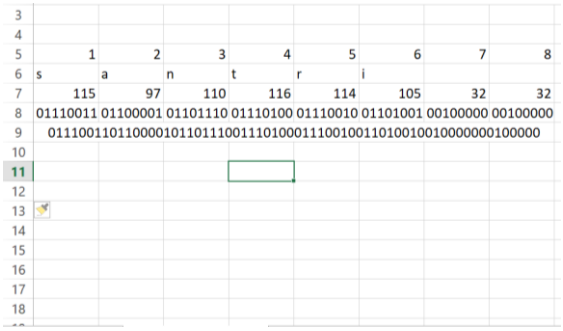
Tabel 2. Hasil Reposisi

No	Reposisi	Algoritma Reposisi
1	LFSR BE	a13 ⊕ a16 ⊕ a17 ⊕ a15
2	LFSR BF	a13 ⊕ a16 ⊕ a17 ⊕ a14
3	LFSR BH	a13 ⊕ a16 ⊕ a17 ⊕ a12
4	LFSR BI	a13 ⊕ a16 ⊕ a17 ⊕ a11
5	LFSR BJ	a13 ⊕ a16 ⊕ a17 ⊕ a10
6	LFSR BK	a13 ⊕ a16 ⊕ a17 ⊕ a9
7	LFSR BL	a13 ⊕ a16 ⊕ a17 ⊕ a8
8	LFSR BM	a13 ⊕ a16 ⊕ a17 ⊕ a7
9	LFSR BN	a13 ⊕ a16 ⊕ a17 ⊕ a6
10	LFSR BO	a13 ⊕ a16 ⊕ a17 ⊕ a5
11	LFSR BP	a13 ⊕ a16 ⊕ a17 ⊕ a4
12	LFSR BQ	a13 ⊕ a16 ⊕ a17 ⊕ a3
13	LFSR BR	a13 ⊕ a16 ⊕ a17 ⊕ a2
14	LFSR BS	a13 ⊕ a16 ⊕ a17 ⊕ a1
15	LFSR BT	a13 ⊕ a16 ⊕ a17 ⊕ a0
16	LFSR CE	a13 ⊕ a16 ⊕ a18 ⊕ a15
17	LFSR CF	a13 ⊕ a16 ⊕ a18 ⊕ a14
18	LFSR CH	a13 ⊕ a16 ⊕ a18 ⊕ a12
19	LFSR CI	a13 ⊕ a16 ⊕ a18 ⊕ a11
20	LFSR CJ	a13 ⊕ a16 ⊕ a18 ⊕ a10
21	LFSR CK	a13 ⊕ a16 ⊕ a18 ⊕ a9
22	LFSR CL	a13 ⊕ a16 ⊕ a18 ⊕ a8
23	LFSR CM	a13 ⊕ a16 ⊕ a18 ⊕ a7
24	LFSR CN	a13 ⊕ a16 ⊕ a18 ⊕ a6
25	LFSR CO	a13 ⊕ a16 ⊕ a18 ⊕ a5

26	LFSR CP	a13 ⊕ a16 ⊕ a18 ⊕ a4
27	LFSR CQ	a13 ⊕ a16 ⊕ a18 ⊕ a3
28	LFSR CR	a13 ⊕ a16 ⊕ a18 ⊕ a2
29	LFSR CS	a13 ⊕ a16 ⊕ a18 ⊕ a1
30	LFSR CT	a13 ⊕ a16 ⊕ a18 ⊕ a0
31	LFSR DE	a13 ⊕ a17 ⊕ a18 ⊕ a15
32	LFSR DF	a13 ⊕ a17 ⊕ a18 ⊕ a14
33	LFSR DH	a13 ⊕ a17 ⊕ a18 ⊕ a12
34	LFSR DI	a13 ⊕ a17 ⊕ a18 ⊕ a11
35	LFSR DJ	a13 ⊕ a17 ⊕ a18 ⊕ a10
36	LFSR DK	a13 ⊕ a17 ⊕ a18 ⊕ a9
37	LFSR DL	a13 ⊕ a17 ⊕ a18 ⊕ a8
38	LFSR DM	a13 ⊕ a17 ⊕ a18 ⊕ a7
39	LFSR DN	a13 ⊕ a17 ⊕ a18 ⊕ a6
40	LFSR DO	a13 ⊕ a17 ⊕ a18 ⊕ a5
41	LFSR DP	a13 ⊕ a17 ⊕ a18 ⊕ a4
42	LFSR DQ	a13 ⊕ a17 ⊕ a18 ⊕ a3
43	LFSR DR	a13 ⊕ a17 ⊕ a18 ⊕ a2
44	LFSR DS	a13 ⊕ a17 ⊕ a18 ⊕ a1
45	LFSR DT	a13 ⊕ a17 ⊕ a18 ⊕ a0
46	LFSR GE	a16 ⊕ a17 ⊕ a18 ⊕ a15
47	LFSR GF	a16 ⊕ a17 ⊕ a18 ⊕ a14
48	LFSR GH	a16 ⊕ a17 ⊕ a18 ⊕ a12
49	LFSR GI	a16 ⊕ a17 ⊕ a18 ⊕ a11
50	LFSR GJ	a16 ⊕ a17 ⊕ a18 ⊕ a10
51	LFSR GK	a16 ⊕ a17 ⊕ a18 ⊕ a9
52	LFSR GL	a16 ⊕ a17 ⊕ a18 ⊕ a8
53	LFSR GM	a16 ⊕ a17 ⊕ a18 ⊕ a7
54	LFSR GN	a16 ⊕ a17 ⊕ a18 ⊕ a6
55	LFSR GO	a16 ⊕ a17 ⊕ a18 ⊕ a5

56	LFSR GP	a16 ⊕ a17 ⊕ a18 ⊕ a4
57	LFSR GQ	a16 ⊕ a17 ⊕ a18 ⊕ a3
58	LFSR GR	a16 ⊕ a17 ⊕ a18 ⊕ a2
59	LFSR GS	a16 ⊕ a17 ⊕ a18 ⊕ a1
60	LFSR GT	a16 ⊕ a17 ⊕ a18 ⊕ a0

Hasil dari reposisi akan menjadi bahan untuk pengujian statistik. Proses dimulai dengan memasukkan "plaintext" dan mengubahnya menjadi angka biner. Terdapat batas 8 karakter dan 30 pengulangan pengujian. Untuk menghitung nilai inisialisasi, input (plaintext) pertama-tama dikonversi ke kode ASCII dan kemudian ke angka biner. Input dengan kurang dari 8 karakter diisi dengan spasi secara otomatis, sedangkan karakter setelah 8 akan diabaikan. Salah satu input yang digunakan adalah "santri" dengan hasil konversi: 01110011011000010110111001110100011100100110100100100000000100000. Titik perpotongan berfungsi sebagai faktor pembatas untuk setiap fungsi linier sesuai dengan Persamaan 2. A_1 memiliki panjang 19-bit dengan nilai awal 0111001101100001011, A_2 memiliki panjang 22-bit dengan nilai awal 0111001110100011100100, dan A_3 memiliki panjang 23-bit dengan nilai awal 11010010010000000100000.



Gambar 3. Proses Pembangkitan Bilangan

Penelitian ini menggunakan berbagai metode pengujian statistik, yang meliputi Mono Bit, Block Bit, dan Runs Test. Metode-metode ini kemudian dirata-ratakan dan dinilai untuk menentukan apakah input menghasilkan output yang acak. Jika output tidak acak terjadi, input diganti dan penyesuaian ini memengaruhi semua hasil reposisi. Sebuah hasil diidentifikasi sebagai acak jika dua atau lebih output dari statistik memiliki nilai $a \geq 0{,}01$. Sepuluh hasil rata-rata teratas dianalisis untuk uji korelasi.

Tabel 3. Rata-Rata

No	Reposisi	P-Value	Hasil Uji
1	LFSR DT	0,702482743	ACAK
2	LFSR GE	0,695395594	ACAK

3	LFSR BJ	0,6952564	ACAK
4	LFSR BT	0,691000465	ACAK
5	LFSR DS	0,677326409	ACAK
6	LFSR GJ	0,676968528	ACAK
7	LFSR GH	0,673968235	ACAK
8	LFSR DH	0,67339333	ACAK
9	LFSR BE	0,668819915	ACAK
10	LFSR BL	0,668256721	ACAK
	Rata-rata	0,677386416	ACAK

Berdasarkan tabel 3 uji statistik, sepuluh pemain teratas memiliki skor rata-rata. Dari jumlah tersebut, "LFSR DT" menempati urutan tertinggi dengan p-value rata-rata 0,702482743. Namun, hanya lima dari sepuluh algoritma ini yang berkinerja lebih baik daripada algoritma dasar "LFSR" dengan p-value 0,677386416 Meskipun demikian, sepuluh pemain teratas dapat menghasilkan output acak.

Tabel 4. Uji Mono Bit

No	Reposisi	P-Value	Hasil Uji
1	LFSR DT	0,516623612	ACAK
2	LFSR GE	0,547279036	ACAK
3	LFSR BJ	0,472483225	ACAK
4	LFSR BT	0,449312969	ACAK
5	LFSR DS	0,481586426	ACAK
6	LFSR GJ	0,441401858	ACAK
7	LFSR GH	0,425111422	ACAK
8	LFSR DH	0,439101814	ACAK
9	LFSR BE	0,442599184	ACAK
10	LFSR BL	0,486490496	ACAK
	Rata-rata	0.470199004	ACAK

Tabel 4 menyajikan hasil keluaran untuk uji Mono Bit dari sepuluh teratas. Nilai p-value optimal sebesar 0,547279036 terjadi selama reposisi "LFSR GE". Rata-rata dari kesepuluh reposisi adalah 0,470199004.

Tabel 5. Uji Block Bit

No	Reposisi	P-Value	Hasil Uji
1	LFSR DT	1,000000000	ACAK
2	LFSR GE	0,999873233	ACAK
3	LFSR BJ	0,999999109	ACAK
4	LFSR BT	0,999999864	ACAK
5	LFSR DS	0,999999808	ACAK
6	LFSR GJ	1,000000000	ACAK
7	LFSR GH	1,000000000	ACAK
8	LFSR DH	0,999999996	ACAK
9	LFSR BE	1,000000000	ACAK
10	LFSR BL	1,000000000	ACAK
	Rata-rata	0.999986982	ACAK

Tabel 5 menyajikan hasil keluaran untuk uji Block Bit dari sepuluh teratas di dapatkan rata-rata 0,470199004.

Tabel 6. Uji Runs Test

No	Reposisi	P-Value	Hasil Uji
1	LFSR DT	0,590824617	ACAK
2	LFSR GE	0,539034514	ACAK
3	LFSR BJ	0,613286866	ACAK
4	LFSR BT	0,623688563	ACAK
5	LFSR DS	0,573068475	ACAK
6	LFSR GJ	0,59075739	ACAK
7	LFSR GH	0,606867806	ACAK
8	LFSR DH	0,591803774	ACAK
9	LFSR BE	0,57930552	ACAK
10	LFSR BL	0,533689494	ACAK
	Rata-rata	0,5842332702	ACAK

Tabel 6menyajikan hasil keluaran untuk uji Runs Test dari sepuluh teratas. Nilai p-value optimal sebesar 0,623688563 terjadi selama reposisi "LFSR BT". Rata-rata dari kesepuluh reposisi adalah 0,5842332702.

Uji enkripsi menggunakan batasan ”ciphertext” sepanjang 32 karakter. Operasi ini menggunakan modulus 256 dalam menguji enkripsi E_k dan deskripsi D_k. Fungsi enkripsi E_k didefinisikan oleh $E_k : P \oplus K = C$, sedangkan deskripsi fungsi deskripsi D_k didefinisikan oleh $D_k : C \oplus K = P$. Dalam proses pengujian enkripsi, tiga buah cipherteks input digunakan untuk mewakili variasi ciphertext yang biasa digunakan oleh pengguna. Diantaranya "pura-pura akrab", yang menandakan tulisan biasa yang hanya terdiri dari kombinasi huruf. Ciphertext lainnya, " S jeian mo12PK&&nn ˆ OJBBk1s¡sln ", menggunakan kombinasi huruf, simbol, dan angka. Hasil uji korelasi mengevaluasi korelasi antara ciphertext dan plaintext untuk menentukan efektivitas kunci dalam menyembunyikan plaintext dan menghapus hubungan mereka.

Tabel 7. Uji Enkripsi

No	Reposisi	1	2	3	Rata-rata	Hubungan Interval
1	LFSR DT	-0,11	-0,10	-0,08	-0,10	SANGAT KECIL
2	LFSR GE	-0,06	-0,10	-0,06	-0,07	SANGAT KECIL
3	LFSR BJ	0,16	0,15	0,20	0,17	SANGAT KECIL
4	LFSR BT	0,08	0,12	0,10	0,10	SANGAT KECIL
5	LFSR DS	-0,05	-0,09	-0,07	-0,07	SANGAT KECIL
6	LFSR GJ	-0,06	-0,10	-0,07	-0,08	SANGAT KECIL
7	LFSR GH	-0,04	-0,10	-0,11	-0,08	SANGAT KECIL
8	LFSR DH	-0,05	-0,06	-0,06	-0,05	SANGAT KECIL
9	LFSR BE	0,29	0,17	0,26	0,24	KECIL
10	LFSR BL	0,15	0,09	0,16	0,13	SANGAT KECIL

Tabel 7 menggambarkan bahwa interval rata-rata hampir nol dengan tingkat korelasi yang minimal. Kombinasi ”LFSR BE” menunjukkan tingkat korelasi yang rendah dengan nilai interval 0,236302348, sedangkan kombinasi lainnya menunjukkan tingkat korelasi yang tidak signifikan. Oleh karena itu, algoritma ini berhasil dirancang sebagai generator kunci angka acak berbasis A5/1.

Dalam penelitian ini, kami menemukan bahwa algoritma reposisi menghasilkan angka acak dengan tingkat korelasi yang ”sangat kecil” dan ”kecil” . Untuk menguji hal ini, kami menggunakan program yang dibuat di Microsoft Office Excel, yang menggunakan metode statistik seperti Mono Bit, Block Bit, dan Runs Test sebagai penghasil angka acak. Nilai p-value rata-rata untuk setiap algoritme ditemukan a≤ 0,01. Setelah mendapatkan nilai output dari metode pengacakan, kami merata-ratakan metode-metode tersebut dan menguji sepuluh metode terbaik untuk enkripsi.

Pengujian enkripsi melibatkan penggunaan tiga ciphertext yang berbeda, masing-masing menunjukkan tingkat korelasi yang sangat rendah. Algoritma yang digunakan sebagai tolok ukur telah menetapkan tingkat korelasi yang rendah

tersebut. Oleh karena itu, dapat disimpulkan bahwa reposisi menghasilkan korelasi kunci yang tinggi dan keluaran acak yang aman dalam kriptografi.

D. Kesimpulan

Dalam penelitian ini, kami menemukan bahwa algoritma reposisi menghasilkan angka acak dengan tingkat korelasi yang ”sangat kecil” dan ”kecil” . Untuk menguji hal ini, kami menggunakan program yang dibuat di Microsoft Office Excel, yang menggunakan metode statistik seperti Mono Bit, Block Bit, dan Runs Test sebagai penghasil angka acak. Nilai p-value rata-rata untuk setiap algoritme ditemukan $a \leq$ 0,01. Setelah mendapatkan nilai output dari metode pengacakan, kami merata-ratakan metode-metode tersebut dan menguji sepuluh metode terbaik untuk enkripsi.

Pengujian enkripsi melibatkan penggunaan tiga masukan ciphertext yang berbeda, masing-masing menunjukkan tingkat korelasi yang sangat rendah. Algoritma yang digunakan sebagai tolak ukur telah menetapkan tingkat korelasi yang rendah tersebut. Oleh karena itu, dapat disimpulkan bahwa reposisi menghasilkan korelasi kunci yang tinggi dan keluaran acak yang aman dalam kriptografi.

Referensi

[1] A. D. Wowor and B. Susanto, “One to many (new scheme for symmetric cryptography),” *Telkomnika (Telecommunication Comput. Electron. Control.*, vol. 21, no. 4, pp. 762–770, Aug. 2023, doi: 10.12928/TELKOMNIKA.v21i4.24789.

[2] T. Enkripsi, “Penerapan Keamanan Basis Data Dengan Teknik Enkripsi,” *J. Sist. Inf. Univ. Suryadarma*, vol. 1, no. 1, pp. 12–25, 2014, doi: 10.35968/jsi.v1i1.30.

[3] Magnus Glendrange;, K. Hove, and E. Hvideberg, “Decoding GSM - Master Thesis,” no. June, 2010, [Online]. Available: http://www.diva-portal.org/smash/get/diva2:355716/FULLTEXT01.pdf

[4] O. D. Jensen, “A5 Encryption In GSM,” no. June, pp. 3–8, 2017.

[5] A. Alhamdan, H. Bartlett, E. Dawson, L. Simpson, K. Koon, and H. Wong, “Weak key-IV Pairs in the A5/1 Stream Cipher.”

[6] A. Jain and N. S. Chaudhari, “Two Trivial Attacks on A5/1:A GSM Stream Cipher,” no. Cryptography and Security (cs.CR), 2013, [Online]. Available: http://arxiv.org/abs/1305.6817%5Cnhttp://www.arxiv.org/pdf/1305.6817.pdf

[7] T. Gendrullis, M. Novotný, and A. Rupp, “A real-world attack breaking A5/1 within hours,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2008, pp. 266–282. doi: 10.1007/978-3-540-85053-3_17.

[8] A. Mahalanobis and J. Shah, “An Improved Guess-and-Determine Attack on the A5/1 Stream Cipher,” *Comput. Inf. Sci.*, vol. 7, no. 1, pp. 115–124, 2014, doi: 10.5539/cis.v7n1p115.

[9] C. Xenakis and C. Ntantogian, “Attacking the Baseband Modem of Mobile Phones to Breach the Users ’ Privacy and Network Security”.

[10] S. B. Sadkhan and N. H. Jawad, “Simulink Based Implementation of Developed A5/1 Stream Cipher Cryptosystems,” in *Procedia Computer Science*, Elsevier, 2015, pp. 350–357. doi: 10.1016/j.procs.2015.09.096.

[11] A. Ilmiah Diajukan kepada Fakultas Teknologi Informasi untuk memperoleh

gelar Sarjana Komputer Peneliti, A. Julian Herman, and A. Danny Wowor, "Desain Pembangkit Kunci LFSR dengan Skema A5/1 Menggunakan 7 Blok Bit Fungsi XOR," 2022.

[12] S. Yohana, A. Mohd, M. Othman, F. Masyitah, and M. Shuib, "Randomness Evaluation of Modified A5 / 1 Stream Cipher for Global System for Mobile Communication," *Malysian J. Sci. technoloogy*, vol. 2, pp. 31–34, 2018.

[13] A. J. Herman and A. D. Wowor, "Desain Pembangkit Kunci LFSR dengan Skema A5 / 1 Menggunakan 7 Blok Bit Fungsi XOR Fakultas Teknologi Informasi," no. 672020704, 2022.

[14] R. Elin Thomas, G. Chandhiny, K. Sharma, H. Santhi, and P. Gayathri, "Enhancement of A5/1 encryption algorithm," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 263, no. 4, 2017, doi: 10.1088/1757-899X/263/4/042084.

[15] H. Bahjat and M. Ali, "Improvement Majority Function in A5/1 stream cipher Algorithm," *Eng. Technol. J.*, vol. 34, no. 1, pp. 16–25, 2016, doi: 10.30684/etj.34.1b.2.

[16] N. H. Zakaria, K. Seman, and I. Abdullah, "Modified A5/1 Based Stream Cipher for Secured GSM Communication," *Int. J. Comput. Sci. Netw. Secur.*, vol. 11, no. 2, pp. 223–226, 2011.