p-ISSN : 2407 – 1846
e-ISSN : 2460 – 8416
Website : jurnal.umj.ac.id/index.php/semnastek

ALGORITMA GENETIKA UNTUK PEMBENTUKAN KUNCI MATRIKS 3 X 3 PADA KRIPTOGRAFI HILL CIPHER

Andysah Putera Utama Siahaan
Fakultas Ilmu Komputer, Universitas Pembangunan Panca Budi
Jl. Jend. Gatot Subroto Km. 4,5 Sei Sikambing, 20122, Medan, Sumatera Utara, Indonesia
E-mail : andiesiahaan@gmail.com

ABSTRAK

Kunci enkripsi Hill Cipher dapat menetukan apakah enkripsi dan dekripsi berhasil dilakukan. Hal ini disebabkan tidak sembarang kunci yang dapat diberikan pada matriks yang digunakan untuk proses tersebut. Kunci yang memiliki determinan yang tidak sesuai tidak dapat dimanfaatkan dalam proses karena dapat mengakibatkan pesan yang terenkripsi tidak dapat dikembalikan ke pesan aslinya. Algoritma genetika dapat menentukan kunci yang digunakan untuk enkripsi dan dekripsi pada Hill Cipher. Dengan menentukan fungsi evaluasi dalam algoritma genetika, kunci yang memiliki komposisi yang benar akan diperoleh. Untuk menerapkan algoritma ini, pencarian kunci pada Hill Cipher akan mudah dilakukan tanpa menghabiskan terlalu banyak waktu.

Kata Kunci - Cryptography, Algoritma Genetika, Hill Cipher

ABSTRACT

The encryption key Hill Cipher can determine whether encryption and decryption is successful. This is due not just a key that can be given to the matrix used for the process. The key that has no corresponding determinants can not be utilized in the process because it can lead encrypted message can not be returned to the original message. Genetic algorithms can determine the key used for encryption and decryption on the Hill Cipher. By determining the evaluation function in the genetic algorithm, the key that has the correct composition to be obtained. To implement this algorithm, the search key on the Hill Cipher will be easily done without spending too much time.

Keywords - Cryptography, Genetic Algorithm, Hill Cipher

I. PENDAHULUAN

Enkripsi Hill Cipher adalah cara yang digunakan untuk mengenkripsi pesan menggunakan matriks sebagai kunci [1]. Dalam kunci ini, ada sembilan buah bilangan bulat acak yang mengisi pola matriks 3x3. Setiap nomor akan berinteraksi satu sama lain untuk menghasilkan ciphertext, tapi tidak semua angkat dapat secara permanen mengembalikan pesan yang terenkripsi kembali pada pesan aslinya [2][4]. Angka-angka harus memiliki nilai yang tepat dalam perhitungan determinan. Sebelum angka bisa dimanfaatkan, angka tersebut harus diuji apakah memenuhi aturan determinan yang berlaku. Pengujian ini sendiri membutuhkan waktu yang cukup lama sementara angka-angka ini yang menghasilkan determinan yang benar belum tentu diperoleh. Jika hasilnya adalah salah, pencarian bilangan bulat acak harus dilakukan lagi. Pencarian secara berulang-ulang akan membuang waktu yang sangat lama.

Masalah yang timbul adalah waktu yang tidak efisien jika kunci matriks pada algoritma Hill Cipher harus dilakukan secara manual [3][8]. Algoritma genetika diharapkan dapat meghasilkan kunci matriks Hill Cipher dengan efisien dan efektif. Dengan memanfaatkan dan menggabungkan algoritma genetika, proses enkripsi dan dekripsi akan terlaksana dengan baik dan tidak perlu menghabiskan waktu yang lama [11].

II. TEORI

A. Hill Cipher

Hill Cipher adalah teknik enkripsi yang menggunakan aritmatika modulo dalam

p-ISSN : 2407 – 1846
e-ISSN : 2460 – 8416
Website : jurnal.umj.ac.id/index.php/semnastek

kriptografi [3][9]. Hill Cipher menggunakan kunci simetris sebagai password untuk mengkonversi plaintext ke ciphertext. Kunci simetris adalah salah satu sistem kriptografi yang memiliki jenis yang sama kunci dalam enkripsi dan dekripsi. Kunci yang digunakan untuk enkripsi sebenarnya berbeda dari dekripsi, tetapi kunci tersebu diambil dari rumus yang sama. Kunci yang digunakan pada proses enkripsi harus diinvers sebelum digunakan untuk mendekripsi ciphertext. Teknik kriptografi ini memiliki matriks sebagai tempat dari pertukaran informasi baik enkripsi atau sebagian dekripsi. Teori umum dari matriks yang digunakan pada Hill Cipher adalah perkalian antara matriks dan invers dari matriks. Tanpa mendapatkan kunci yang tepat, proses enkripsi dan dekripsi tidak dapat dilakukan. Proses enkripsi pada Hill Cipher dapat dililhat pada gambar berikut.

$$\begin{bmatrix} C1 \\ C2 \\ C3 \end{bmatrix} = \begin{bmatrix} K11 & K12 & K13 \\ K21 & K22 & K23 \\ K31 & K32 & K33 \end{bmatrix} \begin{bmatrix} P1 \\ P2 \\ P3 \end{bmatrix} mod\ TC$$

Gambar 1 : Enkripsi Hill Cipher

C adalah ciphertext, P adalah plaintext, K adalah kunci dan TC adalah total penggunaan karakter. Nilai C1 diperoleh dari perkalian antara K11, K12, K13 dan P1, P2, P3. Hasil perkalian tersebut akan mengalami modulo terhadap total karakter yang digunakan.

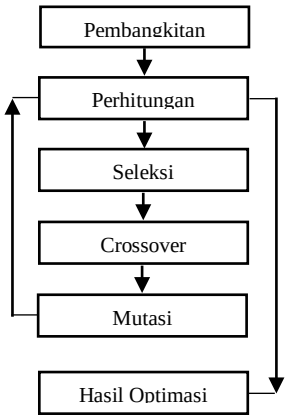
$$\begin{bmatrix} P1 \\ P2 \\ P3 \end{bmatrix} = \begin{bmatrix} Ki11 & Ki12 & Ki13 \\ Ki21 & Ki22 & Ki23 \\ Ki31 & Ki32 & Ki33 \end{bmatrix} \begin{bmatrix} C1 \\ C2 \\ C3 \end{bmatrix} mod\ TC$$

Gambar 2 : Dekripsi Hill Cipher

Gambar 2 memperlihatkan proses dekripsi. Proses tersebut adalah kebalikan dari proses enkripsi. Akan tetapi, pada proses dekripsi ini, nilai dari K harus terlebih dahulu diubah menjadi K inverse dengan ketentuan nilai determinan dari matriks kunci tersebut harus bernilai 1. Jika tidak, ciphertext tidak akan pernah kembali menjadi plaintext awal.

B. Algoritma Genetika

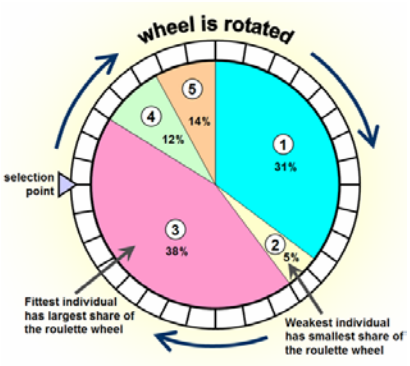
Algoritma genetika adalah algoritma komputasi yang terinspirasi dari teori evolusi yang kemudian diadopsi menjadi algoritma komputer yang kemudian digunakan untuk memecahkan nilai pencarian atau solusi dalam menyelesaikan masalah optimasi [5][6][7]. Algoritma ini dibangun melalui proses genetik pada organisme hidup. Pada gambar berikut merupakan langkah dari proses algoritma genetika.



Gambar 3 : Tahapan algoritma genetika

Seleksi

Ada tiga langkah utama dalam algoritma genetika yaitu seleksi, crossover dan mutasi [10][12]. Seleksi digunakan untuk menggabungkan kembali populasi dengan probabilitas tertinggi. Nomor acak yang dihasilkan dikombinasikan dengan probabilitas kumulatif. Nilai terdekat diambil untuk menggantikan nilai gen dari populasi.



Gambar 4 : Seleksi Roulette Wheel

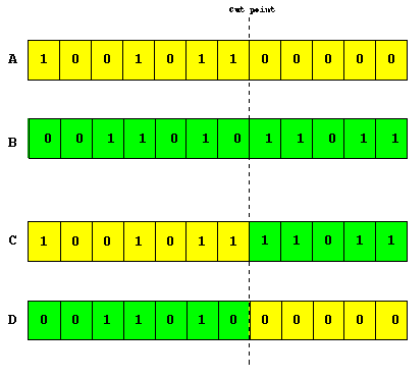
Gambar 4 menunjukkan salah satu metode seleksi. Metode tersebut adalah Roulette Wheel. Individu dipetakan menjadi segmen garis dalam urutan sehingga setiap segmen individu memiliki ukuran yang sama persis seperti fitness-nya. Sebuah nomor acak akan dihasilkan untuk menentukan posisi individu.

p-ISSN : 2407 – 1846
e-ISSN : 2460 – 8416
Website : jurnal.umj.ac.id/index.php/semnastek

Individu yang memiliki nilai probabilitas yang mendekati dengan nilai acak yang dibangkitkan akan berpotensi terpilih menjadi pengganti populasi lain. Proses ini diulang sampai jumlah populasi terselesaikan seluruhnya.

Crossover

Crossover adalah operator algoritma genetika untuk mencampur kromosom dengan kromosom ekstra yang dipilih untuk menghasilkan kromosom anak dari satu generasi ke generasi berikutnya [10][12]. Biasanya teknik crossover ini akan memilih beberapa orang tua yang berkualitas. Kualifikasi adalah nilai tingkat crossover. Nilai ini berkaitan untuk memilih kromosom induk.

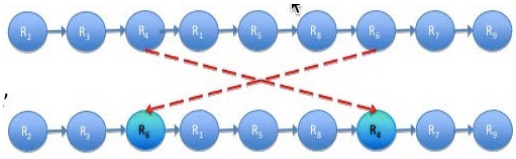


Gambar 5 : One cut point crossover

Gambar 5 menunjukkan crossover berdasarkan satu divisi titik potong. Pada teknik crossover ini akan diambil titik potong yang akan menjadi pertukaran divisi parent dengan child. Kromosom A pada divisi pertama akan ditukar pada Kromosom B divisi pertama. Kromosom C dan D merupakan anak dari pertukaran kedua kromosom tersebut,

Mutasi

Mutasi adalah operator genetik yang digunakan untuk mempertahankan keragaman genetik dari satu generasi populasi kromosom algoritma genetika pada generasi berikutnya [10][12]. Operator ini akan mengubah posisi kromosom dengan bertukar nilai kromosom.



Gambar 6 : Swap Mutation

Gambar 6 menunjukkan contoh swap mutation. Pada ilusterasi tersebut dapat dilihat bahwa R4 dan R6 mengalami pertukaran. Indeks kromosom tidak berubah, tetapi nilai dari indeks akan saling bertukar antara satu dengan yang lainnya. Mutasi menyebabkan perubahan dalam ruang pencarian dan dapat mendekati solusi yang ingin dicapai.

III. METODOLOGI

Kromosom pada matriks kunci Hill Cipher berjumlah sebanyak sembilan kromosom. Setiap gen mempunyai nilai antara 0 sampai 255. Nilai ini merupakan nilai bertipe byte. Nilai-nilai yang diperoleh pada pembangkitan acak akan diletakkan pada tiap-tiap sel pada matriks kunci tersebut.

$$\begin{bmatrix} Ki11 & Ki12 & Ki13 \\ Ki21 & Ki22 & Ki23 \\ Ki31 & Ki32 & Ki33 \end{bmatrix}$$

K11	K12	K13	K21	K22	K23	K31	K32	K33

Gamabr 7 : Pembentukan kromosom pada matriks

Gambar 7 menjelaskan pembentukan kromosom dari kunci matriks Hill Cipher. Kunci yang merupakan matriks 3 x 3 akan berubah menjadi vektor satu dimensi yang mempunyai sembilan buah gen pada tiap kromosomnya.

A. Fitness

Nilai fitness dapat ditentukan setelah seluruh gen telah terisi dengan angka acak. Fitness dapat dicari dengan menentukan nilai determinan. Syarat utama dari nilai ini adalah F = 1 dimana determinan dari matriks kunci tersebut harus memiliki nilai D = 1 juga. Rumus fitness dapat dilihat di bawah ini.

$$F = D \quad (1)$$

p-ISSN : 2407 – 1846
e-ISSN : 2460 – 8416
Website : jurnal.umj.ac.id/index.php/semnastek

Keterangan:

F = Fitness
D = Determinan

IV. HASIL

A. Evaluasi

Pada bagian ini akan dilakukan pengujian untuk mencari nilai-nilai yang tepat yang akan ditempatkan pada ke sembilan sel pada matriks kunci Hill Cipher. Pada inisialisasi awal akan ditentukan beberapa parameter yang menjadi penentu awal.

Generasi = 30
Ukuran Populasi = 20
Tingkat Crossover = 0,8
Tingkat Mutasi = 0,5

Melihat ada 20 buah populasi, akan ada pembangkitan acak sebanyak 20 kali pada tiap populasi sementara pada populasi akan ada 9 kali pembangkitan acak. Sehingga ada 180 buah nilai yang akan dibangkitkan pada tahap pertama ini. Nilai yang tertera pada tabel di bawah ini merupakan hasil dari pembangkitan bilangan acak pada 20 populasi.

Tabel 1 : Initial Population

K11	K12	K13	K21	K22	K23	K31	K32	K33
108	85	165	89	69	185	97	185	54
65	86	135	47	224	116	213	112	6
227	29	41	44	1	141	101	218	32
174	195	136	196	220	37	108	144	43
31	232	46	145	120	234	196	242	63
190	54	140	128	118	179	151	108	43
90	64	85	24	242	106	154	178	244
172	124	121	98	194	81	73	183	135
215	82	163	62	103	13	79	165	164
147	47	77	212	45	112	249	18	80
213	7	244	195	246	197	244	25	119
196	92	115	59	55	190	44	191	27
134	29	216	72	200	78	196	7	131
168	214	80	10	104	177	1	114	177
145	20	91	221	73	79	149	137	73
50	89	246	142	168	108	85	116	244
119	141	61	167	254	239	66	77	65
40	187	243	193	58	195	14	154	172
190	168	210	137	178	63	5	146	173
139	26	47	226	179	242	187	137	228

Populasi yang telah dibangkitkan akan kemudian diterapkan terhadap beberapa searangkaian proses. Perhitungan fitness, probabilitas dan probabilitas kumulatif akan dilakukan untuk menguji seberapa dekat populasi-populasi tersebut dengan solusi yang diinginkan. Pada Tabel 2 dilampirkan hasil perhitungan fitness, probabilitas dan probabilitas kumulatif pada tiap-tiap populasi.

Tabe1 2 : Hasil fitness, probabilitas dan probabilitas kumulatif

F	P	PK
1	0,0003652	0,0003652
64	0,0233747	0,02374
210	0,0766983	0,1004383
238	0,0869248	0,187363
9	0,0032871	0,1906501
76	0,0277575	0,2184076
150	0,0547845	0,2731921
186	0,0679328	0,3411249
203	0,0741417	0,4152666
174	0,06355	0,4788167
241	0,0880205	0,5668371
84	0,0306793	0,5975164
0	0	0,5975164
152	0,055515	0,6530314
172	0,0628196	0,715851
188	0,0686633	0,7845142
223	0,0814463	0,8659606
169	0,0617239	0,9276844
2	0,0007305	0,9284149
196	0,0715851	1

Serangkian proses sekelsi, crossover dan mutasi akan diterapkan setelah perhitungan probabilitas kumulatif. Tiap masing-masing proses akan membentuk populasi terakhir yang terjadi setelah perubahan-perubahan yang terjadi. Tabel 3 merupakan hasil dari populasi setelah mengalami beberapa proses algoritma genetika.

Tabel 3 : Populasi setelah melewati satu generasi

p-ISSN : 2407 – 1846
e-ISSN : 2460 – 8416
Website : jurnal.umj.ac.id/index.php/semnastek

K11	K12	K13	K21	K22	K23	K31	K32	K33
134	47	216	112	213	90	18	7	131
85	47	244	195	85	212	244	196	249
119	63	77	78	145	72	249	18	135
108	47	197	179	47	239	119	234	65
167	89	7	142	179	108	200	116	89
78	64	116	24	242	112	27	191	80
80	44	185	7	1	141	62	242	32
254	29	106	196	200	50	13	29	185
31	232	46	145	120	216	154	178	244
139	85	227	226	212	165	168	137	228
218	244	41	163	168	147	112	242	244
228	85	147	234	69	97	131	141	54
147	47	77	61	232	246	249	18	80
215	82	142	101	103	29	79	165	164
187	120	86	59	55	190	44	6	213
31	45	46	134	116	89	196	242	63
65	86	246	196	224	92	66	112	72
77	115	135	246	224	26	213	112	6
116	196	65	77	45	212	25	137	50
139	26	47	226	45	47	187	242	108

Proses ini akan berlangsung sampai generasi terakhir. Generasi terakhir merupakan kondisi dimana populasi memiliki nilai paling optimal. Pada Tabel 4, dapat dilihat hasil dari pembentukan tiga buah kunci matriks.

Tabel 4 : Hasil algoritme genetika setelah 30 generasi

K11	K12	K13	K21	K22	K23	K31	K32	K33
147	69	62	147	232	82	29	147	147
147	62	147	232	246	29	82	147	246
82	147	232	69	246	82	246	72	147

Tabel 5 merupakan kombinasi tiga buah kunci matriks 3 x 3 pada Hill Cipher. Angka-angka tersebut sudah dipastikan dapat mengembalikan ciphertext menjadi plaintext. Hal ini dapat terjadi karena kunci-kunci tersebut memiliki nilai determinan sesuai dengan yang diharapkan.

Tabel 5 : Kombinasi nilai pada tiap-tiap sel matriks kunci

Key 1			Key 2			Key 3		
147	69	62	147	62	147	82	147	232
147	232	82	232	246	29	69	246	82
29	147	147	82	147	246	246	72	147

B. Implementasi

Berikut ini akan diberikan contoh penggunaaan algoritma Hill Cipher dari kunci yang telah diperoleh dari algoritma genetika sebelumnya. Plaintext yang digunakan adalah “ANDYSAH”. Hill Cipher menggunakan kelipatan 9 untuk melakukan proses enkripsi dan dekripsi. Sehingga panjang teks harus disesuaikan kepada panjang terdekat dari sebelumnya. Kekurangan dari plaintext akan disisip dengan karakter “X”.

Plaintext : ANDYSAHXX

$$\begin{pmatrix} 65 & 78 & 68 \\ 89 & 83 & 65 \\ 72 & 88 & 88 \end{pmatrix}$$

Key : $\begin{pmatrix} 147 & 69 & 62 \\ 147 & 232 & 82 \\ 29 & 147 & 147 \end{pmatrix}$

Ciphertext : ÑË38%`H8

$$\begin{pmatrix} 209 & 203 & 51 \\ 56 & 37 & 17 \\ 96 & 72 & 56 \end{pmatrix}$$

Key Inverse : $\begin{pmatrix} 82 & 131 & 218 \\ 249 & 171 & 228 \\ 57 & 40 & 241 \end{pmatrix}$

Plaintext : ANDYSAHXX

$$\begin{pmatrix} 65 & 78 & 68 \\ 89 & 83 & 65 \\ 72 & 88 & 88 \end{pmatrix}$$

Implementasi pada pembahasan sebelumnya merupakan penerapan dari kunci yang diperoleh dari algoritma genetika. Kunci yang dihasilkan benar-benar akurat sehingga pada saat proses dekripsi terjadi, ciphertext akan berubah kembali kepada plaintext tanpa kesalahan sedikitpun. Hal ini yang menyebabkan algoritma genetika layak membantu kerumitan yang dimiliki oleh algoritma Hill Cipher dalam hal pencarian kombinasi angka pada kunci matriks. Pada saat plaintext dihasilkan ada beberapa karakter

p-ISSN : 2407 – 1846
e-ISSN : 2460 – 8416
Website : jurnal.umj.ac.id/index.php/semnastek

tambahan atau dikenal dengan padding yang berfungsi sebagai pelengkap pada proses enkripsi. Karakter ini dapat diabaikan karena tidak bermakna apa-apa. Contoh karakter ini dapat dilihat pada hasil plaintext yaitu "ANDYSAHXX". Karakter "X" pada kata tersebut tidak mempunyai hubungan apa-apa dengan plaintext sebelumnya.

V. KESIMPULAN

Pada algoritma Hill Cipher yang menggunakan matriks 3x3, mencari kunci yang memiliki determinan yang tepat membutuhkan waktu tertentu. Pencarian dengan cara manual akan memperlambat proses kriptografi. Algoritma genetika sangat membantu proses enkripsi dan dekripsi on the Hill Cipher. Algoritma ini akan menghasilkan serangkaian angka dengan cepat dan tepat. Pembentukan populasi dari algoritma ini akan menghasilkan beberapa alternatif kunci yang dapat digunakan pada algoritma Hill Cipher. Dalam penelitian ini dapat disimpulkan bahwa algoritma genetika memiliki kontribusi yang berharga bila dikombinasikan dengan algoritma Hill Cipher.

REFERENSI

Abdullah, A. A., Khalaf, R., & Riza, M. (2015). A Realizable Quantum Three-Pass Protocol Authentication. *Mathematical Problems in Engineering*.

Ahmed, M., Sanja, B., Aldiaz, D., Rezaei, A., & Omotunde, H. (2008). Diffie-Hellman and Its Application in Security Protocols. *International Journal of Engineering Science and Innovative Technology, 1*(2), 69-73.

Chase, J., & Davis, M. (2010). Extending the Hill Cipher.

Chowdhury, S. I., Shohag, S. A., & Sahid, H. (2011). A Secured Message Transaction Approach by Dynamic Hill Cipher Generation and Digest Concatenation. *International Journal of Computer Applications, 23*(9), 25-31.

Ghanbari, A. A., Broumandnia, A., Navidi, H., & Ahmadi, A. (2012). Brain Computer Interface with Genetic Algorithm. *International Journal of Information and Communication Technology Research, 2*(1), 79-86.

Heidari, E., & Movaghar, A. (2011). An Efficient Method Based On Genetic Algorithms To Solve Sensor Network Optimization Problem. *International journal on applications of graph theory in wireless ad hoc networks and sensor networks, 3*(1), 18-33.

Karegowda, A. G., Manjunath, A., & Jayaram, M. (2011). Application Of Genetic Algorithm Optimized Neural Network Connection Weights For Medical Diagnosis Of Pima Indians Diabetes. *International Journal on Soft Computing, 2*(2), 15-23.

Khalaf, A. A., El-karim, M. S., & Hamed, H. F. (2016). A Triple Hill Cipher Algorithm Proposed to Increase the Security of Encrypted Binary Data and its Implementation Using FPGA. *ICACT Transactions on Advanced Communications Technology, 5*(1), 752-757.

Kumar, R., & C., R. (2015). Analysis of Diffie Hellman Key Exchange Algorithm with Proposed Key Exchange Algorithm. *International Journal of Emerging Trends & Technology in Computer Science, 4*(1), 40-43.

Lin, C. H., Yu, J. L., Liu, J. C., Lai, W. S., & Ho, C. H. (2009). Genetic Algorithm for Shortest Driving Time in Intelligent Transportation Systems. *International Journal of Hybrid Information Technology, 2*(1), 21-30.

Rahman, M. N., Abidin, A. F., Yusof, M. K., & Usop, N. S. (2013). Cryptography: A New Approach of Classical Hill Cipher. *International Journal of Security and Its Applications, 7*(2), 179-190.

Szénási, S., & Vámossy, Z. (2013). Implementation of a Distributed Genetic Algorithm for Parameter Optimization in a Cell Nuclei Detection Project. *Acta Polytechnica Hungarica, 10*(4), 59-86.