 ISSN:2302-8149
e-ISSN:2540-9719

Optimasi Kunci Kriptografi XOR menggunakan Metode Hibrida Algoritma Genetika dan Simulated Annealing

Optimization of XOR Cryptographic Keys using a Hybrid Genetic Algorithm and Simulated Annealing

[1]Naufal Muzakki, [2]Nur Rochmah Dyah Puji Astuti*
[1,2]Program Studi Informatika, Fakultas Teknologi Industri, Universitas Ahmad Dahlan
[1,2]Jl. Ki Ageng Pamenahan, Yogyakarta 55191, Indonesia
*e-mail: rochmahdyah@tif.uad.ac.id

(***received*:** 23 February 2026, ***revised*:** 28 February 2026, ***accepted*:** 2 March 2026)

Abstrak

Tingkat keamanan algoritma enkripsi dasar seperti XOR memiliki ketergantungan yang sangat kuat terhadap derajat keacakan (*randomness*) serta pola distribusi bit dari kunci yang diaplikasikan. Pemanfaatan pendekatan optimasi stokastik seperti *Genetic Algorithm* (GA) dalam proses generasi kunci kerap mengalami hambatan akibat terjadinya konvergensi prematur, yakni kondisi ketika pencarian solusi terhenti pada titik optimum lokal sebelum berhasil mencapai nilai entropi yang maksimal. Penelitian mengajukan strategi algoritma hibrida berurutan yang memadukan GA dengan *Simulated Annealing* (SA) untuk mengatasi masalah stagnasi nilai *fitness* tersebut pada enkripsi dokumen PDF. Strategi dibangun dengan mekanisme dua fase: GA menjalankan fungsi eksplorasi global untuk mengidentifikasi area solusi yang potensial, dilanjutkan oleh SA yang menjalankan eksploitasi lokal dengan mekanisme perturbasi berdasarkan probabilitas Metropolis. Performa algoritma dievaluasi melalui studi komparatif antara GA konvensional dan Hibrida GA-SA. Temuan eksperimen menunjukkan bahwa strategi hibrida berhasil meningkatkan rerata nilai *fitness* sebesar 4,86%, mencapai Entropi Shannon sebesar 7,8952, serta menghasilkan nilai *P-Value* uji NIST sebesar 0,5299. Peningkatan membuktikan bahwa integrasi SA efektif dalam memperbaiki kualitas solusi akhir GA, menghasilkan kunci kriptografi dengan distribusi bit yang lebih homogen, lulus uji keacakan statistik, dan tahan terhadap analisis pola.

Kata kunci: algoritma genetika, optimasi kunci, entropi shannon, *simulated annealing*, kriptografi XOR

Abstract

The security level of basic encryption algorithms, such as XOR, is highly dependent on the randomness and bit distribution pattern of the applied key. The use of stochastic optimization approaches, such as Genetic Algorithm (GA), in key generation often faces challenges due to premature convergence, a condition in which the search halts at a local optimum before achieving maximal entropy. This study proposes a sequential hybrid algorithm strategy that combines GA with Simulated Annealing (SA) to address fitness stagnation in PDF document encryption. The strategy is implemented through a two-phase mechanism: GA performs global exploration to identify potential solution regions, followed by SA performing local exploitation with a perturbation mechanism guided by the Metropolis probability. The algorithm's performance is evaluated through a comparative study between conventional GA and the hybrid GA-SA. Experimental results indicate that the hybrid strategy successfully increases the average fitness value by 4.86%, achieves a Shannon entropy of 7.8952, and attains an NIST test P-value of 0.5299. These improvements demonstrate that the integration of SA effectively enhances the final solution quality of GA, producing cryptographic keys with more uniform bit distribution, passing statistical randomness tests, and exhibiting robustness against pattern analysis.

Keywords: *genetic algorithm, key optimization, shannon entropy, simulated annealing, XOR cryptography*

 ISSN:2302-8149
e-ISSN:2540-9719

1 Pendahuluan

Lonjakan volume transfer data digital pada era transformasi teknologi digital menuntut mekanisme proteksi yang tidak hanya berkecepatan tinggi tetapi juga memiliki fondasi matematis yang solid untuk menghadapi ancaman keamanan siber yang makin rumit. Keamanan informasi digital telah berubah menjadi prioritas strategis yang krusial, mengingat kejadian kebocoran data dapat merusak integritas informasi dan reputasi pengguna secara permanen [1]. Sebagai benteng utama perlindungan informasi, kriptografi menyediakan kerangka matematis untuk menjamin aspek kerahasiaan, integritas, dan ketersediaan data dari akses pihak yang tidak memiliki otorisasi yang sah [2]. Di antara berbagai teknik enkripsi yang ada, algoritma berbasis operasi logika *Exclusive-OR* (XOR) tetap menjadi metode yang paling mendasar dan esensial karena kapasitasnya memproses enkripsi aliran data (*stream cipher*) dengan memadukan bit *plaintext* dan *key* secara efisien [3].

Ditinjau dari sisi performa komputasi, algoritma XOR menawarkan keuntungan yang signifikan berupa kecepatan eksekusi yang tinggi serta konsumsi memori yang minimal, menjadikannya solusi yang ideal untuk lingkungan dengan keterbatasan sumber daya seperti *Internet of Things* (IoT) maupun sistem waktu nyata [4]. Meskipun memiliki efisiensi yang tinggi, tingkat keamanan algoritma memiliki ketergantungan yang bersifat absolut dan linear terhadap kualitas, dimensi, serta tingkat keacakan kunci enkripsi yang digunakan dalam proses pengacakan data [5]. Pemanfaatan kunci yang dibangkitkan secara sembarangan, bersifat statis, atau memiliki periode pengulangan yang pendek akan membuka celah keamanan fatal yang memungkinkan penyerang memecahkan sandi melalui pendekatan statistik sederhana, serangan *brute-force*, maupun *known-plaintext attack* [6].

Untuk mengatasi kerentanan tersebut, pendekatan komputasi cerdas berbasis metaheuristik diperlukan untuk membangkitkan kunci yang memiliki nilai entropi tinggi dan karakteristik *pseudorandom* yang sulit untuk diprediksi. *Genetic Algorithm* (GA) telah terbukti sebagai metode optimasi stokastik yang andal dalam memecahkan masalah pencarian ruang solusi yang besar dan kompleks, termasuk dalam pembentukan kunci kriptografi yang adaptif [7]. Dengan mengadopsi prinsip evolusi biologis Darwin, GA beroperasi secara iteratif melalui mekanisme seleksi alam, rekombinasi genetik (*crossover*), dan mutasi acak untuk mengevaluasi dan mengembangkan populasi solusi potensial menuju kualitas yang lebih optimal [8]. Strategi evolusioner memungkinkan sistem untuk mengeksplorasi ruang pencarian secara global guna menemukan kandidat kunci yang memiliki distribusi bit yang jauh lebih acak dibandingkan metode pembangkitan linear biasa [9].

Meskipun demikian, penerapan GA konvensional seringkali menghadapi kendala teknis berupa fenomena konvergensi prematur, yakni kondisi di mana algoritma kehilangan keberagaman populasi terlalu cepat dan terjebak pada *local optima* sebelum mencapai solusi terbaik secara global [10]. Untuk mengatasi stagnasi nilai *fitness* tersebut, metode hibridisasi algoritma menjadi solusi yang potensial dengan menggabungkan kekuatan eksplorasi GA dengan kapabilitas perbaikan lokal dari algoritma lain [11]. *Simulated Annealing* (SA) dipilih sebagai metode komplementer yang efektif karena memiliki mekanisme probabilistik unik yang mengizinkan penerimaan solusi yang lebih buruk untuk sementara waktu, sehingga memungkinkan algoritma untuk keluar dari jebakan optimum lokal dan melanjutkan pencarian menuju optimum global [12].

Integrasi strategis antara algoritma berbasis populasi dan metode pencarian berbasis trajektori menawarkan keseimbangan dinamis yang penting antara eksplorasi ruang pencarian luas dan eksploitasi solusi terbaik. Studi terdahulu menunjukkan bahwa pendekatan hibrida mampu menutupi kekurangan inheren masing-masing algoritma tunggal, menghasilkan akurasi konvergensi yang lebih akurat dibandingkan metode konvensional [13]. Secara spesifik, penerapan operator SA sebagai tahap lanjutan (*post-processing*) setelah evolusi GA berfungsi efektif untuk melakukan *fine-tuning* terhadap individu terbaik, meningkatkan ketahanan kunci terhadap pola kriptanalisis [14]. Sinergi komputasi memastikan bahwa kunci enkripsi yang dihasilkan tidak hanya acak secara statistik, tetapi juga memiliki kompleksitas tinggi yang memenuhi standar keamanan sistem modern [15].

Penelitian bertujuan untuk merancang dan mengevaluasi strategi hibrida berurutan yang mengintegrasikan GA dan SA untuk mengoptimalkan kunci kriptografi XOR. Pendekatan memanfaatkan kekuatan eksplorasi global GA dan kapabilitas eksploitasi lokal SA secara berurutan untuk menghasilkan kunci dengan kualitas superior. Kontribusi utama penelitian adalah: (1) perancangan arsitektur hibrida sekuensial GA-SA yang spesifik untuk optimasi kunci XOR, (2) analisis komparatif kuantitatif performa keacakan kunci antara metode hibrida dengan GA

 ISSN:2302-8149
e-ISSN:2540-9719

konvensional berdasarkan standar entropi Shannon, (3) pembuktian empiris efektivitas metode hibrida GA-SA sebagai solusi atas masalah konvergensi prematur dalam menghasilkan kunci enkripsi berentropi tinggi yang diterapkan secara spesifik pada pengamanan dokumen berformat PDF.

2 Tinjauan Literatur

Pengembangan keamanan data pada sistem transmisi digital menuntut algoritma kriptografi yang efisien dari segi komputasi namun tetap tangguh terhadap serangan kriptanalisis. Dalam ranah tersebut, algoritma berbasis *Exclusive-OR* (XOR) sering menjadi pilihan utama karena kecepatan eksekusi yang tinggi. Namun, studi literatur menunjukkan bahwa keamanan algoritma XOR memiliki ketergantungan absolut dan linear terhadap kualitas serta tingkat keacakan kunci enkripsi [5]. Kelemahan signifikan sering ditemukan pada penggunaan kunci yang dibangkitkan secara statis, yang membuka celah bagi serangan *brute-force* maupun *known-plaintext attack*. Secara matematis, kerentanan tersebut muncul karena sifat operasi XOR yang reversibel, seperti ditunjukkan pada Persamaan (1):

$$C = P \oplus K \tag{1}$$

di mana P adalah plaintext, C adalah ciphertext, dan K adalah kunci. Keamanan sistem bergantung sepenuhnya pada kualitas kunci K [16]. Jika K memiliki pola berulang, maka P dapat dipulihkan dengan mudah. Oleh karena itu, tren penelitian terkini berfokus pada penggunaan metode metaheuristik untuk membangkitkan kunci dinamis dengan entropi tinggi.

Sejumlah penelitian telah mengeksplorasi penggunaan *Genetic Algorithm* (GA) untuk optimasi kunci. Qasim dan Mahdi [17] mengimplementasikan GA untuk menghasilkan kunci acak pada enkripsi citra, sementara Hussein dan Ayoob [18] membuktikan bahwa kunci hasil optimasi GA pada sistem Vigenere memiliki entropi yang lebih superior dibandingkan kunci manual. Mukherjee dkk. [19] memperkuat pendekatan tersebut dengan memperkenalkan fungsi *fitness* berbasis uji statistik untuk mengukur kualitas kunci secara ketat. Meskipun GA terbukti andal dalam eksplorasi global, penggunaan algoritma secara tunggal (*pure GA*) memiliki kelemahan fundamental berupa kerentanan terhadap konvergensi prematur, di mana solusi terjebak pada optimum lokal sebelum mencapai nilai keacakan maksimal [10].

Untuk mengatasi stagnasi tersebut, integrasi dengan metode pencarian lokal menjadi solusi yang krusial. *Simulated Annealing* (SA) telah terbukti sebagai metode optimasi mandiri yang tangguh. Juniawan dkk. [20] mendemonstrasikan kapabilitas SA dalam mereduksi total jarak rute hingga 36,49% pada masalah kombinatorial skala besar. SA memiliki mekanisme probabilistik unik bernama Kriteria Metropolis yang mengizinkan penerimaan solusi yang lebih buruk untuk sementara waktu untuk keluar dari jebakan optimum lokal, sebagaimana dirumuskan pada Persamaan (2):

$$P(accept) = \begin{cases} 1, \ jika \ \triangle E > 0 \\ e^{\triangle E/T}, \ jika \ \triangle E < 0 \end{cases} \tag{2}$$

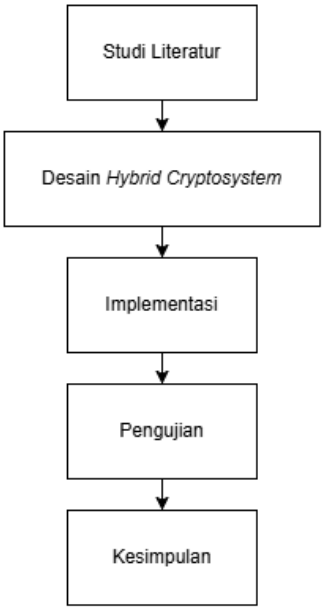
Mekanisme tersebut divalidasi efektif untuk mengatasi stagnasi yang sering dialami algoritma berbasis populasi.

Tren penelitian terkini mengarah pada hibridisasi algoritma. Wu dkk. [14] memvalidasi bahwa strategi hibrida modern merupakan strategi efektif untuk meningkatkan keamanan pada sistem kriptografi kompleks. Namun, penerapan pendekatan hibrida fundamental seperti kombinasi sekuensial GA dan SA pada *cipher* sederhana seperti XOR belum banyak dieksplorasi. Penelitian mengisi celah tersebut (*gap analysis*) dengan menerapkan SA sebagai fase *fine-tuning* lanjutan bagi solusi GA, guna menghasilkan kunci kriptografi yang memenuhi standar acak sempurna.

3 Metode Penelitian

Penelitian menerapkan metode eksperimental algoritmik (*algorithmic experimental research*) untuk menguji performa optimasi kunci. Desain sistem berfokus pada mekanisme komputasi hibrida sekuensial, di mana output optimal dari satu algoritma menjadi input awal bagi algoritma berikutnya untuk mencapai konvergensi yang lebih presisi. Tahapan penelitian dilaksanakan secara sistematis untuk menjamin validitas hasil eksperimen. Alur penelitian secara lengkap ditunjukkan pada Gambar 1.

 ISSN:2302-8149
e-ISSN:2540-9719



Gambar 1 Tahapan penelitian

Tahapan penelitian diawali dengan persiapan dataset sampel berupa dokumen berformat PDF yang berfungsi sebagai objek simulasi enkripsi. Proses selanjutnya berfokus pada perancangan algoritma hibrida yang dijalankan secara berurutan, dimulai dari tahap *Genetic Algorithm* melalui iterasi seleksi, *crossover*, dan mutasi, kemudian dilanjutkan dengan mekanisme transisi untuk mengekstraksi kromosom dengan nilai *fitness* tertinggi. Solusi terbaik tersebut selanjutnya disempurnakan pada tahap *Simulated Annealing* melalui perturbasi dengan penurunan suhu bertahap guna mencapai optimum lokal. Seluruh rancangan algoritma tersebut diimplementasikan dalam bentuk perangkat lunak berbasis bahasa pemrograman Python, yang kemudian divalidasi melalui serangkaian uji coba enkripsi untuk menganalisis perbandingan metrik performa antara metode GA konvensional dan hibrida GA-SA berdasarkan variasi parameter yang telah ditetapkan.

3.1 Desain Hybrid *Cryptosystem*

Mekanisme komputasi dirancang dalam dua fase utama secara sekuensial. Fase pertama bertujuan untuk mengeksplorasi ruang pencarian secara luas, sedangkan fase kedua bertujuan untuk mempertajam solusi yang ditemukan.

3.1.1 Fase 1: Eksplorasi Global (*Genetic Algorithm*)

Genetic Algorithm (GA) digunakan sebagai fase pertama untuk mengeksplorasi ruang pencarian solusi secara global (*global search*). Kunci direpresentasikan sebagai kromosom berupa *string* karakter ASCII atau *byte array*. Komponen utama GA yang diimplementasikan meliputi:

1. Inisialisasi Populasi
   Membangkitkan populasi awal sebanyak N individu secara stokastik (acak). Setiap individu merepresentasikan satu kandidat kunci enkripsi yang unik.
2. Seleksi (*Tournament Selection*)
   Menggunakan metode *Tournament Selection* dengan ukuran turnamen $K = 3$. Metode tersebut memilih individu terbaik dari sekelompok kecil individu yang diambil secara acak, memberikan tekanan seleksi yang seimbang untuk menjaga diversitas [21]. Detail algoritma seleksi disajikan pada Algoritma 1

Algoritma 1
Pseudocode of Tournament Selection Method

```
1. P : Population
2. F : Fitness_Scores
3. K : Tournament_Size = 3
4. Parent1, Parent2 : Selected_Parents
```

 ISSN:2302-8149
e-ISSN:2540-9719

```
5. Initialize Selected_Parents as empty list []
6. For i from 1 to 2 do:
7.    Tournament_Pool ← Randomly select K candidates from P
8.    Winner ← Candidate in Tournament_Pool with highest F value
9.    Add Winner to Selected_Parents
10. End For
11. Return Selected_Parents
```

3. Crossover (*Two-Point Crossover*)
Menerapkan *Two-Point Crossover*. Dua posisi potong dipilih secara acak pada kromosom induk, kemudian segmen di antara kedua titik tersebut dipertukarkan untuk menghasilkan keturunan (*offspring*) dengan variasi genetik baru [22]. Detail langkah komputasi *crossover* disajikan dalam Algoritma 2.

Algoritma 2
Pseudocode of Two-Point Crossover Method

```
1. Parent1, Parent2,
2. CR : Crossover Rate
3. L : Chromosome Length
4. Child1, Child2
5. Parent1, Parent2 : Selected_Parents
6. Initialize Child1 as copy of Parent1
7. Initialize Child2 as copy of Parent2
8. Generate Random_Num between 0 and 100
9. IF Random_Num < Pc THEN:
10.   P1 = Random(0, L-1)
11.   P2 = Random(0, L-1)
12.   IF P1 > P2 THEN
13.      Swap(P1, P2)
14.   END IF
15.   FOR i FROM P1 TO P2 DO:
16.      Temp = Child1[i]
17.      Child1[i] = Child2[i]
18.      Child2[i] = Temp
19.   End For
20. End If
21. Return Child1, Child2
```

4. Mutasi
Strategi mutasi yang diterapkan menggabungkan dua teknik:
   a. *Inversion Mutation*
   Membalik urutan gen pada segmen tertentu [23]. Detail langkahnya dijelaskan pada Algoritma 3.

   Algoritma 3
   Pseudocode of Inversion Mutation Method

```
1. C : Chromosome
2. L : Length
3. C_new : Mutated_Chromosome
4. C_new ← Copy of C
5. P1 ← Random integer between 0 and L-1
```

 ISSN:2302-8149
e-ISSN:2540-9719

```
6.  P2 ← Random integer between 0 and L-1
7.  If P1 > P2 then
8.      swap P1, P2
9.  End If
10. Reverse the segment of C_new from index P1 to P2
Return C_new
```

b. *Random Resetting*
Mengubah nilai gen tertentu menjadi nilai acak baru. Kombinasi kedua teknik bertujuan mempertahankan diversitas genetik dan mencegah konvergensi dini [24]. Detail langkahnya dijelaskan pada Algoritma 4.

Algoritma 4
Pseudocode of Random Resetting Mutation Method

```
1.  C : Chromosome
2.  MR : Mutation_Count
3.  L : Length
4.  C_new : Mutated_Chromosome
5.  C_new ← Copy of C
6.  For i from 1 to MR do:
7.      Idx ← Random integer between 0 and L-1
8.      New_Val ← Random ASCII character (32 to 126)
9.      C_new[Idx] ← New_Val
10. End For
Return C_new
```

3.1.2 Fase 2: Eksploitasi Lokal (*Simulated Annealing*)

Individu terbaik (*best individual*) dari generasi terakhir GA diekstraksi menjadi solusi awal (S_{lama}) untuk fase SA. Algoritma SA bertugas melakukan *fine-tuning* [25] dengan memodifikasi karakter kunci secara stokastik untuk menghasilkan solusi tetangga (S_{baru}) [26]. Keputusan untuk menerima solusi baru didasarkan pada Kriteria Metropolis, seperti yang ditunjukkan pada Persamaan (3):

$$P(accept) = \begin{cases} 1, \ jika \ \triangle E > 0 \\ e^{\triangle E/T}, \ jika \ \triangle E < 0 \end{cases} \quad (3)$$

di mana ΔE adalah selisih *fitness* $\left(Fitness(S_{baru}) - Fitness(S_{lama})\right)$ dan T adalah parameter suhu. Jadwal pendinginan (*cooling schedule*) menggunakan pendekatan geometris, sebagaimana dirumuskan pada Persamaan (4):

$$T_{k+1} = T_k \times \alpha \quad (4)$$

Parameter suhu awal diatur pada T dengan laju pendinginan $\propto$. Proses SA berulang hingga suhu mencapai batas minimum T_{min}.

3.2 Implementasi

Tahap implementasi merupakan proses perwujudan desain rancangan *hybrid cryptosystem* ke dalam bentuk perangkat lunak berbasis bahasa pemrograman Python. Tahapan ini menguraikan spesifikasi lingkungan komputasi yang digunakan selama pengujian, beserta konfigurasi parameter algoritma yang mendasari jalannya simulasi.

3.2.1 Lingkungan Implementasi

Pengujian dilakukan pada perangkat keras dengan spesifikasi prosesor Intel Core i7, RAM 16 GB, dan sistem operasi Windows 11. Spesifikasi tersebut dipilih untuk memastikan tidak ada *bottleneck* perangkat keras yang mempengaruhi pengukuran waktu komputasi selama proses iterasi algoritma yang intensif.

 ISSN:2302-8149
e-ISSN:2540-9719

3.2.2 Parameter Eksperimen

Konfigurasi parameter yang digunakan dalam eksperimen untuk memastikan reproduktibilitas hasil ditunjukkan pada Tabel 1. Penentuan nilai parameter didasarkan pada studi literatur terdahulu dan hasil uji coba pendahuluan untuk mendapatkan keseimbangan antara kecepatan konvergensi dan diversitas solusi.

Tabel 1 Konfigurasi parameter algoritma

Parameter	menggunakan (*t*)
Jumlah Generasi GA [27]	50
Crossover GA	4
Mutasi GA	2
Suhu Awal SA [28]	100
Cooling Rate SA	0,95

3.3 Pengujian

Tahap pengujian dilakukan untuk mengevaluasi kinerja dan kualitas kunci kriptografi yang dihasilkan oleh *hybrid cryptosystem*. Evaluasi ini bertujuan untuk memvalidasi secara kuantitatif apakah integrasi *Simulated Annealing* pada *Genetic Algorithm* mampu memberikan peningkatan performa yang signifikan dibandingkan dengan metode konvensional.

3.3.1 Skenario Pengujian

Untuk memverifikasi efektivitas metode yang diusulkan, skenario pengujian dirancang dalam tiga aspek: (1) komparasi metode, yang membandingkan secara langsung kinerja GA konvensional dengan hibrida GA-SA berdasarkan metrik *fitness* dan entropi, (2) konsistensi panjang kunci, di mana pengujian difokuskan pada panjang kunci tetap 128-bit untuk melihat stabilitas konvergensi pada ruang pencarian yang sama, dan (3) validasi data, menggunakan 15 file dokumen PDF yang berbeda untuk memastikan algoritma bekerja konsisten pada berbagai input.

3.3.2 Metrik Evaluasi

Untuk mengukur efektivitas metode hibrida yang diusulkan, penelitian menggunakan empat metrik evaluasi kuantitatif yang dihitung secara otomatis oleh sistem:

1. Nilai *Fitness*
   Merupakan nilai fungsi tujuan yang digunakan algoritma untuk memandu pencarian solusi [29]. Nilai *fitness* merepresentasikan tingkat kemiripan kunci kandidat terhadap target optimasi, di mana semakin tinggi nilai F, semakin baik kualitas kuncinya. Persamaan matematisnya didefinisikan pada Persamaan (5):

   $$Fitness = \frac{100.000}{\sum|Target - Kromosom|+1} \quad (5)$$

   pada Persamaan (5), nilai konstanta 100.000 digunakan sebagai faktor skala (*scaling factor*) untuk membesarkan nilai hasil pembagian agar lebih mudah dibaca dan dibandingkan. Penyebut $\sum|Target - Kromosom|$ menghitung total selisih nilai ASCII antara kunci yang dibangkitkan dengan target yang diinginkan. Penambahan angka $+1$ pada penyebut berfungsi untuk menghindari pembagian dengan nol (*division by zero*) jika solusi sempurna ditemukan (selisih = 0). Dengan demikian, semakin kecil selisih (*error*), semakin tinggi nilai *fitness* yang dihasilkan.
2. Nilai Entropi Shannon
   Digunakan untuk mengukur tingkat ketidakpastian informasi atau keacakan distribusi *byte* dalam *ciphertext*. Entropi Shannon dihitung menggunakan rumus standar entropi [30], seperti yang ditunjukkan pada Persamaan (6):

   $$Entropi = -\sum_{i=1}^{n} p(x_i) log_2 p(x_i) \quad (6)$$

   di mana n adalah jumlah total simbol unik (256 kemungkinan untuk satu byte), dan $p(x_i)$ adalah probabilitas kemunculan *byte* ke-i dalam *stream* data. Kunci kriptografi yang baik harus memiliki distribusi bit yang seragam, sehingga nilai entropinya mendekati batas maksimum teoritis untuk sistem 8-bit, yaitu 8.0 *bit/byte*. Semakin dekat nilai H ke 8.0, semakin sulit pola kunci diprediksi.

 ISSN:2302-8149
e-ISSN:2540-9719

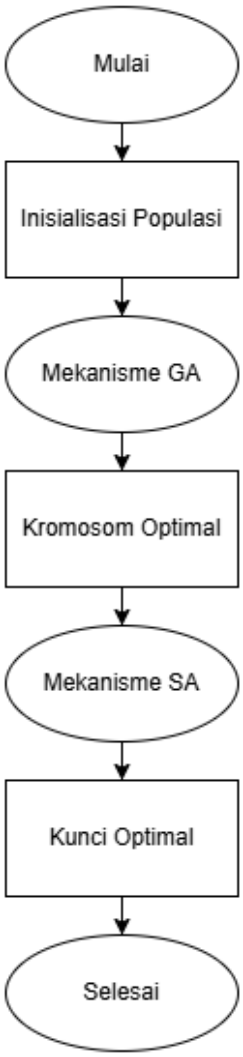
3. Waktu Komputasi
   Waktu komputasi mengukur durasi eksekusi riil (dalam detik) yang dibutuhkan sistem mulai dari inisialisasi populasi awal hingga ditemukannya kunci final. Pengukuran waktu mencakup total waktu proses evolusi GA dan fase *fine-tuning* pada *Simulated Annealing.*
4. *P-Value*
   Validasi statistik menggunakan *Monobit Frequency Test* untuk menentukan apakah distribusi bit 0 dan 1 pada kunci bersifat acak [31]. Kunci dinyatakan acak (lulus uji) jika $P - value \geq 0.01$, dan dinyatakan tidak acak (gagal uji) jika $P - value \leq 0.01$.

4 Hasil dan Pembahasan

Kunci kriptografi yang dihasilkan melalui metode hibrida GA-SA difungsikan sebagai komponen utama dalam skema pengamanan dokumen digital. Dalam mekanisme enkripsi XOR, keamanan data sangat bergantung pada kualitas kunci *keystream*. Kunci hibrida tersebut dirancang untuk menghilangkan pola berulang yang sering muncul pada generator kunci konvensional. Dengan nilai entropi dan *fitness* yang telah dimaksimalkan, kunci hasil optimasi berfungsi untuk meratakan distribusi frekuensi karakter pada *ciphertext*, sehingga dokumen yang dienkripsi menjadi resistan terhadap serangan kriptanalisis berbasis statistik dan *frequency analysis*.

4.1 Proses Hybrid Cryptosystem

Proses optimasi kunci dilakukan menggunakan pendekatan hibrida berurutan yang menggabungkan keunggulan eksplorasi global dari *Genetic Algorithm* (GA) dan eksploitasi lokal dari *Simulated Annealing* (SA). Alur kerja sistem secara keseluruhan divisualisasikan pada Gambar 2.



Gambar 2 Tahapan penelitian

 ISSN:2302-8149
e-ISSN:2540-9719

Berdasarkan Gambar 2, tahapan proses sistem dimulai dengan membangkitkan populasi kunci awal secara acak untuk menciptakan keragaman genetik. Populasi tersebut kemudian diproses melalui dua mekanisme utama untuk mencapai konvergensi yang optimal.

4.1.1 Mekanisme GA

Populasi awal diproses menggunakan operator genetika yang meliputi seleksi turnamen, *two-point crossover*, dan mutasi hibrida. Tahap tersebut bertujuan untuk melakukan eksplorasi ruang solusi yang luas (*global search*) guna menemukan kandidat kunci yang potensial. *Output* dari fase GA adalah "Kromosom Optimal" (individu terbaik dalam populasi akhir). Namun, hasil tersebut sering kali baru mencapai tahap *sub-optimal* karena GA rentan mengalami stagnasi atau konvergensi prematur pada generasi akhir akibat hilangnya diversitas populasi.

4.1.2 Mekanisme SA

Untuk mengatasi kelemahan tersebut, kromosom optimal dari GA tidak langsung digunakan sebagai kunci akhir, melainkan dijadikan *input* awal bagi mekanisme *Simulated Annealing*. Fase SA bertugas memperkuat kualitas kunci dengan melakukan "guncangan" (*perturbation*) terkontrol dan pencarian lokal (*local search*). Mekanisme probabilitas penerimaan pada SA memungkinkan algoritma menerima solusi yang lebih buruk sementara waktu untuk keluar dari jebakan optimum lokal yang dihasilkan GA. Setelah melalui penyempurnaan di fase SA, diperoleh "Kunci Optimal" final yang memiliki nilai entropi dan *fitness* tertinggi, yang siap digunakan untuk enkripsi.

4.2 Perbandingan GA Murni dengan Hibrida GA-SA

Untuk memvalidasi keunggulan metode yang diusulkan, dilakukan komparasi langsung antara GA konvensional dan hibrida GA-SA menggunakan kunci sepanjang *128-bit* dengan 15 data pengujian. Tabel 2 merangkum data statistik performa kedua metode.

Tabel 2 Perbandingan GA murni vs hibrida GA-SA

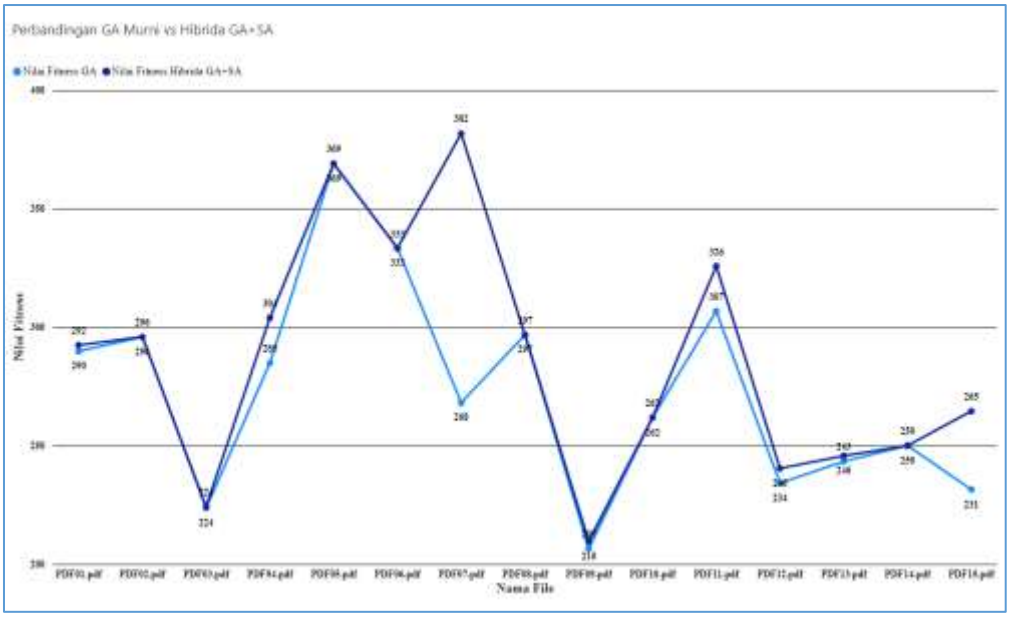
Metrik	GA Murni	Hibrida GA-SA	Peningkatan
Entropi rata-rata	7,8925	7,8952	+ 0,03%
Fitness rata-rata	273,04	286,33	+ 4,86%
Waktu komputasi rata-rata	0,0033	0,0035	-
P-Value rata-rata	0,5085	0,5299	+ 4,20%

Berdasarkan Tabel 2, strategi hibrida GA-SA terbukti secara konsisten unggul dibandingkan GA konvensional. Peningkatan rerata nilai Entropi Shannon menjadi 7,8952 mengindikasikan bahwa distribusi bit dalam kunci enkripsi mendekati kondisi acak sempurna mendekati nilai ideal 8,0. Meskipun selisih kenaikan angka entropi terlihat marjinal secara numerik sebesar (+0,03%), dalam konteks kriptografi, peningkatan desimal tersebut sangat signifikan karena merepresentasikan penurunan drastis pada probabilitas munculnya pola berulang yang dapat dieksploitasi oleh penyerang.

Hal tersebut berkorelasi lurus dengan peningkatan rerata nilai *fitness* sebesar 4,86% dari 273,04 menjadi 286,33, yang menegaskan bahwa mekanisme hibrida mampu membimbing pencarian solusi menuju titik optimal yang lebih presisi. Selain itu, validitas keacakan diperkuat oleh kenaikan rerata *P-Value* menjadi 0,5299. Angka tersebut jauh di atas ambang batas signifikansi 0,01, memberikan bukti statistik kuat bahwa proporsi bit '0' dan '1' sangat seimbang dan tidak dapat dibedakan dari derau acak (*random noise*), sehingga aman dari serangan berbasis frekuensi.

Ditinjau dari aspek efisiensi, penerapan strategi hibrida memang memberikan sedikit penambahan beban komputasi, di mana waktu komputasi meningkat menjadi 0,0035 detik. Namun, peningkatan durasi tersebut sangat tidak signifikan dan dapat diabaikan. Dalam sistem keamanan informasi, *trade-off* antara sedikit penambahan waktu proses demi mendapatkan entropi yang lebih tinggi adalah pertukaran yang sangat menguntungkan. Analisis perilaku algoritma tersebut dikonfirmasi melalui kurva konvergensi pada Gambar 3.

 ISSN:2302-8149
e-ISSN:2540-9719



Gambar 3 Kurva *fitness* GA murni vs hibrida GA-SA

4.3 Validasi Fungsional Enkripsi dan Dekripsi

Selain analisis performa berdasarkan metrik entropi, validasi fungsional dilakukan untuk memastikan bahwa kunci optimal yang dihasilkan mampu melakukan proses pengamanan data secara visual maupun struktural. Pengujian dilakukan dengan mengenkripsi *file* dokumen sampel dan kemudian mendekripsinya kembali. Tahap pertama adalah observasi dokumen asli sebelum enkripsi, seperti yang ditunjukkan pada Gambar 4.

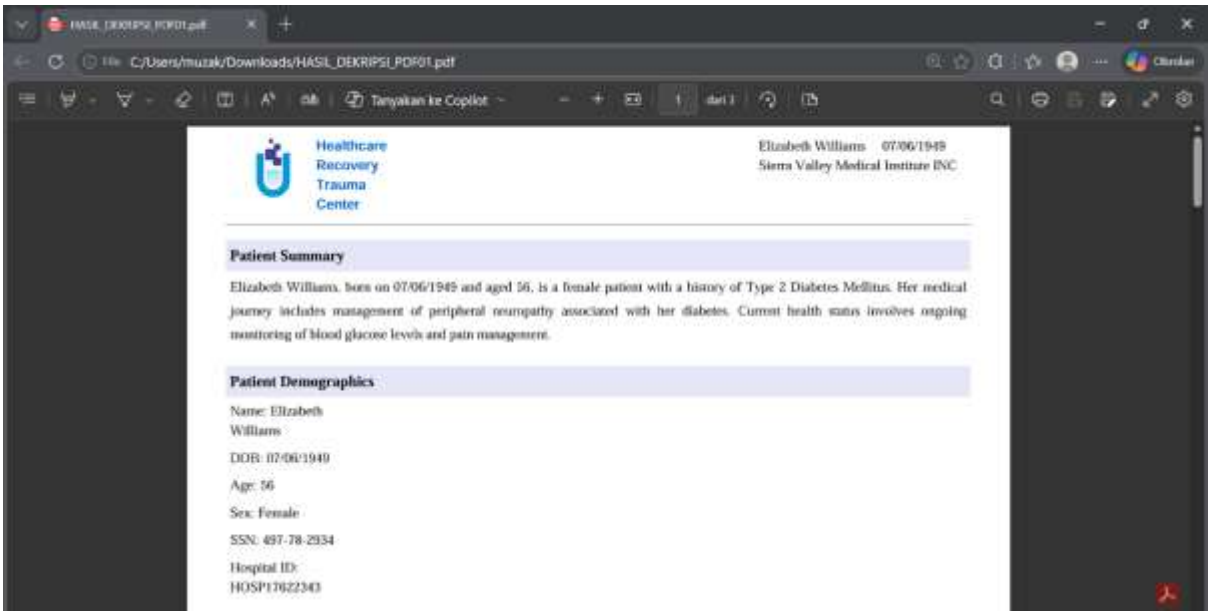
Gambar 4 Dokumen asli yang terbaca jelas

Gambar 4 merepresentasikan kondisi awal dokumen (*plaintext*) berisi informasi yang dapat dibaca dan dipahami. Struktur *file*, termasuk *header* dan konten teks, berada dalam kondisi utuh dan valid. Dokumen tersebut menjadi objek *input* (*P*) untuk proses enkripsi menggunakan kunci hibrida. Selanjutnya, dokumen diproses menggunakan algoritma XOR dengan kunci optimal hasil hibrida GA-SA. Hasilnya dapat dilihat pada Gambar 5.

 ISSN:2302-8149
e-ISSN:2540-9719

Gambar 5 Dokumen terenkripsi (*ciphertext*) yang berisi karakter acak

Pada Gambar 5, terlihat bahwa setelah operasi XOR, struktur *file* asli telah berubah total menjadi susunan *byte* atau karakter acak yang tidak memiliki makna linguistik. Visualisasi ciphertext melalui *code editor* tersebut memperlihatkan distribusi karakter biner yang sepenuhnya teracak. Tidak ada bagian dari informasi asli yang dapat diidentifikasi secara visual. Perubahan drastis tersebut membuktikan bahwa aspek kerahasiaan (*confidentiality*) telah terpenuhi, di mana informasi sensitif berhasil disembunyikan dari akses yang tidak sah. Tahap terakhir adalah proses dekripsi untuk mengembalikan dokumen ke bentuk aslinya, seperti yang ditunjukkan pada Gambar 6.



Gambar 6 Dokumen hasil dekripsi yang kembali ke bentuk semula

Gambar 6 menunjukkan hasil pemulihan dokumen setelah dilakukan operasi XOR kembali menggunakan kunci yang sama. Terlihat bahwa dokumen dapat dibuka dan dibaca dengan format yang identik dengan Gambar 4 tanpa adanya cacat data (*corrupted file*). Hal tersebut memvalidasi bahwa algoritma optimasi kunci yang diterapkan tidak merusak sifat reversibilitas (*reversibility*) operasi XOR dan menjamin integritas (*integrity*) data tetap terjaga selama proses kriptografi.

 ISSN:2302-8149
e-ISSN:2540-9719

5 Kesimpulan

Penelitian berhasil mengembangkan strategi optimasi kunci kriptografi XOR menggunakan pendekatan hibrida berurutan yang mengintegrasikan *Genetic Algorithm* dan *Simulated Annealing*. Berdasarkan temuan eksperimen komparatif, strategi hibrida terbukti unggul secara signifikan dibandingkan *Genetic Algorithm* konvensional dalam menghasilkan kualitas kunci yang lebih baik, dikuantifikasi dengan peningkatan rerata nilai *fitness* sebesar 4,86% (dari 273,04 menjadi 286,33) dan pencapaian nilai Entropi Shannon rerata sebesar 7,8952, yang semakin mendekati nilai ideal acak sempurna 8,0. Peningkatan kualitas tersebut juga divalidasi secara statistik melalui kenaikan nilai *P-Value* pada uji NIST menjadi 0,5299, yang mengindikasikan bahwa distribusi bit kunci hibrida lebih seragam dan sulit diprediksi. Secara algoritmik, integrasi *Simulated Annealing* sebagai fase lanjutan (*post-processing*) terbukti efektif mengatasi kelemahan stagnasi atau konvergensi prematur yang sering dialami oleh *Genetic Algorithm*, di mana mekanisme perturbasi stokastik dan kriteria penerimaan Metropolis pada *Simulated Annealing* mampu mengeluarkan solusi dari jebakan optimum lokal dan menuntun pencarian menuju optimum global dengan presisi yang lebih tinggi. Selain itu, uji validasi fungsional menegaskan bahwa kompleksitas proses optimasi tidak merusak struktur data, di mana dokumen PDF yang dienkripsi berhasil diamankan secara visual (*confidentiality*) dan dapat didekripsi kembali ke format aslinya tanpa cacat data (*integrity*), membuktikan bahwa strategi hibrida aman untuk diimplementasikan. Sebagai rekomendasi untuk pengembangan riset selanjutnya, disarankan untuk memperluas ruang lingkup pengujian dengan menerapkan strategi hibrida pada variasi format multimedia yang lebih kompleks, seperti citra digital atau audio, serta studi komparasi dengan metode metaheuristik modern lainnya, seperti *Particle Swarm Optimization* atau *Ant Colony Optimization*, perlu dilakukan untuk mengeksplorasi efisiensi waktu komputasi yang lebih optimal dalam pembangkitan kunci dinamis. Hal ini penting mengingat strategi hibrida saat ini memiliki *trade-off* berupa sedikit peningkatan waktu eksekusi demi mencapai keamanan yang lebih tinggi.

Referensi

[1] K. Khairani and M. Z. Siambaton, "Pengamanan Data Teks menggunakan Algoritma Kriptografi Elgamal dan XOR dari Serangan *Hacker*," *Sudo J. Tek. Inform.*, Vol. 2, No. 4, pp. 176–187, Dec. 2023, DOI: 10.56211/sudo.v2i4.401.

[2] F. S. Febriyani and A. Arfriandi, "Implementasi Algoritma RC4 pada Sistem Pengamanan Dokumen Digital Soal Ujian," *JISKA (Jurnal Inform. Sunan Kalijaga)*, Vol. 6, No. 3, pp. 171–177, Sep. 2021, DOI: 10.14421/jiska.2021.6.3.171-177.

[3] B. D. Kurniawan, M. A. Rosid, I. A. Kautsar, and N. E. Pratama, "Rancang Bangun *Library Web Token* untuk Enkripsi HTTP Data menggunakan Eksklusif-OR (XOR)," *Phys. SCI. Life SCI. Eng.*, Vol. 1, No. 1, p. 14, Jan. 2024, DOI: 10.47134/pslse.v1i1.164.

[4] D. Jonas, I. Sembiring, A. Setiawan, D. Manongga, I. R. Widiasari, and D. Julianingsih, "*XOR Encryption based on Index Value Enumeration used in IoT based Healthcare*," in *2023 11th International Conference on Cyber and IT Service Management (CITSM)*, IEEE, Nov. 2023, pp. 1–5. DOI: 10.1109/CITSM60085.2023.10455527.

[5] P. Simamora, V. Verawijaya, and S. A. Pasaribu, "Kunci Simetris pada Perangkat IoT: Kajian Literatur," *J. Sist. Inf. dan Teknol. Jar.*, Sep. 2024, DOI: 10.63703/sisfotekjar.v5i2.62.

[6] M. D. Asrofa, S. Bahri, and K. Kasliono, "*One-Time Pad Cryptography for Secure Data Transmission in IoT Smart Door using QR Code*," *J. Media Inf. Teknol.*, Vol. 2, No. 2, pp. 133–148, Oct. 2025, DOI: 10.69616/mit.v2i2.248.

[7] M. T. Gaata, Z. O. Ahmed, and R. S. Ali, "*Secure Big Data Transmission based on Modified Reverse Encryption and Genetic Algorithm*," *Iraqi J. SCI.*, pp. 439–451, Jan. 2023, DOI: 10.24996/ijs.2023.64.1.39.

[8] R. B. Abduljabbar, O. K. Hamid, and N. J. Alhyani, "*Features of Genetic Algorithm for Plain Text Encryption*," *Int. J. Electr. Comput. Eng.*, Vol. 11, No. 1, p. 434, Feb. 2021, DOI: 10.11591/ijece.v11i1.pp434-441.

[9] P. Bagane and D. K. S., "*Enriching AES Through the Key Generation from Genetic Algorithm*," *Indian J. Comput. SCI. Eng.*, Vol. 12, No. 4, pp. 955–963, Aug. 2021, DOI: 10.21817/indjcse/2021/v12i4/211204141.

[10] S. S. Sultana, T. Tanabe, T. Fausten, and M. Irie, "*Avoiding Premature Convergence to Local*

 ISSN:2302-8149
e-ISSN:2540-9719

Optima with Adaptive Exploration for Genetic Algorithms," in *GECCO 2025 Companion - Proceedings of the 2025 Genetic and Evolutionary Computation Conference Companion*, New York, NY, USA: ACM, Jul. 2025, pp. 539–542. DOI: 10.1145/3712255.3726640.

[11] B. A. Kumar, B. Jyothi, A. R. Singh, M. Bajaj, R. S. Rathore, and M. B. Tuka, "*Hybrid Genetic Algorithm-Simulated Annealing based Electric Vehicle Charging Station Placement for Optimizing Distribution Network Resilience*," *SCI. Rep.*, Vol. 14, No. 1, p. 7637, Apr. 2024, DOI: 10.1038/s41598-024-58024-8.

[12] M. A. Rahman, M. J. N. Mim, S. Afrin, A. Islam, and R. Ahmed, "*Simulated Annealing Optimization Algorithm with Self-Escape Mechanism for Travelling Salesman Problem*," *Saudi J. Eng. Technol.*, Vol. 10, No. 05, pp. 236–242, May 2025, DOI: 10.36348/sjet.2025.v10i05.003.

[13] P. Hidayatullah, I. Irwansyah, Q. A, and B. N. Syechah, "*Pipeline Network Optimization using Hybrid Algorithm between Simulated Annealing and Genetic Algorithms*," *Eig. Math. J.*, pp. 30–39, Jan. 2022, DOI: 10.29303/emj.v4i2.100.

[14] W. Yu, Y. Huang, Y. Yang, and A. Garcia-Ortiz, "*Hybrid Genetic Algorithm Combng Simulated Annealing for Task Allocation with Data Security*," in *2023 19th International Conference on Distributed Computing in Smart Systems and the Internet of Things (DCOSS-IoT)*, IEEE, Jun. 2023, pp. 134–141. doi: 10.1109/DCOSS-IoT58021.2023.00033.

[15] G. Singla and G. Kaur, "*New Enhanced Hybrid Cryptographic Algorithms in Cloud Network*," in *2023 1st International Conference on Advances in Electrical, Electronics and Computational Intelligence, ICAEECI 2023*, IEEE, Oct. 2023, pp. 1–6. DOI: 10.1109/ICAEECI58247.2023.10370940.

[16] R. Reymond, Johanes Manullang, Jhoische Tamba, Farel Parasian Sitohang, and Eikel Nioisha Ginting, "*Cryptography With One-Time Pad* (*OTP*) *Algorithm Xor Based*," *J. Tek. Indones.*, Vol. 3, No. 02, pp. 54–60, Nov. 2024, DOI: 10.58471/ju-ti.v3i02.664.

[17] R. A. Q. and B. S. Mahdi, "*Image Protection using Genetic Algorithm and Cipher Technique*," *Iraqi J. Comput. Commun. Control Syst. Eng.*, pp. 160–166, Dec. 2022, DOI: 10.33103/uot.ijccce.22.4.13.

[18] A. A. Hussein and N. K. Ayoob, "*Key Generation for Vigenere Ciphering based on Genetic Algorithm*," *J. Univ. BABYLON Pure Appl. SCI.*, pp. 200–208, Mar. 2022, DOI: 10.29196/jubpas.v30i1.4134.

[19] P. Mukherjee, H. Garg, C. Pradhan, S. Ghosh, S. Chowdhury, and G. Srivastava, "*Best Fit DNA-based Cryptographic Keys: The Genetic Algorithm Approach*," *Sensors*, Vol. 22, No. 19, p. 7332, Sep. 2022, DOI: 10.3390/s22197332.

[20] S. Juniawan, U. Roysen, F. Alam, Z. Zainuddin, D. Daruki, and H. Taher, "*Optimization of Flight Routes Employing the Simulated Annealing Method in the Context of the Indonesian National Airline Industry*," *IJIEM - Indones. J. Ind. Eng. Manag.*, Vol. 5, No. 3, p. 691, Mar. 2025, DOI: 10.22441/ijiem.v5i3.21748.

[21] Y. Pyrih, M. Klymash, M. Kaidan, and B. Strykhalvuk, "*Investigating the Efficiency of Tournament Selection Operator in Genetic Algorithm for Solving TSP*," in *5th IEEE International Conference on Advanced Information and Communication Technologies, AICT 2023 - Proceedings*, IEEE, Nov. 2023, pp. 170–173. DOI: 10.1109/AICT61584.2023.10452423.

[22] I. W. Supriana, M. A. Raharja, I. M. S. Bimantara, and D. Bramantya, "Implementasi Dua Model *Crossover* pada Algoritma Genetika untuk Optimasi Penggunaan Ruang Perkuliahan," *J. Resist. (Rekayasa Sist. Komputer)*, Vol. 4, No. 2, pp. 167–177, Oct. 2021, DOI: 10.31598/jurnalresistor.v4i2.758.

[23] P. Larrañaga, C. M. H. Kuijpers, R. H. Murga, I. Inza, and S. Dizdarevic, "*Genetic Algorithms for the Travelling Salesman Problem: A Review of Representations and Operators*," *Artif. Intell. Rev.*, Vol. 13, No. 2, pp. 129–170, Apr. 1999, DOI: 10.1023/A:1006529012972.

[24] M. Mitchell, *An Introduction to Genetic Algorithms*. The MIT Press, 1996. DOI: 10.7551/mitpress/3927.001.0001.

[25] A. Hemmak, "*Optimal Adjusting of Simulated Annealing Parameters*," *Vojnoteh. Glas.*, Vol. 72, No. 1, pp. 80–93, 2024, DOI: 10.5937/vojtehg72-47242.

[26] D. Henderson, S. H. Jacobson, and A. W. Johnson, "*The Theory and Practice of Simulated*

Annealing," in *Handbook of Metaheuristics*, Boston: Kluwer Academic Publishers, 2006, pp. 287–319. DOI: 10.1007/0-306-48056-5_10.

[27] J. Jasmani and A. Mahmudi, "Optimalisasi Jalur Terpendek menggunakan Algoritma Genetika," *J-INTECH*, Vol. 11, No. 1, pp. 128–139, Jul. 2023, DOI: 10.32664/j-intech.v11i1.809.

[28] P. B. Utomo and W. Setiafindari, "Optimasi Penjadwalan Produksi menggunakan Metode *Simulated Annealing* di Industri XYZ," *Borobudur Eng. Rev.*, Vol. 1, No. 1, pp. 49–55, Mar. 2021, DOI: 10.31603/benr.4610.

[29] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Vol. 27, No. 02. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1989. DOI: 10.5860/choice.27-0936.

[30] C. E. Shannon, "*A Mathematical Theory of Communication*," *Bell Syst. Tech. J.*, Vol. 27, No. 3, pp. 379–423, Jul. 1948, DOI: 10.1002/j.1538-7305.1948.tb01338.x.

[31] A. Rukhin, J. Soto, and J. Nechvatal, "*A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications*," Gaithersburg, MD, 2010. DOI: 10.6028/NIST.SP.800-22r1a.