

## Modifikasi Algoritma Genetika untuk Penyelesaian Permasalahan Penjadwalan Pelajaran Sekolah

**Rahman Erama\*<sup>1</sup>, Retantyo Wardoyo<sup>2</sup>**

<sup>1</sup>Jurusan Teknik Informatika dan Komputer, Politeknik Gorontalo

<sup>2</sup>Jurusan Ilmu Komputer dan Elektronika, FMIPA UGM, Yogyakarta

e-mail: \*[rahman.erima@gmail.com](mailto:rahman.erima@gmail.com), [rw@ugm.ac.id](mailto:rw@ugm.ac.id)

### **Abstrak**

*Modifikasi Algoritma Genetika pada penelitian ini dilakukan berdasarkan temuan-temuan para peneliti sebelumnya tentang kelemahan Algoritma Genetika. Temuan-temuan yang dimaksud terkait proses crossover sebagai salah satu tahapan terpenting dalam Algoritma Genetika dinilai tidak menjamin solusi yang lebih baik oleh beberapa peneliti. Berdasarkan temuan-temuan oleh beberapa peneliti sebelumnya, maka penelitian ini akan mencoba memodifikasi Algoritma Genetika dengan mengeliminasi proses crossover yang menjadi inti permasalahan dari beberapa peneliti tersebut. Eliminasi proses crossover ini diharapkan melahirkan algoritma yang lebih efektif sebagai alternative untuk penyelesaian permasalahan khususnya penjadwalan pelajaran sekolah.*

*Tujuan dari penelitian ini adalah Memodifikasi Algoritma Genetika menjadi algoritma alternatif untuk menyelesaikan permasalahan penjadwalan sekolah, sehingga diharapkan terciptanya algoritma alternatif ini bisa menjadi tambahan referensi bagi para peneliti untuk menyelesaikan permasalahan penjadwalan lainnya.*

*Algoritma hasil modifikasi yang mengeliminasi tahapan crossover pada algoritma genetika ini mampu memberikan performa 3,06% lebih baik dibandingkan algoritma genetika sederhana dalam menyelesaikan permasalahan penjadwalan sekolah.*

**Kata kunci**—algoritma genetika, penjadwalan sekolah, eliminasi crossover

### **Abstract**

*Modified Genetic Algorithm in this study was based on the findings of previous researchers about the weakness of Genetic Algorithms. crossover as one of the most important stages in the Genetic Algorithms considered not guarantee a better solution by several researchers. Based on the findings by previous researchers, this research will try to modify the genetic algorithm by eliminating crossover<sup>2</sup> which is the core problem of several researchers. Elimination crossover is expected to create a more effective algorithm as an alternative to the settlement issue in particular scheduling school.*

*This study is intended to modify the genetic algorithm into an algorithm that is more effective as an alternative to solve the problems of school scheduling. So expect the creation of this alternative algorithm could be an additional resource for researchers to solve other scheduling problems.*

*Modified algorithm that eliminates the crossover phase of the genetic algorithm is able to provide 2,30% better performance than standard genetic algorithm in solving scheduling problems school.*

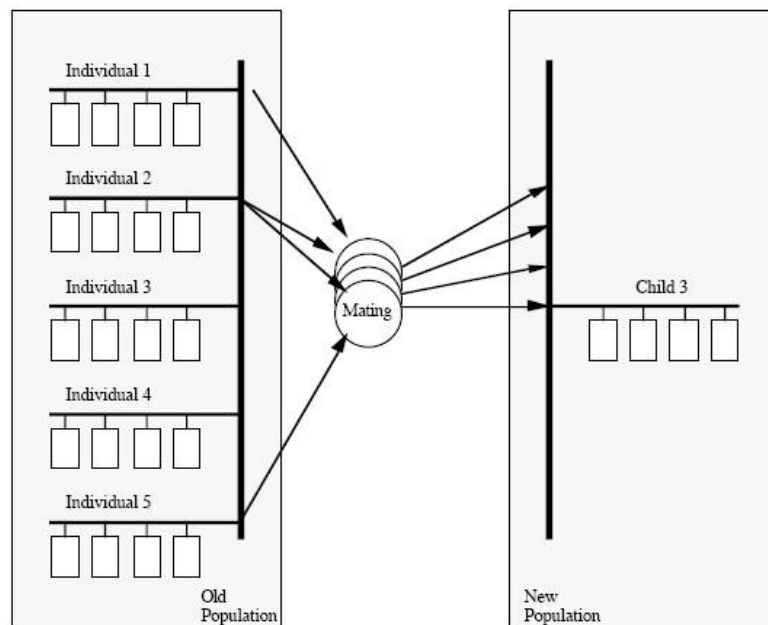
**Keywords**—Genetic Algorithm, timetabling school, eliminate crossover

## 1. PENDAHULUAN

Algoritma Genetika merupakan metode yang menggunakan evolusi alam sebagai gagasan utamanya dalam menyelesaikan suatu permasalahan tertentu. Algoritma ini diterapkan melalui simulasi komputer yang dimulai dari populasi individu yang dibangkitkan secara acak kemudian dievaluasi hingga mencapai solusi terbaik. Pemanfaatannya dalam menyelesaikan permasalahan-permasalahan yang beragam, menyebabkan algoritma ini bertransformasi menjadi Algoritma Genetika dengan wujud berbeda-beda yang dihasilkan melalui modifikasi guna meningkatkan kinerja Algoritma dan disesuaikan dengan permasalahan yang ditemui.

Modifikasi Algoritma genetika telah banyak dilakukan oleh para peneliti dengan tujuan yang berbeda dan disesuaikan dengan permasalahan yang ditemui. Abramson dan Abela [1], melakukan modifikasi terhadap Algoritma Genetika dengan melihat kelemahan pada proses *crossover* yang pada umumnya disimulasikan secara berurutan. Menurutnya, dalam banyak simulasi penyelesaian permasalahan menggunakan Algoritma Genetika, proses yang dilakukan secara berurutan menyebabkan kinerja Algoritma Genetika menjadi lambat. Sehingga dilakukan pemrosesan paralel dalam proses *crossover* untuk memperoleh waktu komputasi yang lebih cepat.

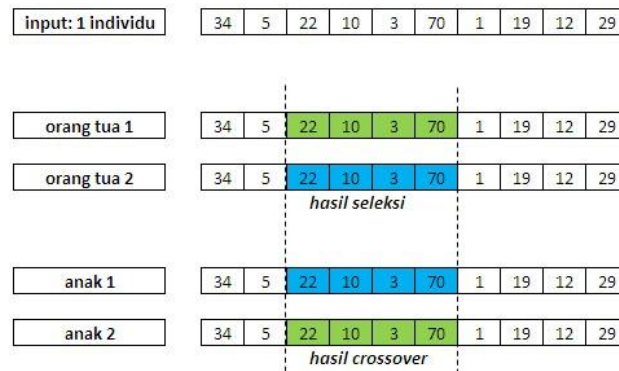
Gambar 1 menunjukkan ilustrasi pemrosesan paralel pada proses *crossover* dimana sejumlah pasangan kromosom orang tua yang akan diproses pada tahap *crossover* dipilih secara acak. Sejumlah pasangan orang tua terpilih di *crossover* secara bersamaan sehingga dihasilkan sejumlah anak untuk menempati populasi baru pada suatu generasi. Berbeda dengan proses *crossover* pada algoritma genetika sederhana yang melakukan proses *crossover* secara berulang untuk menghasilkan sejumlah anak, pada penelitiannya, proses *crossover* hanya dilakukan sekali untuk setiap generasi.



Gambar 1. Pemrosesan paralel pada proses *crossover*

Bambrick [2], melakukan hal yang berbeda dalam melakukan peningkatan kinerja algoritma. Dalam penelitiannya, dilakukan modifikasi dengan menambahkan proses tertentu setelah proses *crossover* dan mutasi untuk menyelesaikan permasalahan penjadwalan universitas. Modifikasi algoritma yang sejenis juga pernah dilakukan oleh [3][4][5], namun dengan usulan algoritma yang berbeda. Modifikasi lainnya dilakukan dengan cara mengkombinasikannya dengan algoritma lainnya seperti yang dilakukan oleh [6][7][8].

Modifikasi pada Algoritma Genetika seperti disebutkan sebelumnya, beberapa diantaranya disebabkan oleh temuan yang menitik-beratkan pada proses *crossover* sebagai sumber kelemahannya. Suyanto [9], berpendapat serupa bahwa proses memindah silangkan dua buah kromosom yang terjadi pada tahapan *crossover* yang merupakan salah satu komponen terpenting dalam Algoritma Genetika, tidak selalu menjamin hasil yang lebih baik bahkan bisa berakibat buruk apabila ukuran populasinya sangat kecil. Sehingga dapat diambil suatu kesimpulan bahwa kinerja tahapan *crossover* pada algoritma genetika akan semakin baik bergantung pada banyaknya individu dalam suatu populasi. Namun tahapan tersebut menjadi kurang/tidak berfungsi apabila jumlah individu yang diberikan sebagai input adalah 1 (satu).



Gambar 2 Proses crossover pada populasi minimum

Gambar 2 memperlihatkan proses *crossover* yang dilakukan pada populasi minimum 1 (satu). Pada proses ini terlihat bahwa proses seleksi menghasilkan dua kromosom orang tua yang memiliki nilai gen sama. Sehingga jika dilakukan *crossover*, tidak terjadi perubahan pada kromosom anak yang dihasilkan. Berdasarkan temuan ini, maka dilakukan modifikasi Algoritma Genetika dengan mengeliminasi proses *crossover* yang menjadi inti permasalahan dari beberapa peneliti sebelumnya. Eliminasi proses *crossover* ini diharapkan melahirkan algoritma yang lebih efektif sebagai alternatif untuk penyelesaian permasalahan khususnya penjadwalan.

## 2. METODE PENELITIAN

### 2.1 Prosedur Penelitian

- a) Pengkajian Model Algoritma Genetika yang dimodifikasi untuk penyelesaian permasalahan penjadwalan pelajaran SMP.
- b) Analisa Data  
Tahap ini dimulai dengan pengumpulan data dari lokasi penelitian, yaitu SMPN 1 Telaga Gorontalo. Sehingga dihasilkan data-data yang dapat dianalisa untuk memperoleh masalah yang akan diselesaikan menggunakan algoritma genetika yang dimodifikasi. Tahapan ini akan menghasilkan input yang akan digunakan, *constraint*, dan model algoritma hasil modifikasi. Input yang dihasilkan adalah berupa kombinasi waktu dan alokasi yang kemudian akan di pasang dengan kombinasi pengajar, kelas, dan pelajaran yang diampunya.
- c) Pemodelan Algoritma Genetika yang dimodifikasi.
  - i. *Generate* Kromosom

Kromosom dibangkitkan menggunakan bilangan integer dimana data kelas yang telah dikombinasikan sebelumnya dengan pengajar dan matapelajaran yang telah dibagi berdasarkan jumlah jam matapelajaran akan diberikan input hari dan alokasi yang dibangkitkan secara acak.

Seperti halnya pada Algoritma Genetika, populasi pada modifikasi algoritma genetika berisi satu atau sejumlah kromosom yang ditentukan oleh masukan dari pengguna. Sebuah kromosom mewakili sebuah solusi jadwal dimana didalam kromosom tersebut terdapat sejumlah gen yang memuat data pengajar, matapelajaran, kelas dan jadwal dalam seminggu. Jumlah gen yang terdapat pada suatu kromosom pada penelitian ini ditentukan oleh banyaknya matapelajaran yang telah dibagi ke dalam 17 slot waktu dan banyaknya kelas yang akan dijadwalkan. Jumlah kelas dalam penelitian ini adalah 23. Sehingga panjang kromosom adalah  $17 \times 23 = 391$  gen.

	kelas	K1					K2					Kn								
		P1	P2	P3	P4	...	PnK1	P5	P6	P7	P8	...	PnK2	P9	P10	P11	P12	...	PnKn	
Kromosom 1	pengajar	A	B	C	D	...	L	A	B	C	D	...	C	...	A	B	C	D	...	C
	matapelajaran	1	1	1	2	...	6	1	1	1	2	...	6	...	1	1	1	2	...	6
	hari	1	2	3	1	...	1	1	2	3	1	...	1	...	1	2	3	1	...	1
	alokasi	1	2	3	1	...	1	1	2	3	1	...	1	...	1	2	3	1	...	1
Kromosom 2	kelas	K1					K2					Kn								
	pengajar	P1	P2	P3	P4	...	PnK1	P5	P6	P7	P8	...	PnK2	...	P9	P10	P11	P12	...	PnKn
	matapelajaran	A	B	C	D	...	L	A	B	C	D	...	C	...	A	B	C	D	...	C
	hari	1	1	1	2	...	6	1	1	1	2	...	6	...	1	1	1	2	...	6
alokasi	1	2	3	1	...	1	1	2	3	1	...	1	...	1	2	3	1	...	1	
...	...	K1					K2					Kn								
Kromosom (m)	kelas	K1					K2					Kn								
	pengajar	P1	P2	P3	P4	...	PnK1	P5	P6	P7	P8	...	PnK2	...	P9	P10	P11	P12	...	PnKn
	matapelajaran	A	B	C	D	...	L	A	B	C	D	...	C	...	A	B	C	D	...	C
	hari	1	1	1	2	...	6	1	1	1	2	...	6	...	1	1	1	2	...	6
alokasi	1	2	3	1	...	1	1	2	3	1	...	1	...	1	2	3	1	...	1	

Gambar 3 Contoh representasi kromosom

Gambar 3 menunjukkan contoh representasi kromosom dimana pada suatu kromosom terdiri atas gen K1, K2, ..., Kn yang mewakili sejumlah kelas, yang dikombinasikan dengan pengajar P1, P2, ..., Pn dan matapelajaran A, B, ..., L, dimana masing-masing kombinasi tersebut akan diberikan slot waktu berupa hari yang dikodekan dengan angka 1, 2, 3, 4, 5, dan 6 serta alokasi pada masing-masing hari yang terbagi atas alokasi 1, alokasi 2, dan alokasi 3.

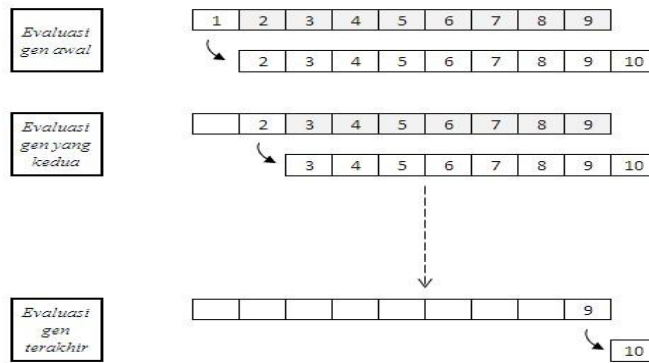
ii. Evaluasi Gen

Evaluasi gen dilakukan dengan mengidentifikasi gen pada kromosom yang dibangkitkan. Identifikasi dilakukan secara sequensial dari gen awal sampai pada gen akhir sehingga akan teridentifikasi jumlah gen rusak. Gen-gen cacat/ rusak yang teridentifikasi akan diperbaiki pada proses selanjutnya. Proses identifikasi dan perbaikan gen ini dilakukan secara berulang hingga mencapai kriteria berhenti. Proses ini akan berhenti jika: (1) gen cacat/ rusak bernilai nol/ mencapai nilai fitness 1, dan (2) gen cacat/ rusak tidak mengalami pengurangan/ tidak terjadi perubahan nilai fitness kromosom pada toleransi perulangan yang ditentukan.

Proses evaluasi dilakukan secara sekuensial berdasarkan batasan-batasan yang telah ditentukan sebelumnya. Batasan-batasan yang diberikan untuk penyelesaian permasalahan penjadwalan matapelajaran sekolah antara lain:

- a. Pengajar tidak boleh mengajar dua kelas pada waktu yang sama.
- b. Pengajar tidak boleh mengajar lebih dari sekali dalam sehari pada kelas yang sama

- c. Matapelajaran hanya boleh diajarkan sekali dalam sehari untuk setiap kelas.
- d. Setiap kelas hanya mendapatkan dua alokasi pada hari jumat dan tiga alokasi pada hari lainnya untuk matapelajaran berbeda.
- e. Matapelajaran olahraga tidak boleh menempati alokasi waktu siang hari.

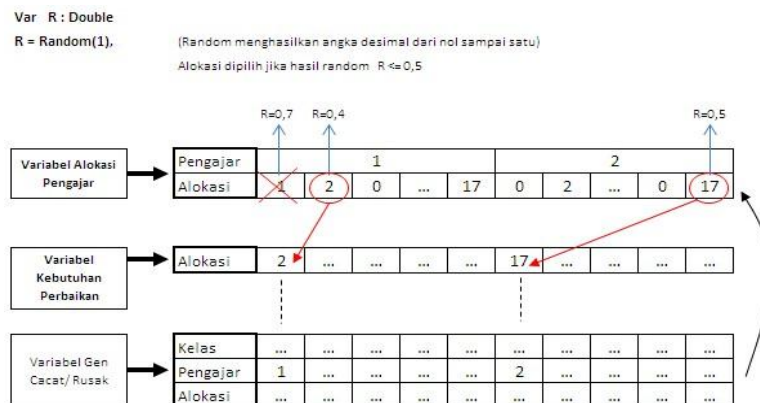


Gambar 4 Ilustrasi proses identifikasi gen cacat/ rusak

Gambar 4 memperlihatkan proses evaluasi yang dimulai dari gen awal sampai gen akhir. Proses ini dilakukan dengan cara membandingkan nilai suatu gen dengan semua nilai gen yang berada pada index berikutnya. Gen yang berada pada index 1 dibandingkan dengan gen yang berada pada index 2, index 3, dan seterusnya sampai index yang terakhir pada kromosom. Gen yang berada pada index 2 dibandingkan dengan gen yang berada pada index 3, index 4, dan seterusnya. Dengan kata lain, proses evaluasi gen ini dilakukan hingga mencapai index (n-1) sebanyak (n-1).

iii. Perbaiki Gen cacat/ rusak

Proses perbaikan gen cacat/ rusak diawali dengan persiapan kebutuhan perbaikan terlebih dahulu. Persiapan nilai kebutuhan perbaikan dimaksudkan untuk memperoleh nilai-nilai yang mungkin dapat digunakan untuk perbaikan gen cacat/ rusak. Proses pemilihan nilai yang akan digunakan sebagai nilai perbaikan gen dilakukan secara acak pada sejumlah nilai yang teridentifikasi memungkinkan untuk digunakan. Proses ini dilakukan sebanyak individu yang diproses.



Gambar 5 Ilustrasi proses pemilihan nilai kebutuhan perbaikan

Gambar 5 memperlihatkan proses pemilihan alokasi waktu yang tidak bernilai nol pada variabel alokasi pengajar berdasarkan variabel gen cacat/ rusak yang

teridentifikasi sebelumnya. Alokasi waktu terpilih kemudian dimasukkan kedalam variabel kebutuhan perbaikan dan menandai index variabel alokasi pengajar terpilih dengan angka nol. Proses pencarian alokasi secara acak dilakukan dengan cara menelusuri alokasi yang belum terpakai (tidak bernilai nol) pada variabel alokasi pengajar berdasarkan pengajar pada variabel gen cacat/ rusak. Proses ini dilakukan secara terus menerus hingga memperoleh alokasi kebutuhan perbaikan untuk semua gen cacat/ rusak yang teridentifikasi.

Proses pencarian alokasi dengan pemilihan secara acak pada variabel alokasi pengajar menghasilkan satu alokasi untuk setiap gen cacat/ rusak. Alokasi yang dihasilkan tersebut diujicobakan pada proses perbaikan gen cacat/rusak menggunakan kromosom duplikat. Jika perbaikan pada kromosom duplikat mengarah pada lingkup toleransi perbaikan yang diharapkan, maka hasil perbaikan akan diberikan pada kromosom aslinya.

d) Implementasi

Pada tahap ini akan dinyatakan sebuah algoritma yang dihasilkan melalui modifikasi Algoritma Genetika, kemudian diterapkan untuk menyelesaikan permasalahan penjadwalan SMP. Output yang dihasilkan adalah sebuah algoritma alternatif yang dapat menyelesaikan permasalahan penjadwalan SMP.

Proses penyelesaian masalah penjadwalan pelajaran sekolah dilakukan dengan cara mengolah data input yang dihasilkan melalui tahap analisis yang kemudian diproses menggunakan algoritma yang dihasilkan melalui modifikasi Algoritma Genetika dengan memberikan berbagai batasan yang dibutuhkan. Sehingga akan dihasilkan jadwal pelajaran yang sesuai dengan kebutuhan sekolah.

e) Pengujian

Pengujian dilakukan menggunakan perangkat lunak visual basic sebagai media implementasi algoritma hasil modifikasi untuk mengetahui tingkat akurasi dalam menyelesaikan permasalahan penjadwalan matapelajaran SMP.

## 2.2 Lokasi Penelitian

Lokasi Penelitian yang dilakukan adalah SMPN 1 Telaga Gorontalo yang memiliki 22 kelas dengan pembagian kelas 7 sejumlah 8, kelas 8 sejumlah 8, dan kelas 9 sejumlah 7.

## 3. HASIL DAN PEMBAHASAN

### 3.1 Modifikasi Algoritma Genetika

Modifikasi Algoritma Genetika pada penelitian ini merupakan modifikasi yang mengeliminasi proses crossover atas dasar temuan-temuan para peneliti sebelumnya. Beberapa peneliti menilai bahwa proses crossover tidak menjamin dapat menghasilkan solusi yang lebih baik dari solusi sebelumnya sehingga dilakukan modifikasi dengan menambahkan proses tertentu untuk meningkatkan kinerja Algoritma. Modifikasi Algoritma Genetikan pada penelitian ini selanjutnya akan diujikan pada studi kasus penjadwalan SMP Negeri 1 Telaga Gorontalo.

### 3.2 Pengujian Algoritma Genetika Hasil Modifikasi

Pengujian Modifikasi Algoritma Genetika pada penyelesaian permasalahan penjadwalan SMP Negeri 1 Telaga ini difokuskan pada pengaruh perubahan jumlah individu dan nilai toleransi perulangan terhadap perolehan nilai fitness dengan jumlah generasi tidak terbatas pada setiap percobaan. Variasi jumlah individu dan nilai toleransi perulangan yang diujikan, diperlihatkan pada Tabel 1.

Tabel 1 Variasi nilai toleransi perulangan dan jumlah individu

Pengujian	Jumlah Individu	Nilai Toleransi Perulangan
1	5	5 & 10
2	10	5 & 10
3	15	5 & 10
4	20	5 & 10

Pengujian berdasarkan perubahan jumlah individu dan nilai toleransi perulangan merupakan pengujian yang dilakukan secara berulang menggunakan jumlah individu dan nilai toleransi yang berbeda, tanpa membatasi jumlah generasi. Tujuan dari pengujian ini adalah mengetahui pengaruh jumlah individu dan nilai toleransi pada perubahan nilai fitness global dan perolehan solusi yang dihasilkan.

### 3.2.1 Pengujian 5 (lima) input

Pengujian 5 (lima) input merupakan pengujian kinerja algoritma genetika termodifikasi menggunakan parameter jumlah input 5 (lima). Hasil dari pengujian ini seperti terlihat pada Tabel 2 mencapai solusi terbaik sebesar 98,21% pada generasi ke-24.

Tabel 2 Hasil pengujian 5 (lima) input

No	Perbandingan	Nilai Toleransi 5
1	Pencapaian akhir fitness terbaik	0,982097186700767
2	Pencapaian generasi yang ditempuh	24

### 3.2.2 Pengujian 10 (sepuluh) input

Pengujian 10 (sepuluh) input merupakan pengujian kinerja algoritma genetika termodifikasi menggunakan parameter jumlah input 10 (sepuluh). Hasil dari pengujian ini seperti terlihat pada Tabel 3 mencapai solusi terbaik sebesar 99,23% pada generasi ke-32.

Tabel 3 Hasil pengujian 10 (sepuluh) input

No	Perbandingan	Nilai Toleransi 5
1	Pencapaian akhir fitness terbaik	0,9923273657289
2	Pencapaian generasi yang ditempuh	32

### 3.2.3 Pengujian 15 (lima belas) input

Pengujian 15 (lima belas) input merupakan pengujian kinerja algoritma genetika termodifikasi menggunakan parameter jumlah input 15 (lima belas). Hasil dari pengujian ini seperti terlihat pada Tabel 4 mencapai solusi terbaik sebesar 99,49% pada generasi ke-41.

Tabel 4 Hasil pengujian 15 (lima belas) input

No	Perbandingan	Nilai Toleransi 5
1	Pencapaian akhir fitness terbaik	0,994884910485934
2	Pencapaian generasi yang ditempuh	41

### 3.2.4 Pengujian 20 (dua puluh) input

Pengujian 20 (dua puluh) input merupakan pengujian kinerja algoritma genetika termodifikasi menggunakan parameter jumlah input 20 (dua puluh). Hasil dari pengujian ini seperti terlihat pada Tabel 5 mencapai solusi terbaik sebesar 99,74% pada generasi ke-47.

Tabel 5 Hasil pengujian 20 (dua puluh) input

No	Perbandingan	Nilai Toleransi 5
1	Pencapaian akhir fitness terbaik	0,997442455242967
2	Pencapaian generasi yang ditempuh	47
3	Hasil (%)	99,74

### 3.2.5 Pengujian nilai toleransi perulangan

Pengujian nilai toleransi dilakukan dengan melakukan perbandingan hasil pengujian input yang menggunakan nilai toleransi 5 (lima) pada pengujian sebelumnya dengan pengujian menggunakan jumlah input yang sama namun dengan nilai toleransi yang berbeda. Contoh pengujian yang akan dibandingkan adalah pengujian 5 (lima) dengan hasil pada Tabel 2. Perbandingan hasil yang diperoleh melalui penggunaan 5 (lima) input dengan nilai toleransi 5 (lima) dan nilai toleransi 10 (sepuluh) diperlihatkan pada Tabel 6.

Tabel 6 Perbandingan hasil menggunakan nilai toleransi 5 dan 10.

No	Perbandingan	Nilai Toleransi 5 (5 individu)	Nilai Toleransi 10 (5 individu)
1	Pencapaian akhir fitness terbaik	0,982097186700767	0,9923273657289
2	Pencapaian generasi yang ditempuh	24	63
3	Hasil (%)	98,21	99,23

### 3.3 Perbandingan Algoritma Hasil Modifikasi dengan Algoritma Genetika

Perbandingan hasil yang akan dilakukan adalah perbandingan hasil menggunakan Algoritma Genetika dengan hasil yang telah dilakukan pada penelitian ini. Penentuan jumlah individu dan jumlah generasi dalam perbandingan disesuaikan dengan data contoh hasil pengujian pada Tabel 5. Persentase hasil dari beberapa percobaan yang dilakukan untuk perbandingan tingkat akurasi yang diperoleh, ditunjukkan pada Tabel 7 yang disimbolkan P1 (percobaan ke-1), P2 (percobaan ke-2), dan P3 (percobaan ke-3).



Tabel 7 Perbandingan hasil Algoritma Genetika dan Modifikasi Algoritma Genetika

No	Perbandingan	Modifikasi Algoritma Genetika			Algoritma Genetika		
		P1	P2	P3	P1	P2	P3
1	Jumlah Individu	20	20	20	20	20	20
2	Maksimum generasi yang dicapai	47	41	40	50	50	50
5	Persentase	<b>99,74%</b>	<b>99,74%</b>	99,23%	96,68%	96,16%	<b>97,44%</b>

#### 4. KESIMPULAN

Berdasarkan hasil pembahasan pada bab sebelumnya, maka hal-hal yang dapat disimpulkan pada penelitian ini adalah:

1. Semakin kecil jumlah individu yang digunakan pada Modifikasi Algoritma Genetika, maka semakin cepat proses pencapaian solusi yang dihasilkan. Namun solusi yang dihasilkan tidak lebih baik dari proses pencapaian menggunakan jumlah individu yang lebih besar. Proses yang menggunakan 5 (lima) individu yang ditunjukkan pada Tabel 2 berhasil mencapai solusi terbaik pada generasi ke-24, sedangkan untuk jumlah individu yang lebih besar, mencapai solusi pada generasi ke-32 (Tabel 3), generasi ke-41 (Tabel 4), dan generasi ke-47 (Tabel 5). Namun jumlah individu yang lebih besar memberikan hasil pencapaian solusi yang lebih baik dibandingkan dengan hasil yang dicapai oleh proses yang menggunakan jumlah individu yang lebih kecil.
2. Besarnya nilai toleransi perulangan yang digunakan pada penelitian ini ditujukan untuk memberikan waktu/ kesempatan yang lebih lama untuk mencapai solusi yang lebih baik. Hasil perbandingannya pada Tabel 6 menunjukkan pencapaian solusi yang lebih baik pada penggunaan nilai toleransi yang lebih besar walaupun jumlah generasi yang dicapai lebih besar.
3. Modifikasi yang dihasilkan melalui eliminasi proses crossover pada Algoritma Genetika memberikan hasil pencapaian solusi yang lebih baik dengan perbandingan menggunakan jumlah individu dan jumlah generasi yang sama. Berdasarkan data hasil perbandingan persentase hasil pencapaian terbaik dari masing-masing algoritma yg ditunjukkan pada Tabel 7, algoritma genetika termodifikasi mencapai **2,30%** lebih baik dibandingkan dengan algoritma genetika sederhana untuk solusi yang dihasilkan.
4. Modifikasi Algoritma Genetika yang mengeliminasi proses *crossover* memiliki perbedaan yang sangat besar dengan Algoritma Genetika. Perbedaan terbesarnya adalah Algoritma Genetika mengadopsi proses evolusi alam sedangkan algoritma modifikasinya tidak. Algoritma hasil modifikasi lebih mengarah pada teknik terapi gen untuk pengobatan melalui perbaikan gen-gen cacat yang bertanggung jawab terhadap suatu penyakit.
5. Modifikasi Algoritma Genetika pada penelitian ini dapat dijadikan sebagai metode alternatif dalam menyelesaikan permasalahan penjadwalan sekolah.

## 5. SARAN

Mengingat keterbatasan waktu eksplorasi dalam penelitian ini, maka diperlukan saran yang dapat membantu pengembangan Algoritma ini agar menjadi lebih baik, diantaranya adalah:

1. Kecilnya hasil peningkatan yang terjadi pada algoritma hasil modifikasi, membutuhkan penelitian lebih lanjut untuk penyempurnaannya baik dari sisi efisiensi maupun pemodelannya.
2. Tahapan algoritma pada penelitian ini yang mengarah pada definisi terapi gen, belum sempat disesuaikan dengan penerapan terapi gen secara nyata, sehingga butuh kajian lebih lanjut untuk menyempurnakannya sebagai Algoritma Terapi Gen.
3. Untuk penelitian selanjutnya, diharapkan dapat diujikan pada kasus-kasus yang berbeda.

## UCAPAN TERIMA KASIH

Puji syukur penulis panjatkan ke hadirat Allah SWT yang telah memberikan karunia bagi penulis sehingga dapat menyelesaikan penelitian ini. Terimakasih juga penulis ucapkan kepada semua pihak atas segala dukungan yang telah diberikan, terutama kepada Bapak Drs. Retantyo Wardoyo, M.Sc, Ph.D selaku pembimbing, untuk segala arahan dan motivasi yang telah diberikan.

## DAFTAR PUSTAKA

- [1] Abramson, D., and Abela, J., 1992, A Parallel Genetic Algorithm For Solving The School Timetabling Problem, *15 Australian Computer Science Conference (ACSC-15)*, 29 – 31 Januari, pp. 1-11.
- [2] Bambrick, L., 1997, *Lecture Timetabling Using Genetic Algorithms*, Thesis, University of Queensland.
- [3] Beligiannis,GN., Moschopoulos, C., and Likothanassis, S.D., 2009, A Genetic Algorithm Approach To School Timetabling, *Journal of the Operational Research Society* 60, pp. 23-42.
- [4] Pillay,N., and Banzhaf, W., 2010, An informed genetic algorithm for the examination timetabling problem, *Applied Soft Computing*, Vol.10, No.2, Pp. 457–467.
- [5] Raghavjee, R., and Pillay, N., 2010, Using genetic algorithms to solve the South African school timetabling problem, *Nature and Biologically Inspired Computing (NaBIC)*, 15 – 17 Desember, pp. 286-292.
- [6] Chorbev, I., Loskovska, S., Dimitrovski, I., and Mihajlov, D., (2008), *Solving the High School Scheduling Problem Modelled with Constraints Satisfaction Using Hybrid Heuristic Algorithms*, Greedy Algorithms, Witold Bednorz (Ed.), ISBN: 978-953-7619-27-5, InTech, Available from:  
[http://www.intechopen.com/books/greedy\\_algorithms/solving\\_the\\_high\\_school\\_scheduling\\_problem\\_modelled\\_with\\_constraints\\_satisfaction\\_using\\_hybrid\\_heuri](http://www.intechopen.com/books/greedy_algorithms/solving_the_high_school_scheduling_problem_modelled_with_constraints_satisfaction_using_hybrid_heuri) [akses: 27 Maret 2012].
- [7] Al-Milli, N.R., 2010, Hybrid Genetic Algorithms with Great Deluge For Course Timetabling, *IJCSNS International Journal of Computer Science and Network Security*, Vol. 10, No. 4, Pp. 283-288.
- [8] Mehdi B, and Mohamad A. M., 2011, Hybridizing Genetic Algorithm with Biased Chance Local Search, *World Academy of Science, Engineering and Technology*, Vol. 81, No.57, pp. 415-420.
- [9] Suyanto, 2005, *Algoritma Genetika dalam MATLAB*, Penerbit Andi, Yogyakarta.