

Pengembangan Sistem Penjadwalan Perkuliahan Menggunakan Algoritma Genetika Berbasis Web

Syaiful Akbar Abdullah Kusing¹, Tikaridha Hardiani²

*Teknologi Informasi, Fakultas Sains & Teknologi, Universitas 'Aisyiyah Yogyakarta
Jl. Siliwangi (Riang Road Barat) No.63 Nogotirto, Gamping, Sleman, Yogyakarta.*

¹syiafulkusing14@gmail.com

²tikaridha@unisayogya.ac.id

Abstract— Course scheduling is a complex optimization problem involving multiple academic components and constraints. This study implements a genetic algorithm to develop an automated, web-based course scheduling system that minimizes scheduling conflicts and constraint violations. Scheduling solutions are represented as chromosomes and evaluated using a penalty-based fitness function that considers lecturer conflicts, room utilization, and academic constraints. The optimization process consists of selection, crossover, mutation, and elitism stages. Experimental results demonstrate a gradual improvement in fitness values and the generation of a final schedule with near-zero conflicts, indicating that the proposed approach can produce schedules that better satisfy academic and operational requirements.

Keywords—course scheduling, genetic algorithm, seleksi, persilangan, mutasi, elitisme, Laravel

Abstrak— Penjadwalan mata kuliah merupakan permasalahan optimasi yang kompleks karena melibatkan berbagai komponen akademik dan sejumlah kendala. Penelitian ini menerapkan algoritma genetika untuk mengembangkan sistem penjadwalan mata kuliah berbasis web secara otomatis dengan meminimalkan konflik jadwal dan pelanggaran kendala. Solusi penjadwalan direpresentasikan dalam bentuk kromosom dan dievaluasi menggunakan fungsi fitness berbasis penalti yang mempertimbangkan konflik dosen, penggunaan ruang, dan kendala akademik. Proses optimasi dilakukan melalui tahapan seleksi, persilangan, mutasi, dan elitisme. Hasil eksperimen menunjukkan adanya peningkatan nilai fitness secara bertahap serta menghasilkan jadwal akhir dengan tingkat konflik yang mendekati nol. Hasil tersebut menunjukkan bahwa pendekatan yang diusulkan mampu menghasilkan jadwal yang lebih sesuai dengan kebutuhan operasional dan akademik.

Kata kunci— penjadwalan perkuliahan, algoritma genetika, seleksi, persilangan, mutasi, elitisme, laravel

I. PENDAHULUAN

Penjadwalan mata kuliah merupakan komponen krusial dalam manajemen akademik perguruan tinggi karena berkaitan langsung dengan efisiensi pemanfaatan

sumber daya, seperti ruang kelas dan dosen, serta kualitas pengalaman belajar mahasiswa [1]. Kompleksitas penjadwalan meningkat seiring bertambahnya jumlah mata kuliah, variasi kurikulum, keterbatasan ruang, kebutuhan praktikum dan teori, serta ketersediaan dosen [2]. Pada beberapa perguruan tinggi dengan proses penjadwalan masih dilakukan secara manual atau semi-manual rentan terhadap konflik jadwal, ketidaksesuaian kapasitas ruang, dan ketidaktepatan alokasi waktu, yang pada akhirnya berdampak pada efektivitas kegiatan akademik [3].

Program Studi Teknologi Informasi Universitas 'Aisyiyah Yogyakarta proses penjadwalan juga masih dilakukan secara manual atau semi-manual, sehingga sering menimbulkan berbagai permasalahan seperti konflik dosen yang mengajar pada waktu bersamaan, benturan penggunaan ruangan, ketidaksesuaian kapasitas ruang, serta pelanggaran aturan akademik lainnya [4]. Kondisi ini berdampak pada rendahnya efektivitas operasional akademik dan sering kali memerlukan penjadwalan ulang yang memakan waktu dan sumber daya tambahan [5].

Berbagai penelitian terdahulu telah mengkaji pemanfaatan pendekatan komputasional untuk menyelesaikan permasalahan penjadwalan mata kuliah salah satu metode yang banyak digunakan dalam beberapa tahun ini adalah algoritma genetika (GA). Algoritma genetika adalah teknik komputasi terinspirasi oleh prinsip seleksi alam dan evolusi biologi, pertama kali diusulkan oleh John Holland melalui bukunya yang berjudul *"Adaptation in Natural and Artificial Systems"*, di Universitas Michigan (1975) [6].

Pendekatan algoritma genetika telah digunakan di berbagai penelitian karena kemampuannya dalam menangani permasalahan optimasi dengan ruang solusi yang besar dan kendala yang kompleks [7]. Algoritma genetika menunjukkan perbedaan sebagai salah satu metode pencarian yang memiliki kemampuan memulai dengan kondisi solusi dalam bentuk acak disebut sebagai populasi. Individu-individu yang terdapat di dalam populasi ini disebut kromosom yang kemudian merepresentasikan alternatif dari masalah [8]. Penerapan algoritma genetika pada penjadwalan mata kuliah terbukti

mampu menurunkan tingkat konflik dan menghasilkan solusi yang lebih optimal dibandingkan metode konvensional [9].

Penelitian Rohim et al. [10] menunjukkan bahwa algoritma genetika efektif dalam menangani konflik dosen, konflik ruangan, serta konflik waktu perkuliahan berhasil dikurangi hingga mencapai nol konflik dalam kurun waktu 1-5 menit, selain mempercepat waktu penyusunan jadwal juga memberikan hasil jadwal yang lebih optimal. Sementara itu penelitian Liyan [9] menyatakan penjadwalan dengan nilai *fitness* 1 dalam waktu 0,609 detik, namun penelitian lain [11] juga mengungkapkan bahwa dalam mengintegrasikan berbagai aspek yang dimana menghasilkan kromosom yang lebih banyak cenderung membutuhkan iterasi serta waktu yang lebih tinggi, sehingga perlu untuk penyesuaian dalam konfigurasi agar algoritma dapat menemukan solusi lebih baik dan konsisten.

Meskipun berbagai penelitian sebelumnya menunjukkan bahwa algoritma genetika mampu menyelesaikan permasalahan penjadwalan secara efektif, sebagian besar penelitian masih berfokus pada simulasi perhitungan algoritma atau pengujian pada dataset terbatas. Selain itu, belum banyak penelitian yang mengimplementasikan algoritma genetika ke dalam sistem informasi berbasis web dengan mempertimbangkan kebutuhan riil, seperti penjadwalan mata kuliah teori dan praktikum, kapasitas ruangan, konflik dosen, serta ketersediaan waktu perkuliahan. Oleh karena itu, diperlukan pengembangan sistem penjadwalan yang tidak hanya mengutamakan optimasi solusi, tetapi juga mampu diterapkan secara operasional dalam lingkungan perguruan tinggi.

Penelitian ini bertujuan merancang dan membangun Sistem Informasi Penjadwalan Mata Kuliah berbasis web menggunakan metode algoritma genetika untuk membantu proses penyusunan jadwal perkuliahan secara otomatis. Sistem yang dikembangkan diharapkan mampu meminimalkan konflik dosen, benturan penggunaan ruangan, serta pelanggaran constraint penjadwalan lainnya, sehingga menghasilkan jadwal yang lebih optimal, efisien, dan sesuai dengan kebutuhan akademik di Program Studi Teknologi Informasi Universitas 'Aisyiyah Yogyakarta.

II. METODE PENELITIAN

A. Pendekatan Penelitian

Penelitian ini menggunakan pendekatan rekayasa perangkat lunak dan eksperimen komputasional dengan algoritma genetika (GA) sebagai metode optimasi utama. Objek penelitian berupa sistem penjadwalan mata kuliah berbasis web yang dikembangkan menggunakan *framework* Laravel (lihat Gambar 1).



Gambar. 1 Alur Penelitian

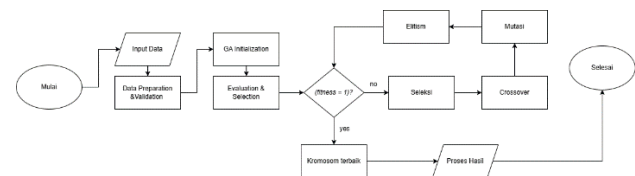
B. Proses Pengumpulan Data

Proses Pengumpulan data dilakukan dengan observasi langsung pada sistem Lensa Unisa dan SIMPTT untuk mengekstraksi informasi yang diperlukan guna membentuk dataset awal yang akan diolah oleh algoritma genetika (GA). data yang bersumber dari Learning Management System (LMS) Lensa Unisa dan Sistem Informasi Manajemen Perguruan Tinggi Terpadu (SIMPTT) meliputi data mata kuliah, dosen, jumlah mahasiswa, jenis perkuliahan (teori dan praktikum), ruang kelas dan laboratorium, serta slot waktu perkuliahan.

C. Algoritma Genetika

Algoritma genetika ini bekerja dengan mengembangkan sekumpulan solusi awal (populasi) yang kemudian diperbaiki secara bertahap untuk memperoleh solusi terbaik atau mendekati optimal [12].

Implementasi dilakukan melalui beberapa tahapan utama, yaitu inisialisasi populasi, evaluasi *fitness*, seleksi, persilangan, mutasi, dan elitisme (lihat Gambar 2). Setiap solusi jadwal direpresentasikan sebagai kromosom yang terdiri dari sejumlah gen [13], di mana satu gen merepresentasikan satu sesi perkuliahan dengan atribut hari, jam, ruangan, dan jenis sesi.



Gambar. 1 Alur Proses Algoritma Genetika

1) *Input Data*: Data ini merupakan komponen awal yang sangat menentukan kualitas solusi jadwal yang dihasilkan oleh algoritma genetika [14]. Pada sistem penjadwalan mata kuliah berbasis web yang dikembangkan, seluruh data yang input dimasukkan oleh pengguna melalui antarmuka sistem dan disimpan dalam basis data untuk selanjutnya diproses oleh algoritma genetika.

2) *Data Preparasi & Validasi*: Fase ini adalah fondasi dari keseluruhan proses scheduling karena data yang buruk akan menghasilkan jadwal yang tidak optimal atau bahkan error. Data input merupakan merepresentasikan gen dengan id 1 yang berisi matakuliah, dosen, periode, dan jumlah mahasiswa.

TABEL I
REPRESENTASI GEN

Id	Jadwal
1	MK1, D1, P1, 50
2	MK2, D2, P1, 50
3	MK3, D1, P1, 50

Setelah proses ini setiap mata kuliah yang belum memiliki penjadwalan telah direpresentasikan sebagai satu *gen*, kemudian *hari_id*, *jam_id*, ruangan masing-masing

direpresentasikan sebagai *gen* lainnya untuk melengkapi *gen* jadwal. Sekumpulan *gen* yang membentuk jadwal lengkap untuk seluruh mata kuliah direpresentasikan sebagai satu *kromosom*. Dengan demikian, satu *kromosom* mencerminkan satu solusi jadwal utuh dalam algoritma genetika ditunjukkan pada Tabel II.

TABEL III
REPRESENTASI KROMOSOM

Kromosom	Gen 1				Gen2	Gen3	Gen4
	MK	Dose n	Periode	Jlm_mhs	Hari	Jam	Ruangan
1	MK1	D1	P1	50	H01	J01	R1

Setelah terbentuk *kromosom* dimana setiap *kromosom* mewakili setiap jadwal matakuliah selanjutnya dibentuk sebuah *individu* yang berisi banyak *kromosom* untuk seluruh jadwal matakuliah yang ada pada periode tertentu ditunjukkan pada Tabel III.

TABEL IIIII
REPRESENTASI INDIVIDU

Individu 1							
Kromosom	Gen 1				Gen2	Gen3	Gen4
	Mata kuliah	Dosen	Periode	Jumlah_mhs	Hari	Jam	Ruangan
1	MK1	D1	P1	50	H01	J01	R1
2	MK2	D2	P1	50	H02	J02	R2
3	MK3	D3	P1	50	H03	J03	R3
4	MK4	D3	P1	50	H04	J04	R3
5	MK5	D5	P1	50	H05	J05	R2

Pada tahap preparation, dilakukan proses filtering dan pengelompokan data untuk mengurangi kemungkinan pelanggaran *constraint* sejak awal. Data ruangan dipisahkan berdasarkan tipe ruangan, yaitu ruangan kelas untuk kuliah teori dan ruangan laboratorium untuk kuliah praktikum. Selain itu, data jam perkuliahan difilter untuk menghilangkan jam istirahat dan jam yang tidak diperbolehkan secara akademik, seperti interval 12.00–13.00 dan 15.00–16.00, serta pembatasan jam akhir perkuliahan.

Validasi juga dilakukan berdasarkan karakteristik mata kuliah, khususnya jumlah SKS dan jenis perkuliahan. Mata kuliah 3 SKS hanya diperbolehkan menggunakan jam tertentu yang sesuai dengan durasi perkuliahan, kemudian mata kuliah 2 SKS teori dan praktikum juga memiliki aturan jam yang berbeda. Pembatasan ini bertujuan untuk memastikan bahwa setiap *individu* yang terbentuk secara struktural telah memenuhi *constraint* dasar sebelum dievaluasi pada fungsi *fitness*.

Pendekatan ini bertujuan untuk mencegah terbentuknya *kromosom* yang tidak valid dan mengurangi beban penalti pada tahap evaluasi *fitness*. Data preparation dan validasi pada tahap pembentukan *gen*, *kromosom* dan *individu* dilakukan agar algoritma genetika dapat bekerja lebih efisien. Proses evolusi solusi menjadi lebih terarah karena populasi awal sudah berada dalam ruang solusi yang layak secara akademik, sehingga meningkatkan peluang menuju jadwal perkuliahan yang optimal.

3) *Inisialisasi*: Inisialisasi membangkitkan sejumlah solusi jadwal awal secara acak. Setiap solusi

direpresentasikan sebagai individu yang terdiri dari kumpulan *kromosom*, di mana satu *kromosom* merepresentasikan satu mata kuliah beserta atribut penjadwalannya, meliputi hari, jam, dan ruangan. Pada tahap ini, pembangkitan jadwal acak tetap memperhatikan *constraint* akademik dasar, seperti pengurutan hari dari Senin hingga Jumat, kesesuaian jenis ruangan dengan jenis perkuliahan (teori atau praktikum), serta penyesuaian slot waktu berdasarkan jumlah SKS. Untuk mata kuliah praktikum dengan jumlah mahasiswa melebihi kapasitas, sistem membagi perkuliahan ke dalam beberapa sesi tanpa terjadi tumpang tindih waktu. Setiap *kromosom* yang terbentuk kemudian dievaluasi menggunakan fungsi *fitness* sebagai dasar proses seleksi pada generasi berikutnya.

4) *Evaluasi & Seleksi*: Evaluasi kualitas solusi pada penelitian ini dilakukan menggunakan fungsi *fitness* yang merepresentasikan tingkat optimalitas suatu jadwal. Nilai *fitness* dihitung berdasarkan total penalti yang berasal dari konflik dan pelanggaran *constraint* akademik dengan persamaan sebagai berikut:

$$\text{fitness} = 1/(1+\text{totalPenalty})$$

Nilai totalPenalty merupakan akumulasi bobot penalti dari setiap konflik dan pelanggaran *constraint* yang terjadi pada solusi jadwal. Semakin kecil nilai totalPenalty, maka semakin besar nilai *fitness* yang dihasilkan, sehingga solusi tersebut dinilai semakin optimal dapat dilihat pada Tabel IV.

TABEL IV
PENALTY & FITNESS

Tahapan	Individu	Total Penalty	Fitness	Keterangan
Populasi Awal	1	25	0,0385	Solusi awal dengan total penalti dan nilai fitness
	2	15	0,0667	
	3	40	0,0243	
	4	60	0,0163	
	5	35	0,0287	

Tabel *populasi* menunjukkan variasi kualitas solusi per *individu* terlihat bahwa setiap *individu* memiliki nilai total penalti dan *fitness* yang berbeda-beda. Individu dengan total penalti lebih kecil menghasilkan nilai *fitness* yang lebih tinggi, yang menandakan solusi jadwal tersebut lebih mendekati kondisi optimal. Sebaliknya, *individu* dengan total penalti besar memiliki nilai *fitness* yang rendah karena masih mengandung banyak konflik jadwal.

5) *Seleksi, Persilangan, Mutasi, Elitisme*: Berdasarkan hasil evaluasi *fitness* pada populasi awal di atas, proses seleksi selanjutnya dilakukan dengan memilih dua individu dengan nilai *fitness* tertinggi sebagai induk. Individu terpilih tersebut kemudian digunakan pada tahap persilangan untuk menghasilkan solusi baru yang diharapkan memiliki kualitas jadwal yang lebih baik pada generasi berikutnya (lihat Tabel V).

TABEL V
SELEKSI, PERSILANGAN, MUTASI, ELITISME

Tahapan		Fitness	Keterangan
Seleksi	Induk 1	0,0667	Fitness tertinggi
	Induk 2	0,0385	Fitness tertinggi ke-2
Persilangan	Keturunan	0,0567	Hasil kombinasi Gen Induk
Mutasi	Keturunan(mutasi)	0,0712	Hasil perbaikan jadwal
Elitisme	Kromosom terbaik	0,0712	Dipertahankan ke generasi berikutnya

Setelah proses seleksi, dua individu dengan nilai fitness tertinggi dipilih sebagai induk untuk membentuk generasi baru. Tahap *persilangan* dilakukan dengan menukar sebagian gen antar induk guna menghasilkan *individu* baru (keturunan). Pertukaran gen ini bertujuan untuk mengombinasikan karakteristik jadwal yang baik dari masing-masing induk, seperti alokasi waktu yang bebas konflik atau pemilihan ruangan yang sesuai.

Melalui proses *persilangan*, struktur kromosom tetap dipertahankan agar setiap gen tetap merepresentasikan satu sesi perkuliahan yang valid. Dengan demikian, keturunan yang dihasilkan memiliki peluang lebih besar untuk memperoleh kualitas solusi yang lebih baik dibandingkan solusi pada populasi awal.

Selanjutnya, tahap *mutasi* diterapkan pada kromosom hasil *persilangan* dengan probabilitas tertentu. *mutasi* berfungsi untuk menambahkan variasi solusi dan mencegah algoritma terjebak pada kondisi konvergensi prematur. Bentuk *mutasi* yang dilakukan meliputi perubahan alokasi hari, jam, atau ruangan pada satu atau beberapa gen jadwal, sehingga memungkinkan munculnya solusi alternatif yang sebelumnya tidak terbentuk.

Untuk menjaga kualitas solusi terbaik, mekanisme *elitisme* diterapkan dengan mempertahankan sejumlah kromosom dengan nilai *fitness* tertinggi agar langsung dibawa ke generasi berikutnya tanpa mengalami perubahan. Dengan demikian, solusi terbaik yang telah diperoleh tidak hilang selama proses evolusi berlangsung.

```

FOR generation = 1 TO max_generations (500):
  Evaluasi fitness semua kromosom
  ↳ best_fitness = max(fitness)

  IF best_fitness == 1.0:
    ↳ FOUND OPTIMAL SOLUTION → BREAK

  Seleksi parent (top 2)

  Buat generasi baru:
  ↳ Elitism: Copy top 20 kromosom (unchanged)
  ↳ FOR i = 20 TO populationSize:
    ↳ Crossover parent → child
    ↳ Mutasi child dengan prob 15%
    ↳ Masukkan ke populasi baru
    ↳ populasi = populasi_baru

  Log best_fitness setiap 10 generasi
  
```

Gambar. 2 Perulangan Evolusi

Pada Gambar 3, rangkaian proses *seleksi*, *persilangan*, *mutasi*, dan *elitisme* ini dilakukan secara berulang pada

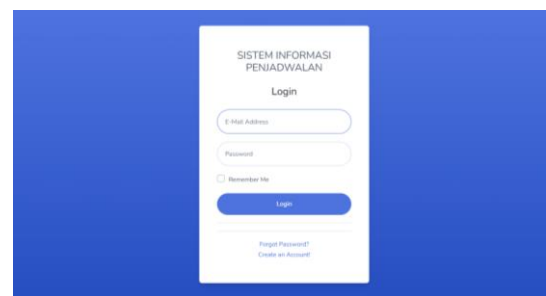
setiap generasi hingga diperoleh peningkatan nilai *fitness* hingga mencapai kondisi berhenti ($\text{best_fitness} == 1.0$;). Melalui mekanisme tersebut, algoritma genetika secara bertahap mampu memperbaiki kualitas jadwal dan menghasilkan solusi penjadwalan mata kuliah yang lebih optimal.

III. HASIL & PEMBAHASAN

A. Implementasi Sistem

1) Halaman Login Sistem

Gambar 4 memperlihatkan halaman login menyediakan kolom email atau username dan password.



Gambar. 4 Halaman Login

2) Halaman Matakuliah

Menampilkan Matakuliah dan atribut lainnya serta Admin dapat melakukan CRUD, dapat dilihat pada Gambar 5.

Gambar. 5 Contoh resolusi gambar yang diterima

3) Halaman Dosen

Halaman Dosen menampilkan NIP dan Nama Dosen seperti pada Gambar 6.

Gambar. 6 Halaman Dosen

4) Halaman Ruang

Menampilkan kode_ruangan, nama_ruangan, kapasitas serta type_ruang. Tipe ruangan terbagi 2

macam yaitu ruang kelas untuk kuliah teori dan lab komputer untuk kegiatan praktikum dapat dilihat pada Gambar 7.

Gambar 7. Halaman Ruang.

5) Halaman Data Jadwal

Halaman ini menampilkan matakuliah beserta atribut Dosen, Periode dan Jumlah mahasiswa dapat dilihat pada Gambar 8.

Gambar 8. Halaman Data jadwal

6) Halaman Hasil Jadwal

Halaman hasil jadwal menampilkan daftar hasil jadwal dengan periode, best fitness, tanggal generate dan jumlah jadwal dan aksi, dapat dilihat pada Gambar 9.

Gambar 9. Halaman Hasil Jadwal.

7) Halaman Detail Hasil Jadwal

Halaman ini menampilkan matakuliah beserta dosen, hari, jam, ruangan, tipe_jadwal dan jumlah mahasiswa. Ruangan disesuaikan dengan tipe jadwal teori atau praktikum dilihat pada Gambar 10.

Gambar 10. Halaman Detail Hasil Jadwal.

B. Hasil Algorithm

Proses evolusi algoritma genetika yang telah dilakukan, diperoleh satu *individu* terbaik yang merepresentasikan solusi jadwal mata kuliah dengan nilai *fitness* tertinggi. *Individu* ini dipilih sebagai hasil akhir dengan jumlah konflik dan pelanggaran constraint yang sangat minimal dapat dilihat pada Gambar 11.

```

{
  "success": true,
  "hasil_jadwal_id": "52",
  "best_fitness": 1,
  "analysis": {
    "conflict": 0,
    "constraintViolation": 0,
    "totalPenalty": 0,
    "fitness": 1,
    "breakdown": {
      "ruangan_salah": 0,
      "jam_praktikum_salah": 0,
      "jam_teoris_salah": 0,
      "jamistirahat": 0,
      "kapasitas_lab": 0,
      "teori_praktikum_minggu": 0,
      "praktikum_split_overlap": 0,
      "konflik_dosen": 0,
      "konflik_ruangan": 0
    }
  },
  "solution_count": 59
}

```

Gambar 11. Detail Hasil Jadwal Analisis

Analisis lebih lanjut pada bagian *breakdown* menunjukkan bahwa seluruh jenis pelanggaran constraint bernilai 0. Tidak ditemukan kesalahan dalam pemilihan ruangan (*ruangan_salah*), penjadwalan praktikum dan teori (*jam_praktikum_salah* dan *jam_teoris_salah*), pelanggaran waktu istirahat, maupun pelanggaran kapasitas laboratorium. Selain itu, tidak terjadi konflik dosen maupun konflik penggunaan ruangan pada waktu yang sama. Hasil ini menunjukkan bahwa seluruh aturan akademik dan teknis yang ditetapkan pada tahap metode telah terpenuhi secara menyeluruh.

Nilai *solution_count* sebesar 59 menunjukkan jumlah solusi yang berhasil dievaluasi selama proses evolusi algoritma genetika. Dari keseluruhan solusi tersebut, sistem mampu mengidentifikasi satu solusi terbaik dengan kualitas tertinggi. Hasil ini membuktikan bahwa algoritma genetika efektif dalam mengeksplorasi ruang solusi dan mengonvergensi pada jadwal perkuliahan yang optimal dan layak diterapkan dalam operasional akademik.

C. Analisis Proses Optimasi dan Fitness

Hasil penelitian diperoleh dari proses eksekusi algoritma genetika pada sistem penjadwalan mata kuliah berbasis web yang telah dikembangkan. Algoritma dijalankan dengan parameter populasi, generasi, mutasi,

dan elitisme yang telah ditentukan pada tahap metode. Setiap individu dalam populasi merepresentasikan satu solusi jadwal lengkap yang dievaluasi menggunakan fungsi fitness berdasarkan jumlah konflik dan pelanggaran constraint akademik.

Pada populasi awal, solusi jadwal dibangkitkan secara acak sehingga masih ditemukan banyak konflik dosen, konflik penggunaan ruangan, serta ketidaksesuaian aturan waktu dan jenis ruangan. Hal ini tercermin dari nilai total penalti yang relatif tinggi dan nilai fitness yang rendah. Seiring bertambahnya generasi, kualitas solusi mengalami peningkatan melalui mekanisme seleksi, persilangan, mutasi, dan elitisme.

Evaluasi terhadap populasi awal menunjukkan adanya variasi kualitas solusi, di mana individu dengan total penalti lebih kecil menghasilkan nilai fitness yang lebih tinggi. Individu-individu dengan fitness terbaik kemudian dipilih sebagai induk untuk membentuk generasi berikutnya. Proses ini berlangsung secara iteratif hingga diperoleh solusi dengan nilai fitness tertinggi yang stabil.

Pada generasi menengah hingga akhir, peningkatan nilai fitness mulai stabil. Kondisi ini menunjukkan bahwa algoritma telah mendekati solusi optimal. Penerapan elitisme berperan penting dalam menjaga solusi terbaik agar tidak hilang, sementara mutasi membantu mencegah stagnasi solusi pada nilai fitness tertentu. Hasil ini menunjukkan bahwa kombinasi seleksi, persilangan, mutasi, dan elitisme mampu mengarahkan proses evolusi menuju solusi jadwal yang lebih optimal.

D. Hasil Seleksi, Persilangan, dan Mutasi

Berdasarkan hasil evaluasi fitness pada populasi awal, dua individu dengan nilai fitness tertinggi dipilih sebagai induk pada tahap seleksi. Individu terpilih ini memiliki jumlah konflik dan pelanggaran constraint yang lebih sedikit dibandingkan individu lainnya.

Tahap persilangan dilakukan dengan menukar sebagian gen antar induk untuk menghasilkan individu baru (keturunan). Proses ini memungkinkan penggabungan karakteristik jadwal yang baik, seperti alokasi waktu yang tidak bertabrakan dan pemilihan ruangan yang sesuai. Keturunan yang dihasilkan kemudian dievaluasi kembali menggunakan fungsi fitness.

Selanjutnya, proses mutasi diterapkan dengan probabilitas tertentu untuk mengubah nilai gen secara acak, seperti perubahan hari, jam, atau ruangan perkuliahan. Mutasi berperan dalam meningkatkan keberagaman solusi dan memberikan peluang bagi algoritma untuk menemukan kombinasi jadwal yang lebih baik. Hasil evaluasi menunjukkan bahwa sebagian keturunan memiliki nilai fitness yang lebih tinggi dibandingkan induk, menandakan adanya perbaikan kualitas solusi.

E. Hasil Jadwal Akhir dan Dampak Sistem

Jadwal akhir yang dihasilkan oleh algoritma genetika merupakan individu dengan nilai fitness tertinggi pada generasi terakhir. Hasil evaluasi menunjukkan bahwa

jadwal tersebut memiliki jumlah konflik dosen dan konflik ruangan sebesar nol atau mendekati nol, serta memenuhi sebagian besar constraint akademik yang ditetapkan.

Seluruh sesi praktikum telah dialokasikan pada ruang laboratorium dengan kapasitas yang sesuai, sementara sesi teori ditempatkan pada ruang kelas dan slot waktu yang diperbolehkan. Selain itu, tidak ditemukan benturan jadwal dosen maupun penggunaan ruangan pada waktu yang sama. Hal ini menunjukkan bahwa algoritma genetika mampu menghasilkan jadwal mata kuliah yang valid dan layak diterapkan dalam lingkungan akademik.

Penerapan sistem penjadwalan mata kuliah berbasis algoritma genetika memberikan dampak positif terhadap operasional akademik. Proses penyusunan jadwal menjadi lebih cepat dan sistematis, serta mengurangi kebutuhan revisi manual akibat konflik jadwal. Selain itu, sistem ini memudahkan pihak akademik dalam melakukan penyesuaian apabila terjadi perubahan data, seperti penambahan mata kuliah atau perubahan jumlah mahasiswa.

IV. KESIMPULAN

Berdasarkan hasil implementasi dan pengujian sistem penjadwalan mata kuliah berbasis algoritma genetika, dapat disimpulkan bahwa algoritma genetika efektif digunakan sebagai metode optimasi dalam menyelesaikan permasalahan penjadwalan. Sistem yang dikembangkan mampu menghasilkan jadwal mata kuliah dengan tingkat konflik dosen dan penggunaan ruangan yang sangat rendah, serta memenuhi sebagian besar constraint akademik yang telah ditetapkan.

Fungsi fitness yang dirancang dengan mempertimbangkan konflik dan pelanggaran constraint mampu merepresentasikan kualitas solusi dan mengarahkan proses evolusi menuju jadwal yang lebih optimal. Melalui mekanisme seleksi, persilangan, mutasi, dan elitisme, kualitas solusi meningkat secara bertahap hingga diperoleh kromosom terbaik sebagai jadwal akhir.

Dengan demikian, penerapan algoritma genetika pada sistem penjadwalan mata kuliah berbasis web dapat menjadi alternatif yang efektif dan efisien dibandingkan metode penjadwalan manual, khususnya dalam meningkatkan konsistensi jadwal, efisiensi pemanfaatan sumber daya akademik, serta mendukung kelancaran proses pembelajaran di perguruan tinggi.

Penelitian ini masih memiliki keterbatasan, yaitu pengujian hanya dilakukan pada satu program studi dan satu periode akademik. Selain itu, parameter algoritma genetika yang digunakan belum dibandingkan dengan metode optimasi lain maupun skenario data yang lebih besar. Hasil *fitness* hingga 1.0 terjadi setelah beberapa hasil jadwal gagal mencapai *fitness* 1.0 pada perbandingan 1:8 sehingga sistem ini masih belum sempurna. Penulis berharap pada penelitian selanjutnya dapat mengembangkan sistem yang lebih baik bahkan multi-program studi serta melakukan optimasi parameter untuk meningkatkan performa sistem.

REFERENSI

- [1] A. B. N. Aji, A. S. Lutfifassa, B. A. Thanta, R. A. Pangestu, and P. Rosyani, "Sistem penjadwalan mata kuliah menggunakan algoritma genetika," *AI dan SPK : Jurnal Artificial Intelligent dan Sistem Penunjang Keputusan*, vol. 2, no. 1, pp. 75–78, Jun. 2024, Accessed: Sep. 26, 2026. [Online]. Available: <https://jurnalmahasiswa.com/index.php/aidanspk/article/view/1496>
- [2] R. Eka Wiratna, A. Lina Nurlaili, and A. Mustika Rizki, "Pembuatan aplikasi penjadwalan mata kuliah menggunakan algoritma genetika," *Jurnal Teknologi dan Manajemen*, vol. 4, no. 1, pp. 13–20, Feb. 2023, doi: 10.31284/J.JTM.2023.V4I1.3990.
- [3] W. Prasetya and D. Jollyta, "Penerapan algoritma genetika dalam penjadwalan mata kuliah," *Jurnal Mahasiswa Aplikasi Teknologi Komputer dan Informasi (JMApTeKsi)*, vol. 5, no. 2, pp. 144–147, Nov. 2023, Accessed: Sep. 26, 2026. [Online]. Available: <https://www.ejournal.pelitaindonesia.ac.id/index.php/jmapteksi/article/view/3931>
- [4] J. M. Al Faiz, A. Firdonsyah, and S. L. Mufreni, "Sistem Penjadwalan Perkuliahan Berbasis Web Menggunakan Algoritma Genetika," *KOMPUTEK*, vol. 9, no. 1, pp. 79–87, Apr. 2025, doi: 10.24269/JKT.V9I1.3260.
- [5] I. H. Sugeha, R. L. Inkiriwang, and P. A. K. Pratisis, "Optimasi penjadwalan menggunakan metode algoritma genetika pada proyek rehabilitasi Puskesmas Minanga," *Jurnal Sipil Statik*, vol. 7, no. 12, pp. 1669–1680, Dec. 2019, Accessed: Sep. 26, 2026. [Online]. Available: <https://ejournal.unsrat.ac.id/v3/index.php/jss/article/view/26145>
- [6] H. A. Sitompul, "Implementasi algoritma genetika untuk menentukan solusi pada optimisasi nonlinier," *Jurnal UDA*, vol. 22, no. 3, pp. 183–188, Jan. 2019, doi: 10.46930/OJSUDA.V22I3.93.
- [7] A. P. Pambudi, A. Waluyo, and E. V. L. N. Fatich, "Perancangan sistem penjadwalan perkuliahan berbasis website menggunakan algoritma genetika," *JATISI*, vol. 8, no. 3, pp. 1133–1146, Sep. 2021, doi: 10.35957/JATISI.V8I3.1051.
- [8] R. F. Syahputra and Y. Yahfizham, "Menganalisis konsep dasar algoritma genetika," *Bhinneka: Jurnal Bintang Pendidikan dan Bahasa*, vol. 2, no. 1, pp. 120–132, Dec. 2024, doi: 10.59024/BHINNEKA.V2I1.643.
- [9] S. Liyan, D. Kriestanto, A. Ramadhan, M. Haries, and L. Lukman, "Lecture scheduling using genetic algorithm method," *Journal of Intelligent Software Systems*, vol. 3, no. 2, pp. 7–12, Dec. 2024, doi: 10.26798/JISS.V3I2.1501.
- [10] M. A. Rohim, F. Wiranto, and D. A. Fauziah, "Optimasi pembuatan jadwal perkuliahan menggunakan algoritma genetika berbasis pendekatan multivariat," *BIOS: Jurnal Teknologi Informasi dan Rekayasa Komputer*, vol. 6, no. 1, pp. 31–38, Mar. 2025, doi: 10.37148/BIOS.V6I1.160.
- [11] S. Katoch, S. S. Chauhan, and V. Kumar, "A review on genetic algorithm: Past, present, and future," *Multimedia Tools and Applications 2020 80:5*, vol. 80, no. 5, pp. 8091–8126, Oct. 2020, doi: 10.1007/S11042-020-10139-6.
- [12] Y. S. Nugraha, U. Darusalam, and A. Iskandar, "Implementasi algoritma genetika pada perancangan aplikasi penjadwalan instalasi antivirus berbasis website menggunakan metode waterfall," *Jurnal JTIK (Jurnal Teknologi Informasi dan Komunikasi)*, vol. 6, no. 1, pp. 125–137, Jan. 2022, doi: 10.35870/JTIK.V6I1.417.
- [13] A. N. Rohmad and M. Akbar, "Penerapan algoritma genetika dalam pengelompokan mahasiswa KKN (Studi KKN: Kkn angkatan XLII Universitas Mercu BuanaY)," *JIKO (Jurnal Informatika dan Komputer)*, vol. 8, no. 1, pp. 50–61, Feb. 2024, doi: 10.26798/JIKO.V8I1.1073.
- [14] W. Tanujaya, D. R. S. Dewi, and D. Endah, "Penerapan algoritma genetik untuk penyelesaian masalah vehicle routing di PT. Mif," *Widya Teknik*, vol. 10, no. 1, pp. 92–102, 2011, doi: 10.33508/WT.V10I1.163.