JAMBURA JOURNAL OF INFORMATICS
Volume 8, No. 2 | October 2026 : 160-168
ISSN 2685-4244, doi: 10.37905/jji.v812.41852

Implementasi algoritma genetika dengan *tournament selection* untuk optimasi penjadwalan pembelajaran

Akhmad Pahdi, Syihabuddin

Program Studi Teknik Informatika, STMIK Banjarbaru, Banjarbaru, Indonesia

Riwayat Artikel:
Diterima 3 Oktober 2026
Direvisi 13 Oktober 2026
Disetujui 27 Oktober 2026

Kata Kunci:
Penjadwalan
Algoritma Genetika
Tournament Selection
Roulette Wheel
Python

ABSTRACT. *Subject scheduling at MAN 4 Banjar is still performed manually using Microsoft Excel. Based on interviews, the manual scheduling and revision process takes up to one week per semester due to repetitive adjustments to avoid conflicts among teachers, classes, and subjects. This study aims to design a Python-based scheduling application using a genetic algorithm and to compare the performance of Tournament Selection and Roulette Wheel methods. The evaluation was conducted through 30 experimental runs and analyzed using the Independent Sample T-Test. The results demonstrate that the system automates the scheduling process, which previously took one week manually, to an average of 108.132–132.127 seconds per scheduling run across 30 experiments, and eliminates teacher schedule conflicts. Tournament Selection exhibited a higher average fitness value (0.004225) and fewer unfulfilled assignments (3.27) than Roulette Wheel (fitness 0.003634; unfulfilled assignments 3.33), although the difference was not statistically significant (p = 0.31). Both methods produced schedules free of teacher conflicts. These findings prove that the developed system is reliable for automating and improving the school's operational efficiency.*

ABSTRAK. Penjadwalan mata pelajaran di MAN 4 Banjar masih dilakukan secara manual menggunakan Microsoft Excel. Berdasarkan wawancara, proses penyusunan dan revisi jadwal manual memakan waktu hingga satu minggu per semester karena penyesuaian berulang untuk menghindari konflik antarguru, kelas, dan mata pelajaran. Penelitian ini bertujuan merancang aplikasi penjadwalan berbasis Python menggunakan algoritma genetika, serta membandingkan kinerja metode seleksi Tournament Selection dan Roulette Wheel. Pengujian dilakukan sebanyak 30 kali percobaan dan dianalisis menggunakan Independent Sample T-Test. Hasil penelitian menunjukkan bahwa sistem mampu mengotomatisasi proses penjadwalan yang sebelumnya memakan waktu satu minggu (manual) menjadi rata-rata 108,132–132,127 detik per proses penjadwalan pada 30 percobaan, serta mengeliminasi konflik jadwal guru. Tournament Selection menunjukkan rata-rata nilai fitness lebih tinggi (0,004225) dan penugasan belum terpenuhi lebih sedikit (3,27) dibandingkan Roulette Wheel (fitness 0,003634; penugasan tersisa 3,33), meskipun perbedaan tersebut tidak signifikan secara statistik (p = 0,31). Kedua metode menghasilkan jadwal bebas konflik guru. Hasil ini membuktikan bahwa sistem yang dikembangkan dapat diandalkan untuk mengotomatisasi dan meningkatkan efisiensi operasional sekolah.

This is an open-access article under the CC–BY-SA license.

Penulis Korespondensi:
Akhmad Pahdi
Prodi Teknik Informatika, STMIK Banjarbaru
Jl. A. Yani Km.33,5 Loktabat, Banjarbaru, Kalimantan Selatan, Indonesia
Email: ahmadpahdi@gmail.com

PENDAHULUAN

Proses penjadwalan kegiatan belajar di sekolah merupakan salah satu kegiatan penting yang menentukan kelancaran proses pembelajaran (Irfan dkk., 2022). Penjadwalan yang tidak tepat dapat menyebabkan konflik jadwal mengajar guru, ketidakseimbangan beban mengajar, dan gangguan

terhadap efektivitas pembelajaran (Suwirmayanti dkk., 2016). Di MAN 4 Banjar, yang melayani lebih dari 1.000 siswa dengan 30+ guru, proses penjadwalan masih dilakukan secara manual menggunakan Microsoft Excel dan aplikasi *Timetable*. Berdasarkan wawancara dengan Wakil Kepala Kurikulum, proses ini memakan waktu hingga satu minggu per semester karena harus melalui revisi berulang untuk menghindari konflik jadwal. Praktik manual ini rentan terhadap kesalahan manusia seperti duplikasi jadwal dan ketidakseimbangan beban kerja guru (Fajrianto dkk., 2021; Siagian dkk., 2024). Akibatnya, Wakil Kepala Kurikulum kehilangan waktu untuk perencanaan pembelajaran yang efektif, dan kualitas pengajaran menurun akibat beban administratif yang berlebihan (Azuani dkk., 2019).

Untuk mengatasi permasalahan tersebut, penelitian ini menerapkan algoritma genetika, yaitu teknik pencarian dan optimisasi metaheuristik yang diadaptasi dari mekanisme seleksi alam dan genetika biologis (Goldberg, 2016; Sivanandam & Deepa, 2021). Pendekatan metaheuristik dipilih karena kemampuannya dalam mengeksplorasi ruang solusi secara global pada permasalahan optimisasi kombinatorial yang kompleks seperti penjadwalan sekolah (Talbi, 2020; Michalewicz, 2017). Sistem yang dikembangkan tidak hanya menghasilkan jadwal secara otomatis, tetapi juga menyediakan akses *real-time* bagi Wakil Kepala Kurikulum untuk memantau dan mengevaluasi jadwal secara efisien.

Penelitian ini menerapkan metode *Tournament Selection* yang dikombinasikan dengan *Order Crossover* dan *Swap Mutation* dalam konteks penjadwalan madrasah dengan kompleksitas batasan yang tinggi, serta membandingkan kinerjanya dengan *Roulette Wheel Selection* pada dataset nyata MAN 4 Banjar. Tujuan penelitian ini adalah: (1) Mengukur waktu eksekusi sistem dalam menghasilkan jadwal; (2) Membandingkan nilai fitness dan penugasan tersisa antara metode Tournament Selection dan Roulette Wheel; serta (3) Memvalidasi pemenuhan batasan (hard constraints) pada jadwal yang dihasilkan.

Kebaruan (*novelty*) penelitian ini terletak pada penerapan algoritma genetika dalam konteks penjadwalan madrasah (MAN 4 Banjar) yang memiliki batasan unik, seperti minimal 24 JP per minggu dan maksimal 8 JP per hari. Berbeda dengan penelitian Panjaitan (2022) yang berfokus pada penjadwalan SMK berbasis web, penelitian ini secara spesifik membandingkan performa *Tournament Selection* dan *Roulette Wheel* pada dataset nyata madrasah dengan kompleksitas batasan yang tinggi. Selain itu, penelitian ini juga menyajikan bukti empiris bahwa sistem mampu mengeliminasi proses revisi manual yang sebelumnya menjadi *bottleneck* utama dalam penjadwalan.

METODE

Penelitian ini menggunakan jenis penelitian eksperimen yang bertujuan menguji kinerja aplikasi penjadwalan yang dikembangkan. Penelitian dilaksanakan mulai 4 September 2025 di MAN 4 Banjar, Jl. Pendidikan No. 01, Sungai Paring, Martapura, Kabupaten Banjar, Kalimantan Selatan. Data yang digunakan adalah jadwal pelajaran kelas XI semester ganjil tahun pelajaran 2025/2026

Analisis Data

Data yang digunakan dalam penelitian ini terdiri dari data guru, data kelas, data mata pelajaran, dan data pengajar. Pengumpulan data dilakukan melalui wawancara dengan Wakil Kepala Kurikulum dan Staff Kurikulum MAN 4 Banjar, studi pustaka, dan dokumentasi resmi sekolah. Data yang berhasil dikumpulkan meliputi 72 data guru, 27 data kelas, 40 data mata pelajaran, dan 521 data pengajar yang merepresentasikan seluruh kombinasi penugasan guru. Parameter penjadwalan yang digunakan mengikuti ketentuan MAN 4 Banjar, yaitu setiap guru harus memiliki minimal 24 jam pelajaran (jp) per minggu dan maksimal 8 jp per hari, dengan tidak adanya mata pelajaran yang bertabrakan atau bentrok antarkelas.

Dalam perancangan sistem penjadwalan menggunakan algoritma genetika, tahap awal yang dilakukan adalah menentukan populasi awal, menentukan batasan penjadwalan, lalu merepresentasikan nilai kromosom, kemudian menginisialisasi populasi awal, melakukan seleksi, crossover (kawin silang), mutasi, lalu sampai ke kondisi selesai.

Populasi Awal

Sebelum pembentukan populasi awal, terlebih dahulu dirancang struktur kromosom yang merepresentasikan komponen penjadwalan, meliputi kode kelas (K), kode guru (G), kode mata pelajaran (M), dan kode jam pelajaran (JP). Struktur kromosom disusun dalam bentuk (K, G, M, J), dengan panjang kromosom ditentukan berdasarkan jumlah keseluruhan jam pelajaran yang harus dijadwalkan. Setiap gen merepresentasikan satu sesi pembelajaran yang memuat informasi mengenai kelas, guru pengajar, mata pelajaran, serta slot waktu pelaksanaan pembelajaran.

Fungsi Fitness

Pada tahap ini dilakukan evaluasi untuk menentukan kromosom terbaik melalui fungsi *fitness*. Nilai *fitness* dirumuskan secara matematis dengan memperhitungkan total penalti dari berbagai jenis pelanggaran batasan keras (*hard constraints*) dan batasan lunak (*soft constraints*), dengan formulasi sebagai berikut:

$$Fitness = \frac{1}{(1+total\ pelanggaran)} \quad (1)$$

Nilai fitness yang dihasilkan berkisar antara 0 hingga 1. Nilai 1 menunjukkan kondisi optimal, yaitu ketika kromosom tidak memiliki pelanggaran sama sekali. Sebaliknya, nilai yang mendekati 0 menunjukkan jadwal yang sangat buruk karena banyak melanggar batasan. Dengan demikian, semakin tinggi nilai fitness, semakin dekat solusi tersebut dengan kondisi optimal.

Seleksi

Untuk membentuk populasi generasi baru, sistem menggunakan metode *Tournament Selection*. Sebagai ilustrasi sederhana, misalkan dalam suatu populasi kecil terdapat 4 kromosom dengan ukuran turnamen = 2. Metode ini memilih kromosom terbaik dari sampel acak kecil (2 kromosom) yang bersaing dalam setiap "turnamen". Kromosom pemenang (fitness tertinggi) maju ke populasi orang tua.

Cara kerja metode ini yaitu dari 4 kromosom populasi tersebut, diambil secara acak 2 kromosom untuk bertanding, pemenang (kromosom dengan fitness tertinggi) maju ke populasi orang tua, dan proses ini diulang sebanyak 4 turnamen untuk memilih 4 orang tua baru. Hasil pemilihan dari metode tournament sebagai berikut:

- Orang Tua 1 (K2):
  K01G02M02J01 K01G01M01J02 K01G03M03J05 K01G04M04J03 K01G02M02J06 K01G01M01J04
- Orang Tua 2 (K1):
  K01G01M01J01 K01G03M03J03 K01G04M04J02 K01G02M02J05 K01G01M01J06 K01G02M02J04
- Orang Tua 3 (K1):
  K01G01M01J01 K01G03M03J03 K01G04M04J02 K01G02M02J05 K01G01M01J06 K01G02M02J04
- Orang Tua 4 (K2):
  K01G02M02J01 K01G01M01J02 K01G03M03J05 K01G04M04J03 K01G02M02J06 K01G01M01J04

Kawin Silang (Crossover)

Kawin silang digunakan sebagai metode pemotongan secarta acak. Kawin silang akan dilakukan jika ada nilai random yang di bangkitkan oleh suatu kromosom kurang dari probabilitas yang ditentukan, yaitu misal ada suatu bilangan random [0,2; 0,8; 0,5; 0,95], dan di tentukan nilai probabilitas pada umumnya yaitu 0,5 (nilai yang mendekati nilai 1).

Bilangan acak dibangkitkan untuk menentukan suatu titik potong adalah [1–N] dengan N yaitu banyak jumlah gen dalam suatu kromosom. Contoh nilai bilangan random yang di bangkitkan dari kawin silang adalah kromosom 1 dan 3 karena memiliki nilai kurang dari probabilitas yang ditetapkan, yaitu kromosom 1 nilai 0,2 dan kromosom 3 nilai 0,5. Untuk titik potongnya ada di posisi gen ke-2, maka proses kawin silangnya:

Kromosom1:	K01G02M02J01	K01G01M01J02	K01G03M03J05	K01G04M04J03
	K01G02M02J06	K01G01M01J04		
Kromosom 3:	K01G01M01J01	K01G03M03J03	K01G04M04J02	K01G02M02J05
	K01G01M01J06	K01G02M02J04		

Hasil kawin silang:

Kromosom 1:	K01G02M02J01	K01G03M03J03	K01G04M04J02	K01G02M02J05
	K01G01M01J06	K01G02M02J04		
Kromosom 3:	K01G01M01J01	K01G01M01J02	K01G03M03J05	K01G04M04J03
	K01G02M02J06	K01G01M01J04		

Fitness kromosom 1 setelah pindah silang = $\frac{1}{1+(0+0+0+1+0+0)} = 0{,}5$

Fitness kromosom 3 setelah pindah silang = $\frac{1}{1+(0+0+0+1+0+0)} = 0{,}5$

Mutasi

Setelah proses crossover selesai, langkah berikutnya adalah mutasi, yang merupakan mekanisme penting dalam algoritma genetika untuk memperkenalkan variasi genetik acak ke dalam kromosom keturunan, sehingga mencegah konvergensi prematur dan menjelajahi ruang pencarian yang lebih luas untuk solusi. Jika angka acak lebih kecil dari probabilitas mutasi, gen tertentu dimodifikasi melalui rekoding untuk hasil kebugaran yang lebih optimal. Untuk menentukan gen yang dimutasi maka di hitung jumlah total gen dalam suatu populasi:

Jumlah Gen = jumlah gen per kromosom × jumlah kromosom dalam populasi. Berdasarkan data ini, jumlah total gen adalah 6 × 4 = 24 gen. Probabilitas mutasi sebesar 0,2 menghasilkan perkiraan mutasi sebesar 0,2 × 24 = 2,4 ≈ 4 gen yang akan mengalami mutasi.

Proses iterasi dilakukan sebanyak 24 kali (indeks 0-23) dengan membangkitkan bilangan acak [0-1] untuk masing-masing posisi; diasumsikan gen ke-2 dan gen ke-3 pada Kromosom 3 memperoleh nilai di bawah probabilitas mutasi, sehingga slot waktu mata pelajaran (J) diubah.

Kondisi Selesai

Kondisi selesai dapat menghentikan proses algortima genetika ini jika jumlah gen dan iterasi telah mencapai target.

Penerapan Algoritma Genetika

Algoritma genetika adalah metode optimisasi metaheuristik yang terinspirasi oleh evolusi biologis, mampu menangani batasan kompleks seperti konflik guru, ruang, dan waktu melalui evolusi paralel

populasi jadwal kandidat (Amal dan Fadhil, 2020). Proses algoritma genetika terdiri dari siklus: inisialisasi populasi, evaluasi fitness, seleksi, crossover, mutasi, penggantian generasi, dan terminasi (Fitriyani dkk, 2024).

Representasi kromosom pada penelitian ini berupa kombinasi kelas-guru-mata pelajaran-jam (K, G, M, J). Fungsi fitness berbasis penalti menghitung nilai kebugaran kromosom berdasarkan beberapa komponen, yaitu kekurangan slot (bobot × 10), pelanggaran booking guru (bobot × 500), blok jam terpecah (bobot × 50), dan blok jam kecil (bobot × 80), sebagaimana ditunjukkan pada persamaan (1).

Dua metode seleksi dibandingkan dalam penelitian ini. *Tournament Selection* memilih individu dengan fitness terbaik dari sejumlah kandidat yang dipilih secara acak sebanyak ukuran turnamen yang ditentukan. *Roulette Wheel Selection* memilih individu berdasarkan probabilitas yang proporsional terhadap nilai fitness, sehingga individu dengan fitness lebih tinggi memiliki peluang lebih besar untuk terpilih. Operator genetika yang digunakan adalah *Order Crossover (OX)* untuk persilangan kromosom dan *Swap Mutation* untuk mutasi, dengan mekanisme elitisme yang mempertahankan kromosom terbaik pada setiap generasi.

Perancangan Sistem

Sistem dikembangkan berbasis aplikasi desktop menggunakan Python dengan dua jenis pengguna, yaitu admin dan guru. Admin memiliki hak akses untuk mengelola data guru, kelas, mata pelajaran, dan pengajar, serta menjalankan proses penjadwalan. Guru memiliki hak akses untuk melihat hasil laporan jadwal yang dihasilkan oleh sistem.

Perancangan sistem menggunakan *Unified Modeling Language* yang meliputi diagram konteks, use case diagram, sequence diagram, activity diagram, dan relasi tabel. Sistem dikembangkan menggunakan Python 3.11.5 dengan antarmuka berbasis HTML, CSS, dan JavaScript native yang dirender melalui PyWebView, serta MySQL sebagai sistem manajemen basis data (Destriana dkk., 2022).

Analisis Proses Penjadwalan Manual

Untuk mendapatkan data pembanding yang valid, dilakukan wawancara semi-terstruktur dengan Wakil Kepala Kurikulum MAN 4 Banjar. Berdasarkan hasil wawancara, proses penjadwalan manual saat ini menggunakan kombinasi Microsoft Excel dan aplikasi *Timetable*. Responden menyatakan bahwa penyusunan jadwal memakan waktu hingga 7 hari (satu minggu) per semester.

Waktu tersebut tidak hanya digunakan untuk menyusun jadwal awal, tetapi juga untuk melakukan revisi berulang. Responden mengonfirmasi bahwa terdapat kendala utama: jika ditemukan kesalahan atau konflik jadwal setelah jadwal selesai disusun, maka proses revisi akan memakan waktu tambahan yang signifikan. Proses iteratif manual inilah yang menjadi akar permasalahan efisiensi di MAN 4 Banjar. Oleh karena itu, pengukuran waktu 7 hari ini dijadikan sebagai *baseline* (titik acuan) untuk mengevaluasi efisiensi sistem algoritma genetika yang dikembangkan.

Teknik Pengujian

Pengujian dilakukan melalui dua tahap utama. Pertama, *White Box Testing* menggunakan metode *Basis Path Testing* untuk memverifikasi struktur logika internal program pada tujuh fungsi inti algoritma genetika. Metode ini dipilih karena algoritma genetika memiliki banyak percabangan kondisional yang saling berkaitan, sehingga perlu dipastikan tidak ada jalur logika yang salah atau tidak pernah dieksekusi.

Kedua, uji komparasi kinerja dilakukan untuk membandingkan efektivitas metode Tournament Selection dan Roulette Wheel Selection. Pengujian dijalankan menggunakan parameter algoritma genetika sebagai berikut: ukuran populasi 50, jumlah generasi 35, ukuran turnamen 4, laju crossover 0,9, dan laju mutasi 0,2. Pemilihan parameter ini didasarkan pada pertimbangan berikut:

- Ukuran Populasi (50): Ukuran populasi 50 dipilih karena berada dalam rentang standar yang direkomendasikan dalam literatur untuk menjaga keragaman genetik yang memadai tanpa membebani waktu komputasi secara berlebihan. Populasi yang terlalu kecil berisiko mengalami konvergensi prematur, sedangkan populasi yang terlalu besar akan memperlambat proses evaluasi fitness. (Sivanandam & Deepa, 2021; Goldberg, 2016).
- Jumlah Generasi (35): Jumlah generasi 35 ditetapkan sebagai kompromi antara kebutuhan eksplorasi ruang solusi dan batasan waktu eksekusi. Berdasarkan pengujian pendahuluan, konvergensi solusi cenderung stabil pada kisaran generasi tersebut untuk dataset yang digunakan. (Talbi, 2020).
- Ukuran Turnamen (4): Ukuran turnamen 4 dipilih karena memberikan tekanan seleksi yang cukup kuat untuk menyaring kromosom berkualitas tanpa menyebabkan hilangnya keragaman populasi secara drastis. Literatur menunjukkan bahwa ukuran turnamen yang lebih besar dari nilai default (2) dapat mempercepat identifikasi solusi yang lebih baik. (Talbi, 2020; Sivanandam & Deepa, 2021).
- Laju Crossover (0,9) dan Mutasi (0,2): Laju crossover yang tinggi (0,9) bertujuan untuk mempercepat penyebaran sifat unggul antar kromosom, sementara laju mutasi 0,2 (sedikit di atas standar umum) dipilih untuk menjaga keragaman genetik dan mencegah terjebak pada *local optima*. (Goldberg, 2016; Michalewicz, 2017).

Mengingat sifat algoritma genetika yang stokastik (berbasis probabilitas acak), setiap metode dijalankan sebanyak 30 kali percobaan dengan variasi seed (dimulai dari seed dasar 42 hingga 71) untuk memastikan objektivitas dan konsistensi hasil. Data hasil dari 30 percobaan tersebut kemudian dianalisis menggunakan statistik deskriptif (rata-rata dan standar deviasi) dan diuji menggunakan *Independent Sample T-Test* dengan tingkat signifikansi $\alpha = 0{,}05$. Parameter yang dibandingkan meliputi waktu eksekusi, nilai fitness, jumlah slot terjadwal, jumlah penugasan yang belum terpenuhi, dan jumlah bentrok jadwal.

HASIL DAN DISKUSI

Implementasi Sistem Berjalan

Sistem memiliki beberapa halaman utama, yaitu halaman login, dashboard yang menampilkan ringkasan data secara real-time (72 guru, 27 kelas, 40 mata pelajaran, 521 pengajar), halaman data master untuk pengelolaan data referensi, halaman buat jadwal untuk mengonfigurasi parameter algoritma genetika dan menjalankan proses penjadwalan, halaman perbandingan metode, serta halaman laporan jadwal yang menampilkan hasil dalam format tabel per kelas dengan fitur ekspor Excel.

Hasil Pengujian WhiteBox

Pengujian Basis Path Testing dilakukan pada tujuh fungsi inti algoritma genetika dalam program, yaitu fungsi perhitungan fitness (_fitness), seleksi tournament (_tournament_select), seleksi roulette wheel (_roulette_select), order crossover (_order_crossover), mutasi (_mutate), pengendali alur utama GA (_solve_schedule_genetic), dan deteksi duplikasi slot jadwal (detect_slot_duplicates). Seluruh jalur eksekusi yang teridentifikasi pada ketujuh fungsi tersebut berhasil diuji tanpa ditemukan kesalahan logika.

Hasil pengujian menunjukkan bahwa fungsi fitness menghitung penalti secara akurat pada setiap kondisi, termasuk kondisi ideal tanpa pelanggaran, fungsi seleksi menangani kondisi batas dengan benar (termasuk pencegahan pembagian dengan nol pada roulette wheel), fungsi crossover menghasilkan kromosom anak berupa permutasi valid tanpa duplikasi, dan fungsi pengendali alur

utama GA terbukti mengarahkan eksekusi ke metode seleksi yang sesuai serta mempertahankan enam kromosom elit pada setiap generasi.

Perbandingan Kinerja Tournament Selection dan Roulette Wheel

Uji komparasi dilakukan dengan menjalankan setiap metode seleksi sebanyak 30 kali percobaan menggunakan variasi seed (dimulai dari seed dasar 42) untuk memastikan objektivitas dan mengakomodasi sifat stokastik algoritma genetika. Hasil rekapitulasi rata-rata kinerja dari 30 percobaan tersebut ditunjukkan pada Tabel 1.

Tabel 1. Rekapitulasi hasil pengujian 30 percobaan

Parameter	Tournament Selection	Roulette Wheel	p-value (T-Test)
Rata-rata Nilai Fitness	0,004225	0,003634	0,31
Standar Deviasi Fitness	0,001788	0,002579	
Rata-rata Waktu Eksekusi	132,127 detik	108,132 detik	
Rata-rata Penugasan Tersisa	3,27	3,33	
Fitness Terbaik (Maksimum)	0,007843 (Seed 71)	0,012821 (Seed 57)	

Berdasarkan Tabel 1, Tournament Selection menunjukkan nilai rata-rata fitness yang lebih tinggi (0,004225) dibandingkan Roulette Wheel (0,003634). Meskipun demikian, hasil uji *Independent Sample T-Test* menunjukkan nilai p-value sebesar 0,31 ($p > 0,05$), yang berarti perbedaan kinerja antara kedua metode tidak terbukti secara statistik. Namun, dari sisi kelengkapan penugasan, Tournament Selection secara konsisten menyisakan lebih sedikit penugasan yang belum terpenuhi (3,27) dibandingkan Roulette Wheel (3,33).

Dari segi waktu eksekusi, Roulette Wheel menunjukkan kinerja numerik yang lebih cepat dengan rata-rata 108,132 detik dibandingkan Tournament Selection 132,127 detik. Meskipun demikian, selisih waktu ini masih dapat ditoleransi karena kedua metode sama-sama menghasilkan jadwal tanpa bentrok (0 konflik) pada seluruh percobaan, membuktikan bahwa sistem telah memenuhi batasan keras (*hard constraints*) penjadwalan. Selisih nilai rata-rata fitness yang lebih tinggi pada Tournament Selection terjadi karena mekanisme pemilihan yang lebih terarah: dengan ukuran turnamen 4, sistem membandingkan 4 solusi secara acak dan memilih yang terbaik, sehingga solusi berkualitas tinggi lebih sering terpilih untuk diproses pada generasi berikutnya. Rekapitulasi parameter operasional penjadwalan antara kedua metode ditunjukkan pada Tabel 2.

Tabel 2. Rekapitulasi parameter operasional penjadwalan (30 percobaan)

Parameter	Tournament Selection	Roulette Wheel
Rata-rata Slot Terjadwal	303,03	303,60
Rata-rata Penugasan Belum Terpenuhi	3,27	3,33
Jumlah Bentrok Jadwal	0	0

Berdasarkan Tabel 2, kedua metode berhasil menghasilkan jadwal tanpa bentrok (0 konflik) pada seluruh 30 percobaan. Dari sisi kelengkapan penugasan, Tournament Selection menyisakan lebih sedikit penugasan yang belum terpenuhi (3,27) dibandingkan Roulette Wheel (3,33). Meskipun rata-rata slot terjadwal pada Roulette Wheel sedikit lebih tinggi (303,60 berbanding 303,03), namun Tournament Selection secara konsisten menunjukkan nilai fitness yang lebih tinggi dan lebih andal dalam memastikan seluruh slot mata pelajaran terisi secara optimal sesuai dengan batasan yang ditetapkan.

Perbandingan Penjadwalan Manual dan Sistem Otomatis

Berdasarkan hasil wawancara dengan pihak kurikulum, jadwal manual final yang dihasilkan sebenarnya sudah bebas dari konflik. Namun, untuk mencapai kondisi bebas konflik tersebut, prosesnya memerlukan waktu hingga 7 hari karena harus melalui revisi berulang secara manual setiap kali ditemukan bentrok jadwal antar guru atau kelas.

Sebaliknya, pada jadwal yang dihasilkan sistem secara otomatis, tidak ditemukan konflik guru (0 bentrok) pada seluruh slot waktu yang dianalisis. Berdasarkan 30 percobaan, sistem membutuhkan rata-rata 132,127 detik untuk Tournament Selection dan 108,132 detik untuk Roulette Wheel per proses penjadwalan hingga generasi ke-35. Waktu tersebut hanya mencakup proses inti algoritma genetika, tidak termasuk input data dan ekspor laporan. Perbandingan ini menunjukkan bahwa sistem tidak hanya mempercepat proses komputasi, tetapi juga mengurangi kebutuhan terhadap revisi manual berulang yang sebelumnya menjadi salah satu kendala dalam proses penjadwalan.

Pembahasan

Hasil penelitian menunjukkan bahwa penerapan algoritma genetika dengan representasi kromosom (K, G, M, J), fungsi fitness berbasis penalti, operator Order Crossover, dan Swap Mutation dapat menghasilkan jadwal pembelajaran yang optimal dan bebas konflik. Hasil pengujian fungsional melalui White Box Testing menunjukkan bahwa seluruh fungsi inti berjalan sesuai dengan rancangan tanpa kesalahan logika, sehingga sistem dapat diandalkan untuk menghasilkan jadwal yang konsisten.

Dari segi waktu eksekusi, sistem menyelesaikan satu proses penjadwalan dalam rata-rata 108,132–132,127 detik pada 30 percobaan, jauh lebih singkat dibandingkan baseline proses manual yang memerlukan waktu hingga satu minggu berdasarkan wawancara dengan Wakil Kepala Kurikulum. Perbandingan ini bersifat perbandingan antara proses komputasi inti algoritma genetika dengan proses manual end-to-end, sehingga klaim efisiensi perlu dipahami dalam batas cakupan tersebut.

Jika dibandingkan dengan penelitian terdahulu, Panjaitan (2022) membandingkan metode peringkat dan roda roulette pada penjadwalan SMK berbasis web, sedangkan penelitian ini berfokus pada penerapan Tournament Selection di lingkungan madrasah dengan kompleksitas batasan yang lebih tinggi, seperti alokasi jam mengajar guru dan ketersediaan ruang. Hasil penelitian ini sejalan dengan temuan Amal (2020) yang menunjukkan bahwa algoritma genetika mampu menghasilkan jadwal optimal dalam waktu singkat dibandingkan metode manual.

Kontribusi penelitian ini terletak pada evaluasi empiris perbandingan *Tournament Selection* dan *Roulette Wheel* pada penjadwalan madrasah. Hasil pengujian menunjukkan adanya perbedaan karakteristik kinerja pada nilai fitness, penugasan yang belum terpenuhi, dan waktu eksekusi. *Tournament Selection* menunjukkan rata-rata fitness yang lebih tinggi dan penugasan belum terpenuhi yang sedikit lebih rendah, sedangkan *Roulette Wheel* memiliki waktu eksekusi yang lebih cepat. Namun, perbedaan kinerja kedua metode tidak terbukti signifikan secara statistik ($p = 0,31$).

KESIMPULAN

Berdasarkan hasil perancangan dan implementasi, aplikasi penjadwalan mata pelajaran di MAN 4 Banjar berhasil dibangun menggunakan Python 3.11.5 dengan antarmuka PyWebView dan basis data MySQL. Sistem mampu mengelola data 72 guru, 27 kelas, 40 mata pelajaran, dan 521 pengajar secara terintegrasi. Algoritma genetika berhasil diterapkan dengan representasi kromosom (K, G, M, J), fungsi fitness berbasis penalti, operator Order Crossover dan Swap Mutation, serta mekanisme elitisme. Hasil pengujian komparatif (30 percobaan) dan uji statistik T-Test (p-value > 0,05) menunjukkan bahwa Tournament Selection memberikan performa numerik yang lebih baik dibandingkan Roulette Wheel Selection, dengan nilai fitness rata-rata 0,004225 berbanding

0,003634, serta penugasan yang lebih lengkap (rata-rata penugasan tersisa 3,27 berbanding 3,33). Kedua metode juga berhasil menghasilkan jadwal yang bebas bentrok (0 konflik). Hasil White Box Testing membuktikan seluruh fungsi inti algoritma genetika berjalan tanpa kesalahan logika. Sistem terbukti mampu mengotomatisasi proses penjadwalan yang sebelumnya dilakukan secara manual dengan memakan waktu hingga satu minggu per semester. Dengan adanya sistem ini, proses penjadwalan dapat diselesaikan dalam rata-rata 108,132–132,127 detik per proses penjadwalan pada 30 percobaan dan langsung menghasilkan jadwal yang bebas dari konflik guru (0 bentrok) tanpa memerlukan revisi manual. Pengembangan selanjutnya dapat dilakukan dengan menambahkan batasan lunak (*soft constraints*) tambahan pada fungsi fitness, seperti preferensi jam mengajar guru dan pemerataan beban mengajar, serta memperluas cakupan data hingga mencakup seluruh tingkat kelas di MAN 4 Banjar.

Ucapan Terima Kasih

Penulis mengucapkan terima kasih kepada Sekolah Tinggi Manajemen Informatika (STMIK) Banjarbaru atas dukungan yang diberikan dalam pelaksanaan penelitian ini. Ucapan terima kasih kepada rekan-rekan dosen yang telah memberikan bantuan dan masukan, serta kepada pihak MAN 4 Banjar yang telah memberikan kesempatan, akses data, dan dukungan selama penelitian.

Referensi

Amal, M. F. A. (2020). *Implementasi algoritma genetika untuk otomatisasi penjadwalan sekolah menengah pertama* [Skripsi, Universitas Islam Negeri Maulana Malik Ibrahim]. Etheses UIN Malang. http://etheses.uin-malang.ac.id/24007/

Azuani, A., Rahman, M. M., & Abdullah, N. A. (2019). Automated student timetable scheduling system based on genetic algorithm. *INTI Journal, 2019*(69), 1–5.

Destriana, R., Husain, S. M., Handayani, N., & Siswanto, A. T. P. (2022). *Diagram UML dalam membuat aplikasi Android Firebase: Studi kasus aplikasi Bank Sampah*. Deepublish.

Fajrianto, A., Ilhamsyah, I., & Hidayati, R. (2021). Sistem penjadwalan mata pelajaran sekolah dasar berbasis web. *Jurnal Khatulistiwa Informatika, 10*(1), 32–38.

Fitriyani, N., Rahmawati, D., & Siregar, H. (2024). Analisis kinerja algoritma genetika untuk optimasi penjadwalan mata pelajaran. *Jurnal Sistem Informasi dan Komputer, 12*(1), 25–34.

Goldberg, D. E. (2016). *Genetic algorithms in search, optimization, and machine learning*. Addison-Wesley.

Irfan, M., Sari, D. P., & Siregar, R. (2022). Implementasi algoritma genetika dalam penjadwalan mata pelajaran. *JOMLAI: Journal of Machine Learning and Artificial Intelligence, 1*(2), 10–18.

Michalewicz, Z. (2017). *Genetic algorithms + data structures = evolution programs* (3rd ed.). Springer-Verlag.

Panjaitan, D. J. (2022). *Sistem informasi penjadwalan mata pelajaran berbasis web menggunakan algoritma genetika (Studi kasus: SMK Negeri 3 Medan)* [Skripsi, Universitas Medan Area]. Repositori UMA. https://repositori.uma.ac.id/handle/123456789/16691

Siagian, Q. A. A., Hasibuan, M. S., & Suhardi, S. (2024). Sistem penjadwalan mata pelajaran memakai algoritma genetika berbasis web. *INFOMATEK: Jurnal Informatika, 6*(1), 55–64.

Sivanandam, S. N., & Deepa, S. N. (2021). *Introduction to genetic algorithms*. Springer Nature. https://doi.org/10.1007/978-3-540-73190-0

Suwirmayanti, N., Budi, I., & Widyastuti, R. (2016). Sistem informasi penjadwalan mata pelajaran sekolah menengah kejuruan. *Jurnal Informatika, 10*(1), 45–52.

Talbi, E.-G. (2020). *Metaheuristics: From design to implementation*. John Wiley & Sons.