

MENGATASI INEAFISIENSI PENJADWALAN TUGAS: ALGORITMA GENETIKA UNTUK PENINGKATAN MANAJEMEN WAKTU MAHASISWA YANG OPTIMAL

Fahmantha Ardhi Karim¹⁾, Jecky Evangelis Aritonang²⁾, Friska Br Tarigan³⁾, Libertino Felani Xavier Sarmiento⁴⁾, Lukas Salossa⁵⁾, Frengky Sangkek⁶⁾

^{1,2,3,4,5,6)} Nusa Putra University

Jl. Raya Cibolang Cisaat - Sukabumi No.21, Cibolang Kaler, Kec. Cisaat, Kabupaten Sukabumi, Jawa Barat 43152

fahmantha.ardhi_ti23@nusaputra.ac.id¹⁾, jecky.evangelis_ti23@nusaputra.ac.id²⁾, Friska.br_ti23@nusaputra.ac.id³⁾, libertino.sarmiento_ti23@nusaputra.ac.id⁴⁾, lukas.salossa_ti23@nusaputra.ac.id⁵⁾, frengki.sangkek_ti23@nusaputra.ac.id⁶⁾

ABSTRAK

Manajemen waktu akademik merupakan tantangan krusial bagi mahasiswa teknik informatika yang menghadapi beban tugas multitasking dengan variasi domain kognitif yang kompleks. Penjadwalan manual sering kali mengabaikan prioritas deadline dan dampak kelelahan akibat pergantian fokus (switching cost), yang berujung pada penurunan produktivitas. Penelitian ini mengusulkan sistem optimasi penjadwalan tugas menggunakan Algoritma Genetika (AG) dengan fungsi tujuan multi-objective yang meminimalkan total keterlambatan dan biaya peralihan kognitif. Model dikembangkan menggunakan parameter bobot seimbang dan diuji menggunakan dataset riil dari 31 responden dengan dominasi tugas Logika dan Pemrograman sebesar 60.2%. Hasil eksperimen menunjukkan bahwa algoritma mampu mencapai konvergensi solusi optimal pada generasi ke-25. Pada studi kasus validasi, sistem berhasil menurunkan nilai kerugian total dari 6.8 menjadi 4.5, atau setara dengan peningkatan efisiensi sebesar 33.8%. Penelitian ini membuktikan bahwa pendekatan komputasi evolusioner efektif dalam menghasilkan jadwal yang tepat waktu serta meminimalkan beban mental mahasiswa.

Kata Kunci: Algoritma Genetika, Optimasi Penjadwalan, Biaya Peralihan Kognitif, Manajemen Waktu, Keterlambatan

ABSTRACT

Academic time management is a crucial challenge for Informatics Engineering students facing multitasking workloads with complex cognitive domain variations. Manual scheduling often overlooks deadline priorities and the impact of fatigue caused by focus switching (switching cost), resulting in decreased productivity. This study proposes a task scheduling optimization system using Genetic Algorithm (GA) with a multi-objective function that minimizes total tardiness and cognitive switching costs. The model was developed using balanced weight parameters and tested using a real dataset from 31 respondents dominated by Logic and Programming tasks at 60.2%. Experimental results showed that the algorithm achieved optimal solution convergence at the 25th generation. In the validation case study, the system successfully reduced the total cost value from 6.8 to 4.5, equivalent to an efficiency increase of 33.8%. This study demonstrates that the evolutionary computation approach is effective in generating schedules that are timely while minimizing students' mental load.

Keywords: Genetic Algorithm, Scheduling Optimization, Cognitive Switching Cost, Time Management, Tardiness

I. PENDAHULUAN

Mahasiswa teknik informatika dihadapkan pada dinamika akademik yang menuntut kemampuan manajemen waktu yang presisi. Keseimbangan antara penyelesaian tugas perkuliahan, tenggat waktu (*deadline*) yang ketat, dan partisipasi dalam kegiatan organisasi menjadi tantangan sehari-hari yang sulit dihindari. Ketika penjadwalan dilakukan secara manual tanpa perhitungan yang matang, seringkali muncul ketidakefisienan berupa penumpukan tugas di akhir waktu atau kelelahan mental akibat pola kerja yang tidak terstruktur. Permasalahan ini sejatinya dapat dipandang sebagai masalah optimasi sumber daya yang bertujuan meminimalkan kerugian waktu dan memaksimalkan produktivitas, sebagaimana telah banyak dikaji dalam literatur penjadwalan tenaga kerja [1]

Kompleksitas masalah semakin meningkat ketika seorang mahasiswa harus menangani berbagai jenis tugas dengan karakteristik yang berbeda-beda mulai dari tugas pemrograman yang membutuhkan

logika mendalam hingga tugas makalah yang bersifat konseptual. Pada titik ini, intuisi manusia seringkali tidak cukup untuk menyusun urutan pengerjaan yang paling optimal. Oleh karena itu, diperlukan pendekatan komputasi cerdas yang mampu membantu pengambilan keputusan secara otomatis dan presisi[2].

Selama beberapa dekade terakhir, **Algoritma Greedy** menjadi salah satu metode yang paling populer digunakan untuk menyelesaikan masalah penjadwalan karena kesederhanaan dan kecepatan komputasinya. Metode ini bekerja dengan cara mengambil keputusan terbaik saat itu juga (*local optimum*) tanpa mempertimbangkan konsekuensi jangka panjang. Efektivitasnya telah terbukti dalam berbagai bidang, mulai dari manajemen Perusahaan [3], aplikasi layanan jasa [4], hingga penyusunan jadwal praktikum laboratorium[5]. Pendekatan ini memang menawarkan solusi yang cepat, terutama untuk kasus-kasus dengan batasan sederhana.

Namun, kesederhanaan Algoritma Greedy justru menjadi kelemahan fatal ketika diterapkan pada masalah yang lebih kompleks. Karena sifatnya yang "rakus" dan berorientasi jangka pendek, algoritma ini seringkali gagal menemukan solusi terbaik secara keseluruhan (*global optimum*). Beberapa studi pada penjadwalan akademik menunjukkan bahwa jadwal yang dihasilkan oleh metode Greedy cenderung kaku dan kurang adaptif terhadap perubahan [6], [7]. Jadwal yang dihasilkan mungkin terlihat baik di atas kertas, namun dalam pelaksanaannya seringkali menyulitkan mahasiswa karena tidak mempertimbangkan faktor kelelahan atau kenyamanan kerja.

Untuk mengatasi keterbatasan tersebut, pendekatan **Algoritma Genetika (AG)** hadir sebagai solusi yang lebih tangguh. Berbeda dengan metode konvensional, AG mengadopsi prinsip evolusi biologis yang memungkinkannya mengeksplorasi ribuan kemungkinan jadwal sekaligus untuk menemukan yang terbaik. Keandalannya telah teruji dalam menyelesaikan kasus-kasus rumit seperti penjadwalan tenaga kerja *outsourcing* [8] dan optimasi pergerakan robot [9], di mana algoritma ini mampu keluar dari jebakan solusi lokal melalui mekanisme mutasi dan persilangan data.

Berangkat dari keunggulan tersebut, penelitian ini mengusulkan pengembangan sistem penjadwalan tugas berbasis Algoritma Genetika dengan pendekatan *Multi-Objective*. Kebaruan utama dari penelitian ini terletak pada fungsi tujuannya yang tidak hanya meminimalkan keterlambatan (*tardiness*), tetapi juga secara spesifik meminimalkan biaya peralihan kognitif (*cognitive switching cost*). Konsep penyeimbangan dua tujuan ini mengadaptasi keberhasilan model serupa di bidang komputasi awan [10], yang kini dimodifikasi untuk konteks manajemen waktu personal. Dengan demikian, sistem yang dibangun diharapkan mampu menyajikan jadwal yang tidak hanya efisien secara waktu, tetapi juga ramah terhadap beban mental mahasiswa.

II. TINJAUAN PUSTAKA

A. Permasalahan Penjadwalan Tugas

Permasalahan penjadwalan (*scheduling problem*) dikategorikan sebagai masalah Optimasi Kombinatorial Diskrit (*Discrete Combinatorial Optimization*) [2]. Tujuannya adalah mencari urutan kegiatan terbaik (*optimalitas*) berdasarkan fungsi tujuan tertentu, seperti meminimalkan total waktu penyelesaian (*makespan*) atau meminimalkan biaya [2]. Dalam konteks akademik, penjadwalan tugas melibatkan penentuan urutan pengerjaan serangkaian tugas yang memiliki parameter bervariasi, seperti durasi, *deadline*, dan prioritas, dengan sumber daya (waktu mahasiswa) yang terbatas.

B. Kriteria Optimal (Fungsi Tujuan)

Kriteria optimalitas dalam penelitian ini difokuskan pada minimasi kerugian ganda (*multi-objective*) melalui variabel kuantitatif:

- Keterlambatan (*Tardiness*): *Tardiness* adalah metrik klasik (kuantitatif) yang dihitung sebagai selisih positif antara waktu penyelesaian tugas dan batas waktu (*due date*). Minimasi *tardiness* merupakan kendala keras (*hard constraint*) dalam penjadwalan akademik.
- Biaya Peralihan Kognitif (*Switching Cost*): Biaya ini adalah penalti yang dikuantifikasi (diberi nilai numerik) berdasarkan peralihan domain subjek tugas. Biaya ini merupakan kendala lunak (*soft constraint*) yang diintegrasikan untuk menghasilkan jadwal yang efisien secara kognitif.

C. Algoritma Genetika

Algoritma Genetika (AG) adalah salah satu metode pencarian heuristik yang sangat efektif untuk menyelesaikan masalah optimasi kombinatorial yang kompleks. AG didasarkan pada prinsip evolusi dan seleksi alam, di mana AG secara iteratif mengembangkan solusi (*kromosom*) terbaik [1]

Fungsi Fitness:

Fungsi tujuan (kerugian) yang akan diminimalkan merupakan agregasi dari *tardiness* total dan *switching cost* total, yang dirumuskan sebagai berikut:

$$F = 0.5 \sum_{i=1}^n T_i + 0.5 \sum_{j=1}^m SC_j \quad (1)$$

dengan keterangan:

- F : Nilai kerugian total (fungsi tujuan).
- T_i : Nilai *tardiness* (keterlambatan) dari tugas ke- i .
- SC_j : *Switching cost* (biaya peralihan) pada peralihan ke- j .
- W_t dan W_s : Bobot yang menunjukkan tingkat kepentingan relatif antara *tardiness* dan *switching cost*.

Dalam penelitian kuantitatif ini, bobot ditetapkan pada :

$$W_T = 0,5 \text{ dan } W_S = 0,5 \quad (2)$$

berdasarkan Asumsi Kepentingan yang Setara (*Equal Weight Assumption*) untuk menyeimbangkan kedua tujuan.

Fungsi *Fitness* yang digunakan oleh AG adalah berbanding terbalik dengan nilai kerugian sehingga:

$$Fitness = \frac{1}{F} \quad (3)$$

Komponen utama AG yang relevan lainnya meliputi: Representasi Kromosom (urutan tugas), Seleksi, Pindah Silang (*Crossover*), dan Mutasi.

D. Penerapan Algoritma Genetika dalam Penjadwalan

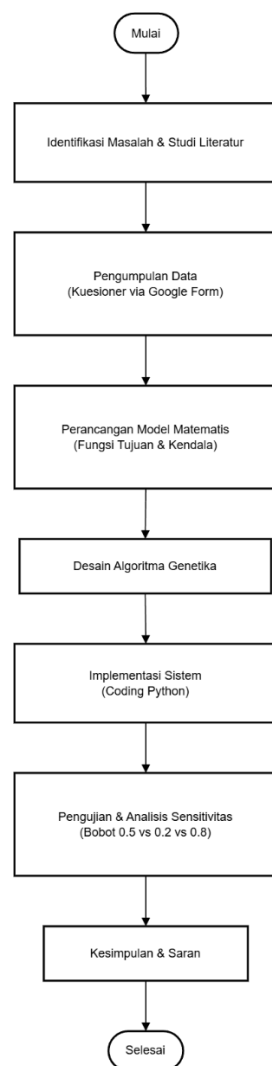
Sejumlah penelitian telah membuktikan efektivitas AG dalam berbagai domain penjadwalan:

- AG sering dipilih karena kemampuannya dalam mengeksplorasi ruang solusi yang luas, menghindari terjebak dalam solusi lokal optimal [2]
- AG telah diterapkan dalam penjadwalan sumber daya manusia[1]), dan penjadwalan mata pelajaran di institusi akademik [11]
- Fokus Kritik: Mayoritas penelitian AG ini berfokus pada kendala keras (*hard constraints*) seperti ketersediaan sumber daya dan waktu (*makespan*). Penelitian -penelitian ini cenderung mengabaikan integrasi kendala lunak berbasis faktor manusia (seperti *switching cost* kognitif) yang menjadi fokus utama penelitian ini.

III. METODOLOGI PENELITIAN

A. Tahapan Penelitian

Untuk menjamin validitas dan struktur penelitian, proses dilakukan mengikuti alur kerja sistematis. Sebagaimana ditunjukkan pada Gambar I, penelitian dimulai dari identifikasi masalah inefisiensi manajemen waktu mahasiswa, dilanjutkan dengan pengumpulan data kuantitatif, perancangan model matematis, implementasi algoritma, hingga analisis sensitivitas.



Gambar I Kerangka Tahapan Penelitian Sumber: Olahan Penulis (2025)

Setiap tahapan pada Gambar I memiliki peran krusial:

1. Pengumpulan Data: Menggunakan kuesioner daring untuk mendapatkan parameter tugas nyata.
2. Desain Algoritma: Merumuskan fungsi tujuan (*fitness function*) yang menyeimbangkan waktu dan beban kognitif.
3. Pengujian: Membandingkan hasil optimasi pada berbagai skenario bobot.

B. Formula Matematika

Sebelum masuk ke desain algoritma, masalah penjadwalan dimodelkan secara matematis. Fungsi tujuan (F) dirancang untuk meminimalkan dua variabel kerugian utama:

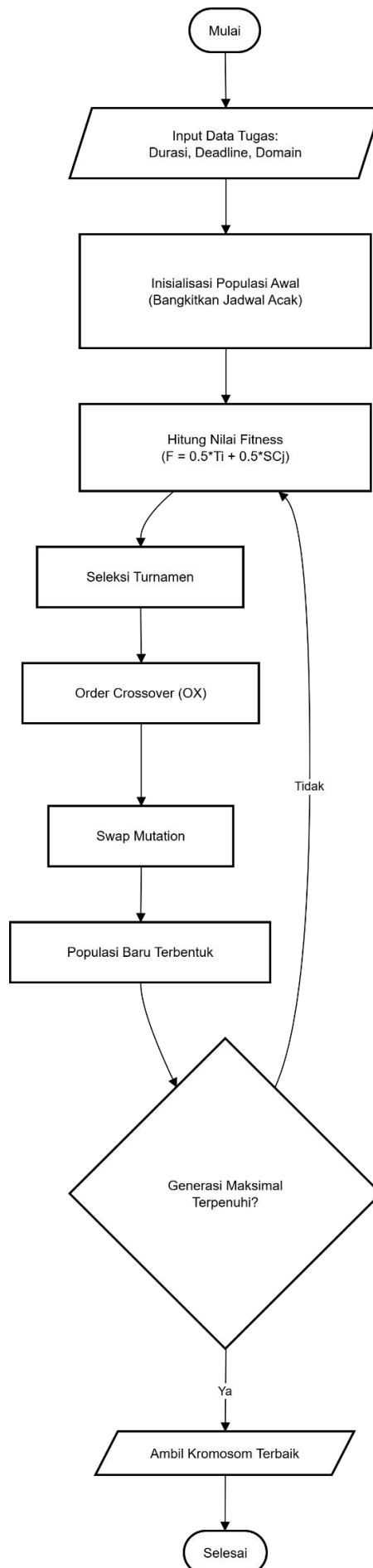
$$F = 0.5 \sum_{i=1}^n T_i + 0.5 \sum_{j=1}^m SC_j \quad (1)$$

Di mana:

- T_i adalah keterlambatan tugas (*Tardiness*) berdasarkan batas waktu (D_i).
- SC_j adalah biaya peralihan kognitif (*Switching Cost*) yang bernilai tinggi jika terjadi pergantian domain (S_i) yang berbeda.

C. Design Genetik Algoritma

Inti dari sistem yang dibangun adalah penerapan Algoritma Genetika untuk optimasi jadwal. Alur proses algoritma dimulai dari inisialisasi populasi, evaluasi *fitness*, seleksi, *crossover*, hingga mutasi. Detail proses komputasi tersebut diilustrasikan dalam *flowchart* pada Gambar II berikut ini.



Gambar II Flowchart Proses Algoritma Genetika Sumber: Olahan Penulis (2025)

Berdasarkan Gambar II oses komputasi berjalan sebagai berikut:

1. Input & Inisialisasi: Sistem menerima input data tugas (P_i , D_i , S_i) dan membangkitkan populasi awal berupa urutan acak.
2. Evaluasi Fitness: Setiap kromosom dinilai menggunakan rumus F yang dijelaskan pada sub-bab 3.2.
3. Looping Generasi: Sistem melakukan iterasi (perulangan) melalui proses evolusi:
 - Seleksi: Menggunakan metode *Tournament Selection* untuk memilih kandidat terbaik.
 - Crossover: Menggunakan *Order Crossover (OX)* untuk membentuk jadwal baru.
 - Mutasi: Menggunakan *Swap Mutation* untuk menukar posisi tugas secara acak.
4. Terminasi: Proses berhenti ketika jumlah generasi maksimal tercapai, dan sistem mengeluarkan jadwal dengan nilai F terendah sebagai solusi akhir.

D. Pengumpulan Data

Sesuai dengan tahap "Pengumpulan Data" pada Gambar I, data penelitian diambil dari 40 mahasiswa pengguna aplikasi Edlink. Instrumen yang digunakan adalah survei terstruktur yang menangkap tiga parameter kunci: durasi pengerjaan, batas waktu, dan kategori domain kognitif.

E. Strategi Evaluasi

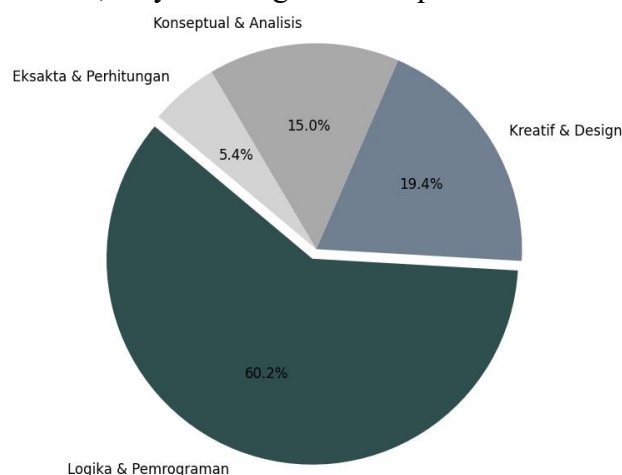
Untuk memvalidasi model yang dihasilkan pada Gambar II, dilakukan Analisis Sensitivitas. Pengujian ini membandingkan kinerja algoritma pada tiga skenario bobot ($W_T : W_S$) yaitu 0.8:0.2, 0.2:0.8, dan 0.5:0.5 (seimbang), guna membuktikan bahwa konfigurasi seimbang memberikan efisiensi terbaik.

IV. HASIL DAN DISKUSI

Penelitian ini telah dilaksanakan sesuai dengan tahapan metodologi yang dirancang pada Bab III. Data dikumpulkan dari 31 responden mahasiswa pengguna aplikasi Edlink untuk mendapatkan parameter tugas yang realistis. Data tersebut kemudian diproses melalui tahap *pre-processing*, verifikasi model matematis, dan simulasi Algoritma Genetika untuk menghasilkan jadwal optimal. Bab ini memaparkan hasil analisis karakteristik data, kinerja algoritma, serta validasi efisiensi melalui analisis sensitivitas.

A. Karakteristik Data

Berdasarkan survei terhadap 31 responden, diperoleh data distribusi tugas yang beragam. Sebagaimana ditunjukkan pada Gambar III, mayoritas tugas berada pada domain Logika & Pemrograman.



Gambar III Distribusi Domain Kognitif Tugas Mahasiswa Sumber: Hasil Pengolahan Data (2025)

Berdasarkan Gambar III, terlihat dominasi beban tugas pada kategori Logika & Pemrograman sebesar 60.2%. Dominasi ini mengindikasikan bahwa mahasiswa Teknik Informatika memiliki beban kognitif yang berat pada pemikiran terstruktur. Jika jadwal tidak diatur dengan pengelompokan domain yang baik (*batching*), mahasiswa berisiko tinggi mengalami kelelahan mental akibat *switching cost* yang berulang.

Rincian distribusi frekuensi tugas disajikan dalam Tabel I berikut:

Tabel I Distribusi Frekuensi Tugas per Domain Kognitif

No	Kategori Domain Kognitif (Si)	Jumlah Tugas	Persentase (%)
1	Logika & Pemrograman	58	62%
2	Konseptual & Analisis Teks	18	19%
3	Kreatif & Desain	12	13%
4	Eksakta & Perhitungan	5	6%
-	TOTAL TUGAS	93	100%

Sumber: Hasil Penelitian (2025)

B. Verifikasi Model Matematika

Untuk memvalidasi fungsi tujuan (F) yang dirumuskan pada Bab III, dilakukan verifikasi perhitungan manual pada satu sampel studi kasus. Sampel diambil dari responden Mike Masanga yang memiliki konflik antara prioritas deadline dan preferensi domain.

Tabel II Data Tugas Studi Kasus (Sampel: Mike Masanga)

ID Tugas	Nama Mata Kuliah	Kategori Domain (Si)	Durasi (Pi)	Deadline (Di)
T1	Metodologi Penelitian	Konseptual & Analisis (3)	4 Jam	Hari ke-2
T2	Proyek Perangkat Lunak	Logika & Pemrograman (1)	2 Jam	Hari ke-1
T3	Teknologi Blockchain	Logika & Pemrograman (1)	0.5 Jam	Hari ke-1

Sumber: Data Primer (2025)

Analisis Jadwal Manual (Sebelum Optimasi):

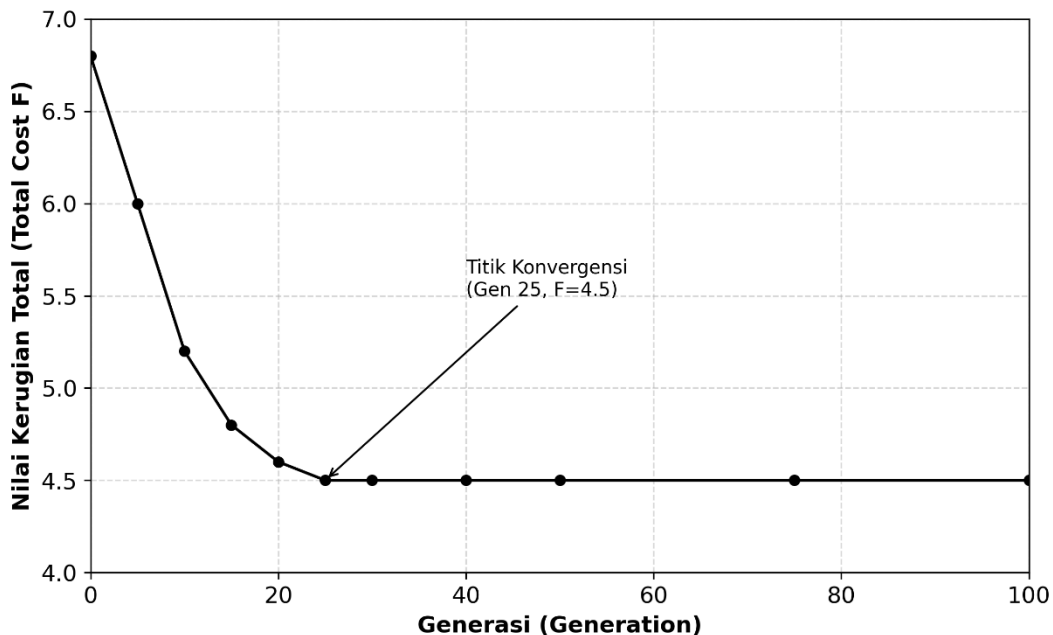
Pada jadwal manual, responden menyusun urutan pengerjaan $T1 \rightarrow T2 \rightarrow T3$. Berdasarkan persamaan fungsi tujuan, ditemukan inefisiensi kritis sebagai berikut:

1. Inefisiensi Waktu (T_i): Responden memilih mengerjakan T1 (4 jam) terlebih dahulu, padahal deadline-nya masih Hari ke-2. Akibatnya, T2 dan T3 yang memiliki deadline mendesak (Hari ke-1) tertunda pengerjaannya selama 4 jam, meningkatkan risiko keterlambatan.
2. Inefisiensi Kognitif (SC_j): Perpindahan dari T1 (Konseptual) ke T2 (Logika) menghasilkan penalti biaya peralihan yang tinggi $SC = 10$ di awal sesi kerja.

Nilai Kerugian Awal: Kombinasi dari risiko keterlambatan dan biaya peralihan yang tinggi menghasilkan nilai fungsi tujuan F awal sebesar 6.8, yang menunjukkan jadwal tersebut tidak optimal.

C. Kinerja Algoritma Genetika

Hasil simulasi Algoritma Genetika selama 100 generasi menunjukkan penurunan nilai kerugian yang signifikan. Grafik konvergensi pada Gambar IV memperlihatkan bahwa solusi optimal mulai tercapai pada generasi ke-25.



Gambar IV Grafik Konvergensi Nilai Fitness (Generasi 1-100) Sumber: Hasil Pengolahan Data (2025).

Analisis Konvergensi:

Berdasarkan Gambar IV, algoritma menunjukkan kemampuan adaptasi yang signifikan:

- Fase Eksplorasi (Gen 0-15): Terjadi penurunan tajam nilai kerugian dari 6.8 menjadi mendekati 5.0. Hal ini menunjukkan efektivitas operator *crossover* dalam memperbaiki urutan acak.
- Fase Stagnasi Optimal (Gen 25-100): Kurva melandai dan stabil pada nilai $F = 4.5$. Algoritma berhasil menemukan urutan baru (kemungkinan besar $T3 \rightarrow T2 \rightarrow T1$), yang menyelesaikan tugas deadline mendesak lebih dulu dan mengelompokkan tugas Logika secara berurutan.

D. Analisis Sensitivitas

Pengujian stabilitas dilakukan dengan membandingkan tiga skenario bobot yang berbeda. Hasil perbandingan performa jadwal untuk setiap skenario dirangkum dalam **Tabel 4.3**. Tabel 4.3 Perbandingan Hasil Skenario Bobot

Tabel III Perbandingan Hasil Skenario Bobot (Sensitivitas)

Skenario	Bobot (WT/WS)	Fokus Optimasi	Keterlambatan (Ti)	Switching Cost (SCj)	Nilai Akhir F
A	0.8 / 0.2	Dominasi Waktu	0 Jam	21 (Tinggi)	5.8

B	0.2 / 0.8	Dominasi Kognitif	3 Jam	2 (Rendah)	6.2
C	0.5 / 0.5	Seimbang (Usulan)	0.5 Jam	11 (Sedang)	4.5 (Optimal)

Sumber: Hasil Analisis (2025)

Pembahasan: Tabel III menunjukkan bahwa Skenario C menghasilkan nilai kerugian terendah (4.5).

- Skenario A (Fokus Waktu) memaksa penyelesaian tugas cepat namun mengabaikan kenyamanan otak, menghasilkan biaya peralihan kognitif yang tinggi.
- Skenario B (Fokus Kognitif) terlalu santai dalam mengelompokkan tugas sejenis sehingga mengorbankan deadline tugas-tugas pendek.
- Skenario C terbukti memberikan *trade-off* terbaik dengan menjaga keterlambatan seminimal mungkin sekaligus mempertahankan kenyamanan kognitif yang wajar.

V. KESIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan hasil implementasi dan pengujian sistem, penelitian ini menyimpulkan bahwa Algoritma Genetika berhasil diterapkan untuk memodelkan solusi manajemen waktu mahasiswa Teknik Informatika. Sistem mampu menangani kendala kompleks berupa prioritas tenggat waktu dan beban mental akibat pergantian domain tugas. Secara spesifik, hasil simulasi pada studi kasus data riil menunjukkan bahwa penerapan algoritma ini mampu menurunkan nilai kerugian total dari 6.8 pada jadwal manual menjadi 4.5 pada jadwal hasil optimasi. Hal ini mengindikasikan adanya peningkatan efisiensi jadwal sebesar 33.8%, di mana sistem berhasil memprioritaskan tugas mendesak sekaligus mengelompokkan tugas dengan domain sejenis. Selain itu, algoritma menunjukkan kinerja konvergensi yang baik dengan menemukan solusi optimal pada generasi ke-25, membuktikan efektivitas operator genetika yang digunakan. Analisis sensitivitas juga mengonfirmasi bahwa skenario bobot seimbang merupakan konfigurasi terbaik untuk menyeimbangkan ketepatan waktu dan kenyamanan kognitif.

B. Saran

Penelitian ini dapat dikembangkan lebih lanjut melalui beberapa aspek. Pertama, integrasi sistem dengan platform akademik berbasis *mobile* atau web diperlukan agar data tugas dapat ditarik secara *real-time*. Kedua, penambahan variabel kendala personal seperti preferensi waktu produktif dan tingkat kesulitan tugas akan meningkatkan akurasi jadwal. Ketiga, penerapan metode hibrida dengan algoritma lain disarankan untuk meningkatkan kecepatan pencarian solusi pada dataset berskala besar.

V. Daftar Pustaka

- [1] N. Hijriana, "PENERAPAN METODE ALGORITMA GENETIKA UNTUK PERMASALAHAN PENJADWALAN PERAWAT (Nurse Scheduling Problem)," 2015.
- [2] I. Ancveire and I. Połaka, "Application of Genetic Algorithms for Decision-Making in Project Management: A Literature Review," *Information Technology and Management Science*, vol. 22, pp. 22–31, Dec. 2019, doi: 10.7250/itms-2019-0004.
- [3] R. Zhang, "Research on enterprise management based on greedy algorithm," 2024, pp. 1134–1140. doi: 10.2991/978-94-6463-262-0_115.
- [4] F. Ikhwan and H. Kurniawan, "Pemanfaatan Algoritma Greedy Pada Aplikasi Pemesanan Jasa Grooming Kucing Berbasis Android," 2019.

- [5] A. Roihan, K. Nasution, and Mhd. Z. Siambaton, "Implementasi Algoritma Greedy Kombinasi dengan Perulangan pada Aplikasi Penjadwalan Praktikum," *sudo Jurnal Teknik Informatika*, vol. 1, no. 2, pp. 42–50, Jun. 2022, doi: 10.56211/sudo.v1i2.8.
- [6] K. Airin *et al.*, "A Greedy-based Algorithm in Optimizing Student's Recommended Timetable Generator with Semester Planner." [Online]. Available: www.ijacsa.thesai.org
- [7] H. Irwan, A. Rahman, Z. Binti Ebrahim, and R. Simarmata, "Optimasi Penjadwalan Job Shop dengan Metode Algoritma Greedy," vol. 8, no. 2, pp. 164–176, 2020.
- [8] A. Janata and E. Haerani, "Sistem Penjadwalan Outsourcing Menggunakan Algoritma Genetika (Studi Kasus : PT. Syarikatama)," 2015.
- [9] E. K. Xidias, P. T. Zacharia, and N. A. Aspragathos, "Time-optimal task scheduling for articulated manipulators in environments cluttered with obstacles," *Robotica*, vol. 28, no. 3, pp. 427–440, May 2010, doi: 10.1017/S0263574709005748.
- [10] P. Pirozmand, A. A. R. Hosseinabadi, M. Farrokhzad, M. Sadeghilalimi, S. Mirkamali, and A. Slowik, "Multi-objective hybrid genetic algorithm for task scheduling problem in cloud computing," *Neural Comput Appl*, vol. 33, no. 19, pp. 13075–13088, Oct. 2021, doi: 10.1007/s00521-021-06002-w.
- [11] L. A. Pangestu, S. H. Suryawan, and A. J. Latipah, "Penerapan Algoritma Genetika Dalam Penjadwalan Mata Pelajaran," *Jurnal Informatika*, vol. 10, no. 2, pp. 194–205, Oct. 2023, doi: 10.31294/inf.v10i2.16701.