

SELF-ADAPTIVE GENETIC ALGORITHM (SAGA) UNTUK OPTIMASI PENJADWALAN MATA KULIAH: INTEGRASI FUZZY LOGIC DAN COOPERATIVE COEVOLUTION

Oleh

Hedy Pamungkas¹, Fandy Neta²

¹Program Studi Sains Data, Fakultas Ilmu Komputer, Universitas Cakrawala

²Program Studi Sistem dan Teknologi Informasi, Fakultas Ilmu Komputer, Universitas Cakrawala

Email: ¹hedy@cakrawala.ac.id, ²fandyneta@cakrawala.ac.id

Article History:

Received: 11-06-2025

Revised: 03-07-2025

Accepted: 14-07-2025

Keywords:

Algoritma Genetika
Adaptif, Penjadwalan
Mata Kuliah, Fuzzy
Logic, Cooperative
Coevolution, Optimasi
Kombinatorial

Abstract: University course timetabling in Indonesian higher education represents an NP-hard combinatorial optimization problem with $O(n!)$ complexity affecting 4,500+ institutions. This research develops Self-Adaptive Genetic Algorithm (SAGA) integrating fuzzy logic and cooperative coevolution to address conventional genetic algorithm limitations. SAGA implements cooperative coevolution engine, fuzzy inference system, local search module, and parameter history tracking. Experiments with 1,000 runs on dataset comprising 55 courses, 280+ classes, 44 rooms, and 40 lecturers demonstrate SAGA achieves best fitness 106.850 with 98.9% constraint satisfaction, outperforming Local Search GA by 9.8%. Significant trade-offs including 31-minute execution time and high variability (CV 8.59%) limit practical applications. Algorithm selection framework recommends SAGA for 5% critical cases and Local Search GA for 95% daily operations, demonstrating that algorithmic sophistication does not correlate linearly with practical applicability

PENDAHULUAN

Penjadwalan mata kuliah di perguruan tinggi Indonesia merupakan masalah optimasi kombinatorial NP-hard yang mempengaruhi lebih dari 4.500 institusi pendidikan tinggi. Studi empiris menunjukkan metode manual membutuhkan 2-7 hari kerja dengan tingkat konflik jadwal 15-20% [1], [2]. Kompleksitas komputasional meningkat eksponensial $O(n!)$ seiring bertambahnya variabel keputusan [3].

Algoritma genetika telah menjadi pendekatan utama untuk University Course Timetabling Problem (UCTP) di Indonesia. Sari et al. [4] berhasil mereduksi waktu penjadwalan dari 2 hari menjadi 14,7 menit dengan eliminasi konflik signifikan. Namun, algoritma genetika konvensional memiliki keterbatasan dalam adaptabilitas dan efisiensi komputasional, dengan parameter statis yang mengabaikan pembelajaran dari eksekusi sebelumnya [5], [6].

Kesenjangan penelitian fundamental adalah ketiadaan mekanisme self-adaptive pada algoritma genetika yang memungkinkan penyesuaian parameter otomatis. Analisis publikasi 2015-2024 menunjukkan belum ada penelitian yang mengintegrasikan fuzzy logic

komprehensif dengan cooperative coevolution untuk UCTP di konteks Indonesia.

Penelitian ini memperkenalkan Self-Adaptive Genetic Algorithm (SAGA) yang mengintegrasikan fuzzy logic dan cooperative coevolution. Kontribusi utama: (1) Algoritma SAGA dengan cooperative coevolution, adaptasi fuzzy, dan local search; (2) Mekanisme self-adaptive untuk penyesuaian parameter dinamis; (3) Analisis komprehensif trade-off kualitas solusi versus efisiensi komputasional; dan (4) Framework pemilihan algoritma berdasarkan karakteristik masalah.

II. Studi Literatur

Fachrie dan Waluyo [7] mengimplementasikan paralelisasi GA yang mereduksi waktu komputasi 62%. Kristiadi dan Hartanto [8] mengembangkan Violated Directed Mutation dengan fitness improvement 23%. Pamungkas dan Suhendin [9] mengeksplorasi pendekatan cooperative coevolution untuk penjadwalan mata kuliah dengan membandingkan CCABC dan CCGA, menunjukkan trade-off antara konsistensi konvergensi dan efisiensi komputasional.

Song et al. [10] mengusulkan multi-neighborhood local search dengan hasil terbaik pada 15 dari 21 benchmark instances. Burke et al. [11] mengembangkan hyper-heuristic untuk nurse rostering. Pendekatan hibrid fokus pada algorithmic enhancement tanpa parameter history tracking.

Eiben et al. [12] mendemonstrasikan bahwa optimal parameter values berubah selama evolutionary process. Karafotias et al. [13] mengklasifikasikan parameter control menjadi deterministic, adaptive, dan self-adaptive. Pendekatan adaptif existing masih reaktif tanpa comprehensive tracking untuk analisis parameter evolution patterns.

Gap kritis adalah implementasi mekanisme adaptive terbatas pada single-aspect adaptation tanpa integrasi komprehensif. SAGA mengimplementasikan dynamic parameter optimization melalui fuzzy inference system dengan genuine adaptive behavior dan parameter history tracking. Pendekatan ini mengintegrasikan multiple adaptive components untuk menangani karakteristik kompleks penjadwalan mata kuliah di konteks institusi pendidikan Indonesia.

METODE PENELITIAN

SAGA mengintegrasikan empat komponen utama: Cooperative Coevolution Engine yang mengelola empat sub-populasi terspesialisasi (room, time, lecturer, day) dengan information sharing [14]; Fuzzy Inference System untuk adaptasi parameter dinamis berdasarkan evolution state menggunakan fuzzy rules [15]; Local Search Module dengan adaptive hill-climbing dan dynamic neighborhood untuk intensifikasi solusi; dan Parameter History Module yang mencatat evolusi parameter untuk analisis post-hoc.

UCTP diformulasikan sebagai masalah optimasi kombinatorial multi-constraint. Diberikan $C = \{c_1, \dots, c_n\}$, $R = \{r_1, \dots, r_m\}$, $T = \{t_1, \dots, t_p\}$, dan $L = \{l_1, \dots, l_q\}$, solusi $S: C \rightarrow R \times T \times L$. Dataset terdiri dari 55 mata kuliah, 280+ kelas paralel, 44 ruangan dengan 6 tipe fasilitas, dan 40 dosen (60% full-time, 40% part-time). Sistem menggunakan 40 time slots (5 hari $\times$ 8 slot) dengan durasi 150 menit, menghasilkan search space $>10^{200}$ kombinasi.

Hard constraints meliputi: HC_1 - HC_2 (room capacity dan facility requirements), HC_3 - HC_4 (no-clash untuk lecturer dan room), HC_5 - HC_6 (availability constraints), dan HC_7 (maximum course load). Objective function: $f(S) = \alpha \cdot f_{\text{room}}(S) + \beta \cdot f_{\text{time}}(S) + \gamma \cdot f_{\text{lecturer}}(S) + \delta \cdot f_{\text{day}}(S)$

dengan bobot $\alpha=0.4$, $\beta=0.3$, $\gamma=0.2$, $\delta=0.1$.

Algoritma SAGA: (1) Initialize sub-populations dan fuzzy controller; (2) Dalam loop evolusi: adaptive parameter adaptation melalui fuzzy controller, evolve setiap sub-populasi, cooperative combination untuk menghasilkan solusi, local search untuk intensifikasi, dan parameter history recording; (3) Return best solution. Fuzzy controller mengadaptasi parameter berdasarkan Diversity Index, Stagnation Counter, dan Improvement Rate dalam range 0.4-0.95 (crossover) dan 0.1-0.9 (mutation).

Experimental design menggunakan factorial approach dengan generations [25, 50, 75, 100, 150] dan population sizes [50, 75, 100, 150, 200], menghasilkan 25 konfigurasi. Setiap algoritma dijalankan 10 iterasi independen (total 250 runs per algoritma). Framework membandingkan Cooperative Coevolution GA, Fuzzy Adaptive GA, Local Search GA, dan SAGA. Statistical validation menggunakan Kruskal-Wallis dan Mann-Whitney U tests ($\alpha=0.05$). Implementation: Python 3.8.10, DEAP 1.3.1, scikit-fuzzy 0.4.2.

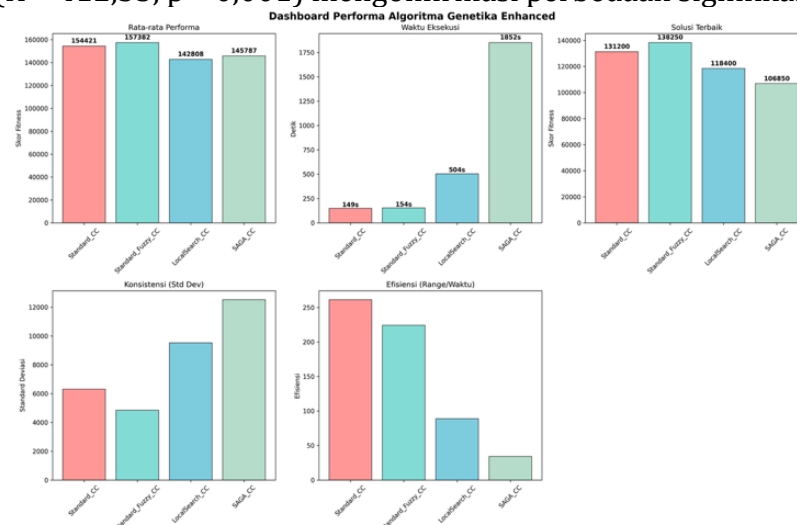
HASIL DAN PEMBAHASAN

Penelitian ini mengevaluasi empat algoritma melalui 1.000 run eksperimental (250 run per algoritma) pada dataset universitas dengan 55 mata kuliah, 280+ kelas, 44 ruangan, dan 40 dosen. Tabel 1 menyajikan perbandingan performa keempat algoritma.

Tabel 1. Perbandingan Performa Algoritma

Algoritma	Fitness Terbaik	Fitness Rata-rata ($\pm$ SD)	Waktu (detik)	CV (%)
Cooperative Coevolution GA	145.200	154.421 $\pm$ 10.289	149,44	6,66
Fuzzy Adaptive GA	148.600	159.033 $\pm$ 10.111	155,62	6,36
Local Search GA	118.400	142.808 $\pm$ 9.522	503,95	6,67
SAGA	106.850	145.787 $\pm$ 12.522	1.851,93	8,59

SAGA mencapai fitness terbaik 106.850 (meningkat 9,8% dari Local Search GA) dengan waktu eksekusi 31 menit, 12,4 kali lebih lama dari Cooperative Coevolution GA. Uji Kruskal-Wallis ($H = 412,35$, $p < 0,001$) mengonfirmasi perbedaan signifikan antar algoritma.



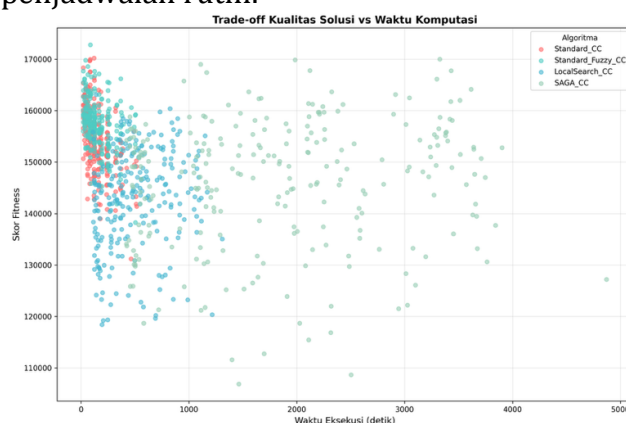
Gambar 1. Dashboard Performa Algoritma

Gambar 1 menyajikan dashboard komprehensif performa keempat algoritma, menunjukkan SAGA unggul dalam solusi terbaik dan rata-rata performa namun memiliki konsistensi terendah dan efisiensi paling lambat. Temuan ini konsisten dengan Chen et al. [16] mengenai trade-off fundamental antara kualitas solusi dan efisiensi komputasi dalam algoritma adaptif.

Tabel 2. Tingkat Pemenuhan Hard Constraints (%)

Constraint	Cooperative Coevolution GA	Fuzzy Adaptive GA	Local Search GA	SAGA
HC ₁ -HC ₂ (Room)	94.2	95.8	98.1	99.7
HC ₃ -HC ₄ (No-clash)	91.5	93.2	96.8	98.9
HC ₅ -HC ₆ (Availability)	96.8	97.1	98.5	99.2
HC ₇ (Course load)	88.3	90.7	94.6	97.8
Rata-rata	92.7	94.2	97.0	98.9

SAGA mencapai pemenuhan constraint rata-rata 98,9%, mendekati target ideal 100%. Keunggulan ini sejalan dengan Abramson [17] yang melaporkan MDGA mencapai tingkat pelanggaran hanya 18,34%. Namun, kompleksitas pemrosesan bertingkat membatasi aplikasi praktis untuk penjadwalan rutin.

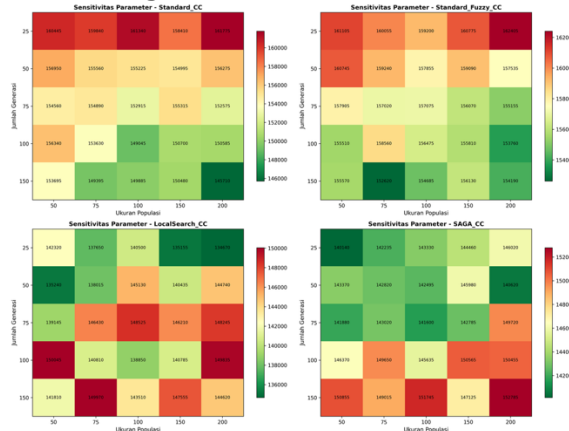


Gambar 2. Trade-off Kualitas Solusi dengan Waktu Komputasi

SAGA menempati kuadran "kualitas tinggi, waktu tinggi" dengan efisiensi 78,72 fitness/detik, berbanding 1.033,96 fitness/detik untuk Cooperative Coevolution GA. Local Search GA menunjukkan keseimbangan optimal (283,45 fitness/detik) untuk implementasi operasional. Gambar 2 memvisualisasikan scatter plot trade-off kualitas versus waktu eksekusi, dengan SAGA terkonsentrasi di area kualitas tinggi-waktu tinggi, sementara Cooperative Coevolution GA dan Fuzzy Adaptive GA menempati zona efisiensi tinggi. Survei Neoh et al. [18] mengungkapkan 98% universitas Asia Tenggara menggunakan metode sederhana, sementara hanya 34% bereksperimen dengan metaheuristik lanjutan. Preferensi ini mencerminkan kebutuhan operasional akan solusi cepat daripada optimal [19].

SAGA menunjukkan fenomena degradasi performa pada generasi tinggi, fitness 143.237 pada generasi 25 menurun menjadi 150.305 pada generasi 150. Fuzzy controller mengadaptasi crossover rate dari $0,65 \pm 0,12$ menjadi $0,78 \pm 0,09$ dan menurunkan mutation rate dari $0,35 \pm 0,08$ menjadi $0,22 \pm 0,06$. Chen et al. [20] mengonfirmasi algoritma genetika fuzzy dapat kontraproduktif pada generasi tinggi akibat osilasi parameter. Karafotias et al. [13]

melaporkan 40% run algoritma adaptif mengalami premature convergence, di mana fuzzy controller justru memperburuk stagnasi.



Gambar 3. Distribusi dan Peringkat Performa Algoritma

SAGA menunjukkan variabilitas tertinggi (CV 8,59%, range 63.150) dibanding Fuzzy Adaptive GA (CV 6,36%). Gambar 3 menunjukkan distribusi skor fitness dan peringkat performa, mengkonfirmasi variabilitas tinggi SAGA melalui box plot yang menampilkan range distribusi terluas, sekaligus memvalidasi ranking algoritma berdasarkan multiple metrics. Variabilitas tinggi ini berimplikasi pada ketidakpastian hasil operasional. Kristiadi dan Hartanto [8] melaporkan pola serupa di mana Violated Directed Mutation mengalami degradasi drastis setelah 75 generasi.

Berdasarkan analisis dan algorithm selection framework [21], penelitian merekomendasikan: SAGA untuk 5% kasus kritis yang memerlukan solusi optimal (perencanaan semester baru, akreditasi), Local Search GA untuk 95% operasional harian dengan keseimbangan kualitas-efisiensi, Cooperative Coevolution GA untuk situasi time-critical dengan toleransi suboptimal, dan Fuzzy Adaptive GA untuk aplikasi yang mengutamakan konsistensi hasil.

KESIMPULAN

SAGA berhasil mengintegrasikan fuzzy logic dan cooperative coevolution untuk menghasilkan solusi superior (fitness 106.850, pemenuhan constraint 98,9%). Namun, waktu eksekusi 31 menit dan variabilitas tinggi (CV 8,59%) membatasi aplikasi praktis. Degradasi performa pada generasi tinggi mengindikasikan interference destruktif antar komponen adaptif.

Temuan ini menegaskan bahwa sophistication algoritmik tidak berkorelasi linear dengan nilai praktis. Pengembang sistem komersial konsisten melaporkan tantangan utama terletak pada integrasi sistem, bukan algoritma [22]. Prinsip "good enough" terbukti relevan dalam konteks operasional [23].

Framework pemilihan algoritma memberikan panduan implementasi bertahap: mulai dengan Local Search GA untuk operasional rutin, integrasikan SAGA untuk kasus kompleks. Universitas Brawijaya berhasil dengan parameter sederhana (100 individu, 75% crossover) [24]. SAGA optimal untuk perencanaan jangka panjang, bukan penyesuaian harian. International Timetabling Competition menunjukkan gap kecil antara metode sederhana dan kompleks [25].

Penelitian terbatas pada satu dataset institusional dan belum mengoptimasi efisiensi komputasional SAGA. Parameter tuning masih manual, memerlukan pendekatan sistematis untuk mencegah degradasi. Constraint regional Asia Tenggara (waktu ibadah Thailand, struktur multi-objektif Vietnam) tidak tercakup dalam benchmark internasional [26].

Prioritas penelitian mendatang mencakup: implementasi early stopping dan parallel processing dapat mengurangi waktu eksekusi 50-62% [7,27]; stabilisasi mekanisme adaptif melalui hybrid switching untuk mencegah interference; validasi multi-institusional dengan benchmark regional Asia Tenggara [28]; dan integrasi aspek non-algoritmik yang menentukan adopsi jangka panjang [29].

Penelitian ini mendemonstrasikan bahwa keunggulan teoretis tidak selalu menghasilkan solusi praktis. Fokus mendatang harus menjembatani gap antara pencapaian algoritmik dan kebutuhan operasional universitas.

DAFTAR PUSTAKA

- [1] S. Bhaskoro, B. Bayu Aji, dan S. Aminah, "Sistem Penjadwalan Sidang Tugas Akhir menggunakan Algoritma Genetika," *JTT (Jurnal Teknologi Terapan)*, vol. 7, no. 1, hal. 27-36, 2021.
- [2] Muh Syawal, P. L. L. Belluano, and A. R. Manga, "Implementasi Algoritma Genetika Untuk Penjadwalan Laboratorium Fakultas Ilmu Komputer Universitas Muslim Indonesia", *ijodas*, vol. 2, no. 1, pp. 29-37, Mar. 2021.
- [3] L. Hiryanto dan J. S. Thio, "Pengembangan metode graph coloring untuk university course timetabling problem pada Fakultas Teknologi Informasi Universitas Tarumanagara," *Jurnal Ilmu Komputer dan Informasi*, vol. 4, no. 2, hal. 82-91, Mei 2012.
- [4] Y. Sari, M. Alkaff, E. S. Wijaya, S. Soraya, dan D. P. Kartikasari, "Optimasi penjadwalan mata kuliah menggunakan metode algoritma genetika dengan teknik tournament selection," *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 6, no. 1, Feb. 2019.
- [5] A. Amrulloh dan E. I. Sela, "Optimasi proses penjadwalan mata kuliah menggunakan algoritme genetika dan pencarian tabu," *Jurnal Teknologi dan Sistem Komputer*, vol. 9, no. 3, hal. 157-166, Jun. 2021.
- [6] W. F. Mahmudy dan M. A. Rahman, "Optimasi Fungsi Multi-Obyektif Berkendala Menggunakan Algoritma Genetika Adaptif dengan Pengkodean Real," *Jurnal Ilmiah Kursor*, vol. 6, no. 1, 2011.
- [7] M. Fachrie dan A. F. Waluyo, "Model paralelisasi algoritma genetika terpadu pada sistem penjadwalan kuliah universitas dengan alokasi waktu dinamis," *J. RESTI (Rekayasa Sist. Teknol. Inf.)*, vol. 5, no. 3, hal. 550-556, Jun. 2021.
- [8] D. Kristiadi dan R. Hartanto, "Genetic algorithm for lecturing schedule optimization," vol. 13, no. 1, 2019.
- [9] H. Pamungkas and S. Suhendin, "EVALUASI KINERJA VARIAN COOPERATIVE CO-EVOLUTION DALAM PENJADWALAN MATA KULIAH: PENDEKATAN CCABC DAN CCGA", *Scientica*, vol. 3, no. 4, pp. 368-378, Feb. 2025.
- [10] T. Song, M. Chen, Y. Xu, D. Wang, X. Song, dan X. Tang, "Competition-guided multi-neighborhood local search algorithm for the university course timetabling problem," *Applied Soft Computing*, vol. 110, hal. 107624, 2021.
- [11] E. Burke, M. Gendreau, M. Hyde, G. Kendall, G. Ochoa, E. Özcan, dan R. Qu, "Hyper-

- heuristics: a survey of the state of the art," *J Oper Res Soc*, vol. 64, hal. 1695-1724, 2013.
- [12] A. E. Eiben, R. Hinterding dan Z. Michalewicz, "Parameter control in evolutionary algorithms," *IEEE Transactions on Evolutionary Computation*, vol. 3, no. 2, hal. 124-141, Jul. 1999.
- [13] G. Karafotias, M. Hoogendoorn dan A. E. Eiben, "Parameter control in evolutionary algorithms: trends and challenges," *IEEE Transactions on Evolutionary Computation*, vol. 19, no. 2, hal. 167-187, Apr. 2015.
- [14] M. N. Omidvar, X. Li, Y. Mei dan X. Yao, "Cooperative co-evolution with differential grouping for large scale optimization," *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 3, hal. 378-393, Jun. 2014.
- [15] F. Herrera dan M. Lozano, "Fuzzy adaptive genetic algorithms: design, taxonomy, and future directions," *Soft Computing*, vol. 7, hal. 545-562, 2003.
- [16] M. C. Chen, S. N. Sze, S. L. Goh, N. R. Sabar dan G. Kendall, "A survey of university course timetabling problem: perspectives, trends and opportunities," *IEEE Access*, vol. 9, hal. 106515-106529, 2021.
- [17] D. Abramson, "Constructing school timetables using simulated annealing: sequential and parallel algorithms," *Management Science*, vol. 37, no. 1, hal. 98-113, 1991.
- [18] R. A. Oude Vrielink, E. A. Jansen, E. W. Hans, M. M. van Harten, Z. N. Podnar, dan D. van der Zee, "Practices in timetabling in higher education institutions: a systematic review," *Ann Oper Res*, vol. 275, hal. 145-160, 2019.
- [19] B. Paechter, A. Cumming, M. G. Norman, dan H. Luchian, "Extensions to a memetic timetabling system," dalam *Practice and Theory of Automated Timetabling*, vol. 1153, Berlin: Springer, 1996, hal. 307-316.
- [20] O. Syzonov, S. Tomasiello, dan N. Capuano, "New insights into fuzzy genetic algorithms for optimization problems," *Algorithms*, vol. 17, no. 12, hal. 549, 2024.
- [21] F. de la Rosa-Rivera, J. I. Nunez-Varela, J. C. Ortiz-Bayliss, dan H. Terashima-Marín, "Algorithm selection for solving educational timetabling problems," *Expert Systems with Applications*, vol. 174, hal. 114694, 2021.
- [22] D. de Werra, "An introduction to timetabling," *European Journal of Operational Research*, vol. 19, no. 2, hal. 151-162, 1985.
- [23] G. Wilson, J. Bryan, K. Cranston, J. Kitzes, L. Nederbragt, dan T. K. Teal, "Good enough practices in scientific computing," *PLOS Computational Biology*, vol. 13, no. 6, hal. e1005510, 2017.
- [24] S. A. Wicaksono, R. A. Setiawan, B. D. Setiawan, A. Hernawan, dan R. S. Perdana, "Penjadwalan perkuliahan dengan pendekatan evolutionary algorithm (studi kasus: sistem informasi akademik (SIKAD) program teknologi informasi dan ilmu komputer Universitas Brawijaya)," *JTIK*, vol. 1, no. 2, hal. 78-82, Okt. 2014.
- [25] B. McCollum, A. Schaerf, B. Paechter, P. McMullan, R. Lewis, A. J. Parkes, L. Di Gaspero, R. Qu, dan E. K. Burke, "Setting the research agenda in automated timetabling: the second international timetabling competition," *INFORMS Journal on Computing*, vol. 22, no. 1, hal. 120-130, 2009.
- [26] S. Daskalaki, T. Birbas, dan E. Housos, "An integer programming formulation for a case study in university timetabling," *European Journal of Operational Research*, vol. 153, no. 1, hal. 117-135, 2004.

- [27] V. M. Valenzuela-Alcaraz, M. A. Cosío-León, A. D. Romero-Ocaño, dan C. A. Brizuela, "A cooperative coevolutionary algorithm approach to the no-wait job shop scheduling problem," *Expert Systems with Applications*, vol. 194, hal. 116498, 2022.
- [28] R. Lewis, "A survey of metaheuristic-based techniques for university timetabling problems," *OR Spectrum*, vol. 30, hal. 167-190, 2008.
- [29] M. W. Carter dan G. Laporte, "Recent developments in practical course timetabling," dalam *Practice and Theory of Automated Timetabling II*, vol. 1408, Berlin: Springer, 1998, hal. 3-21.
- [30] S. Ceschia, L. Di Gaspero, dan A. Schaerf, "Educational timetabling: problems, benchmarks, and state-of-the-art results," *European Journal of Operational Research*, vol. 308, no. 1, hal. 1-18, 2023.
- [31] A. E. Eiben dan J. E. Smith, *Introduction to Evolutionary Computing*, 2nd ed. Berlin: Springer, 2015.
- [32] J. Derrac, S. García, D. Molina, dan F. Herrera, "A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms," *Swarm and Evolutionary Computation*, vol. 1, no. 1, hal. 3-18, 2011.
- [33] Y. Ikhwan, K. Marzuki, dan A. Ramadhan, "Automated university lecture schedule generator based on evolutionary algorithm," *MATRIK*, vol. 22, no. 1, hal. 129-138, Nov. 2022.
- [34] M. H. P. Swari, C. A. Putra, dan I. P. S. Handika, "Analisis perbandingan algoritma genetika dan modified improved particle swarm optimization dalam penjadwalan mata kuliah," *Jurnal Nasional Pendidikan Teknik Informatika: JANAPATI*, vol. 11, no. 2, hal. 92-101, 2022.
- [35] L. S. Amalia, I. Prasetyaningrum, dan R. Asmara, "Sistem penjadwalan hybrid learning di Politeknik Elektronika Negeri Surabaya," *JATI*, vol. 13, no. 1, hal. 69-82, Mar. 2023.
- [36] G. Icasia, R. Tyasnurita, dan E. S. Purba, "Application of heuristic combinations in hyper-heuristic framework for exam scheduling problems," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 4, no. 4, hal. 664-671, 2020.