

OPTIMASI PENJADWALAN KULIAH FAKULTAS TEKNIK MENGUNAKAN PENDEKATAN HIBRIDA GREEDY DAN CONSTRAINT SATISFACTION PROBLEM

Rezki Asriani¹⁾, Rizki Yusliana Bakti²⁾, Titin Wahyuni³⁾

1. Informatika, Universitas Muhammadiyah Makassar
email: 105841111522@student.unismuh.ac.id
2. Informatika, Universitas Muhammadiyah Makassar
email: rizkiyusliana@unismuh.ac.id
3. Informatika, Universitas Muhammadiyah Makassar
email: titinwahyuni@unismuh.ac.id

Abstract

Course scheduling is a complex academic process that must consider the availability of lecturers, classrooms, time slots, student groups, and specific regulations of each study program. This study aims to implement the Greedy Algorithm combined with a Constraint Satisfaction Problem (CSP) approach to optimize course scheduling in the Faculty of Engineering. The system was developed using Python and tested using scheduling data from the Even Semester of the 2025/2026 Academic Year at the Faculty of Engineering, Universitas Muhammadiyah Makassar. The dataset consisted of 432 courses from six study programs, 136 lecturers, 16 classrooms, seven operational days, and five class sessions per day. The Greedy Algorithm was employed to generate an initial schedule using a first-fit strategy based on priority and slot availability. Subsequently, CSP was applied to model and resolve constraints, including lecturer conflicts, room conflicts, student conflicts, Zoom-class regulations, MKDU scheduling rules, non-regular class schedules, and fixed PWK schedules. The experimental results indicate that the proposed system successfully generated a conflict-free schedule with a Constraint Satisfaction Rate of 100%. All seven hard constraints were satisfied without any lecturer, room, or student conflicts. The core scheduling process required only 0.679 seconds and successfully resolved nine room conflicts within two iterations. These findings demonstrate that the integration of the Greedy Algorithm and CSP is effective in producing valid, efficient, and adaptive course schedules that meet the academic requirements of the Faculty of Engineering.

Keywords: Greedy Algorithm; **Constraint** Satisfaction Problem; Course Scheduling.

A. PENDAHULUAN

Manajemen pendidikan merupakan aspek penting dalam menjamin terselenggaranya proses pembelajaran yang efektif dan efisien di perguruan tinggi. Pengelolaan akademik yang sistematis dapat membantu pemanfaatan sumber daya, pengaturan ruang kelas, serta peningkatan kualitas pembelajaran [13]. Salah satu komponen penting dalam manajemen akademik adalah penjadwalan perkuliahan. Jadwal kuliah menjadi penghubung antara perencanaan akademik dan pelaksanaan pembelajaran di kelas.

Penjadwalan yang tidak tertata dapat menimbulkan berbagai kendala, seperti bentrok jadwal dosen, bentrok penggunaan ruang, tumpang tindih jadwal mahasiswa, serta kurang optimalnya pemanfaatan waktu perkuliahan. Kondisi tersebut dapat memengaruhi efektivitas belajar mahasiswa, terutama karena mahasiswa memiliki beban akademik yang padat dan membutuhkan pengelolaan waktu yang baik [32]. Kalender akademik memang menjadi pedoman umum pelaksanaan kegiatan pendidikan, tetapi efektivitasnya tetap bergantung pada kualitas penyusunan jadwal operasional perkuliahan [9].

Permasalahan penjadwalan kuliah termasuk dalam masalah optimasi kombinatorial karena melibatkan banyak variabel, seperti mata kuliah, dosen, ruang, kelas, hari, sesi, dan aturan akademik. Kompleksitas tersebut meningkat ketika sistem harus memenuhi berbagai batasan secara bersamaan. Beberapa penelitian menunjukkan bahwa masalah penjadwalan dapat diselesaikan menggunakan pendekatan optimasi seperti algoritma genetika, ant colony optimization, integer linear programming, constraint programming, dan algoritma greedy [15], [21], [25], [26], [28], [30].

Algoritma *Greedy* merupakan salah satu pendekatan yang sering digunakan karena memiliki proses komputasi yang sederhana dan cepat. Algoritma ini bekerja dengan memilih solusi terbaik secara lokal pada setiap tahap penyusunan jadwal. Beberapa penelitian sebelumnya menunjukkan bahwa *Greedy* dapat digunakan dalam penjadwalan mata kuliah, penjadwalan praktikum, dan pengembangan aplikasi jadwal kuliah berbasis web [5], [11], [20]. Namun, kelemahan utama *Greedy* adalah tidak selalu mampu menjamin solusi bebas konflik pada permasalahan dengan *constraint* yang kompleks.

Di sisi lain, *Constraint Satisfaction Problem* (CSP) mampu memodelkan masalah penjadwalan melalui variabel, domain, dan *constraint*. Pendekatan ini sesuai untuk memastikan bahwa jadwal yang dihasilkan memenuhi batasan wajib, seperti tidak adanya bentrok dosen, bentrok ruang, dan bentrok jadwal mahasiswa [3], [14], [31]. Namun, CSP murni dapat membutuhkan proses pencarian yang lebih besar apabila jumlah data dan kombinasi slot semakin banyak.

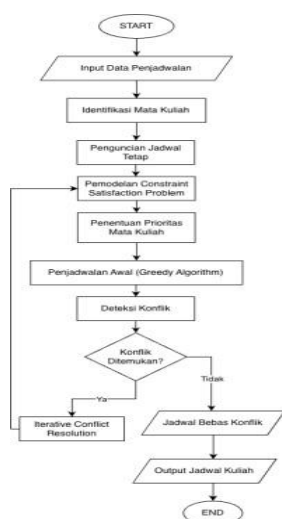
Berdasarkan kondisi tersebut, penelitian ini mengusulkan implementasi pendekatan hibrida antara Algoritma *Greedy* dan *Constraint Satisfaction Problem* untuk optimasi penjadwalan kuliah Fakultas Teknik. Algoritma *Greedy* digunakan untuk menghasilkan jadwal awal secara cepat, sedangkan CSP digunakan untuk memeriksa dan menyelesaikan konflik yang muncul. Pendekatan ini diharapkan mampu menghasilkan jadwal kuliah yang bebas konflik, efisien secara komputasi, dan sesuai dengan *constraint* akademik yang berlaku.

B. METODE PENELITIAN

Penelitian ini dilaksanakan di Universitas Muhammadiyah Makassar, khususnya pada Laboratorium Informatika, Gedung Iqro Lantai 3. Lokasi ini dipilih karena memiliki

fasilitas yang mendukung proses perancangan, pengembangan, dan pengujian sistem penjadwalan kuliah berbasis Algoritma *Greedy* dan *Constraint Satisfaction Problem* (CSP). Adapun waktu pelaksanaan penelitian dilakukan selama kurang lebih tiga bulan, yaitu mulai Maret 2025 sampai dengan Mei 2025.

Alat yang digunakan dalam penelitian ini meliputi laptop Asus Vivobook dengan spesifikasi RAM 8 GB, processor 12th Gen Intel (R) Core (TM) i5-1235U, 12 core, serta perangkat lunak Python dan Visual Studio Code. Perangkat tersebut digunakan untuk proses perancangan, implementasi, dan pengujian sistem penjadwalan kuliah.



Gambar 1. Flowchart Sistem

Perancangan sistem dilakukan melalui beberapa tahapan utama. Tahap pertama adalah input data penjadwalan yang meliputi data mata kuliah, dosen pengampu, ruang kelas atau laboratorium, kelompok mahasiswa, serta slot waktu yang tersedia. Selanjutnya, sistem melakukan identifikasi mata kuliah dengan mengelompokkan mata kuliah ke dalam dua kategori, yaitu mata kuliah dengan jadwal tetap dan mata kuliah dengan jadwal

fleksibel. Mata kuliah yang memiliki jadwal tetap kemudian dikunci oleh sistem agar tidak berubah selama proses penjadwalan berlangsung.

Setelah data dan jadwal tetap ditentukan, permasalahan penjadwalan dimodelkan menggunakan pendekatan *Constraint Satisfaction Problem* (CSP). Dalam model ini, sistem menetapkan batasan berupa *hard constraint* dan *soft constraint*. *Hard constraint* merupakan aturan yang wajib dipenuhi, seperti larangan dosen mengajar pada dua kelas di waktu yang sama, larangan penggunaan ruang yang sama secara bersamaan, serta larangan terjadinya bentrok jadwal mahasiswa. Sementara itu, *soft constraint* digunakan sebagai preferensi tambahan, seperti pemerataan jadwal kuliah dan kenyamanan waktu mengajar dosen.

Tahap berikutnya adalah penentuan prioritas mata kuliah. Mata kuliah yang memiliki jadwal tetap atau kebutuhan khusus diberikan prioritas lebih tinggi agar dapat dijadwalkan terlebih dahulu. Setelah prioritas ditentukan, sistem menyusun jadwal awal menggunakan Algoritma *Greedy*. Algoritma ini bekerja dengan memilih slot waktu terbaik yang tersedia untuk setiap mata kuliah berdasarkan prioritas dan *constraint* yang telah ditentukan. Pendekatan *Greedy* digunakan karena mampu menghasilkan solusi secara cepat dan efisien, meskipun tidak selalu menjamin solusi global yang paling optimal.

Jadwal awal yang dihasilkan kemudian diperiksa untuk mendeteksi kemungkinan konflik, seperti konflik dosen, konflik penggunaan ruang, maupun konflik jadwal mahasiswa. Apabila ditemukan konflik, sistem melakukan proses penyelesaian konflik secara iteratif melalui penjadwalan ulang terhadap mata kuliah yang bermasalah. Proses ini dilakukan dengan mencari alternatif slot waktu atau ruang yang masih tersedia tanpa mengubah keseluruhan jadwal yang sudah valid. Setelah proses perbaikan dilakukan,

sistem kembali memeriksa jadwal hingga seluruh *hard constraint* terpenuhi dan jadwal dinyatakan bebas konflik. *Output* akhir dari sistem berupa jadwal kuliah dalam bentuk tabel atau laporan yang memuat informasi mata kuliah, dosen pengampu, ruang kelas, dan waktu perkuliahan.

Pengujian sistem dilakukan untuk mengevaluasi kemampuan sistem dalam menghasilkan jadwal kuliah yang bebas konflik dan memenuhi *constraint* yang telah ditentukan. Pengujian dilakukan menggunakan data penjadwalan yang terdiri dari data mata kuliah, dosen pengampu, ruang kelas, kelompok mahasiswa, dan slot waktu yang tersedia. Sistem dijalankan untuk menghasilkan jadwal awal, kemudian dilakukan pemeriksaan terhadap kemungkinan bentrok dosen, ruang, dan waktu perkuliahan. Jika konflik ditemukan, sistem melakukan penjadwalan ulang secara iteratif hingga diperoleh jadwal yang memenuhi seluruh *hard constraint*.

Teknik analisis data dilakukan dengan pendekatan evaluasi kuantitatif terhadap hasil jadwal yang dihasilkan oleh sistem. Aspek yang dianalisis meliputi jumlah konflik jadwal sebelum dan sesudah proses optimasi, pemenuhan *hard constraint*, serta waktu komputasi yang dibutuhkan sistem dalam menghasilkan jadwal. Sistem dinyatakan berhasil apabila tidak terdapat pelanggaran *hard constraint*, seperti bentrok dosen, ruang, dan waktu. Selain itu, efisiensi sistem dianalisis berdasarkan kecepatan sistem dalam menghasilkan jadwal bebas konflik serta kemampuannya dalam mengoptimalkan penggunaan sumber daya, seperti ruang kelas, waktu perkuliahan, dan ketersediaan dosen.

C. HASIL DAN PEMBAHASAN

1. Deskripsi Data Penelitian

Penelitian ini menggunakan data penjadwalan Semester Genap Tahun Akademik 2025/2026 Fakultas Teknik Universitas Muhammadiyah Makassar. Data mencakup 432 mata kuliah dari 6 program studi. Sebanyak 312 mata kuliah dilaksanakan secara luring, sedangkan 120 mata kuliah dilaksanakan melalui Zoom. Mata kuliah berbasis Zoom merupakan mata kuliah semester 2 sesuai kebijakan universitas.

Tabel 1. Rincian Data Mata Kuliah

No	Program Studi	Jumlah	Mode	Mode Zoom
		MK	Lurin	
1	Informatika	169	120	49
2	Elektro	87	69	18
3	Pengairan	84	71	13
4	Arsitektur	42	28	14
5	MKDU	30	10	20
6	PWK	20	14	6
Total		432	312	120

2. Implementasi Sistem ChronoSync

Sistem ChronoSync dikembangkan menggunakan Python 3.10+ dengan struktur modular. Sistem terdiri atas modul konfigurasi, modul inti, modul analisis, dan modul utilitas. Data mata kuliah dibaca dari *file* Excel menggunakan *library* pandas dan openpyxl. Sistem kemudian melakukan parsing data mata kuliah, dosen, SKS, semester, kelas, ruang, hari, dan sesi.

Proses penjadwalan dilakukan melalui lima tahap. Pertama, sistem membaca dan menormalisasi data. Kedua, sistem mengunci jadwal tetap PWK sebanyak 20 mata kuliah. Ketiga, sistem membentuk model *Constraint Satisfaction Problem* dengan variabel mata kuliah, domain berupa kombinasi hari, sesi, dan ruang, serta *constraint* berupa 7 *hard constraint* dan 4 *soft constraint*. Keempat, sistem melakukan penempatan awal menggunakan Algoritma *Greedy* dengan

strategi *first-fit*. Kelima, sistem menjalankan *iterative conflict resolution* untuk mendeteksi dan menyelesaikan konflik.

3. Hasil Penjadwalan Awal dan Resolusi Konflik

Pada tahap penempatan awal, Algoritma *Greedy* berhasil menjadwalkan 412 mata kuliah dalam waktu 0,081 detik. Jumlah ini belum termasuk 20 mata kuliah PWK yang sudah dikunci sejak awal. Namun, hasil awal masih menunjukkan 9 konflik ruang. Konflik tersebut kemudian diselesaikan melalui mekanisme *iterative conflict resolution* berbasis CSP.

Tabel 2. Konvergensi Resolusi Konflik

Iterasi	Konflik Ruang	Konflik Dosen	MK Dipindahkan
0	9	0	9
1	0	0	0

Hasil ini menunjukkan bahwa sistem berhasil menyelesaikan seluruh konflik hanya dalam 2 iterasi. Sebanyak 9 mata kuliah dipindahkan ke slot alternatif tanpa mengubah jadwal tetap PWK.

4. Evaluasi Hard Constraint

Evaluasi *hard constraint* dilakukan untuk memastikan jadwal yang dihasilkan memenuhi aturan wajib. Hasil pengujian menunjukkan bahwa seluruh *hard constraint* terpenuhi.

Tabel 3. Evaluasi Hard Constraint

Kode	Hard Constraint	Jumlah Data	Hasil	Status
HC1	Tidak bentrok Dosen	432 slot	0 konflik	TERPENUHI
HC2	Tidak bentrok Ruang	312 slot	0 konflik	TERPENUHI
HC3	Tidak bentrok mahasiswa	432 slot	0 konflik	TERPENUHI

HC4	Semester awal via Zoom	120 MK	120 Zoom	TERPENUHI
HC5	NR di akhir pekan	Sesuai MK	Sesuai Sabtu	TERPENUHI
HC6	MKDU hari Sabtu	30 MK	30 Sabtu	TERPENUHI
HC7	Jadwal tetap PWK tidak berubah	20 MK	20 tetap	TERPENUHI
Constraint Satisfaction Rate (CSaR) = $\frac{7}{7} \times 100\% = 100\%$				

Hasil ini menunjukkan bahwa jadwal yang dihasilkan valid secara *constraint*. Tidak ditemukan bentrok dosen, ruang, maupun mahasiswa. Jadwal semester awal juga berhasil diarahkan ke mode Zoom. Mata kuliah MKDU ditempatkan pada hari Sabtu, dan jadwal PWK tetap dipertahankan.

5. Evaluasi Conflict Rate

Evaluasi *conflict rate* digunakan untuk mengukur tingkat bentrok pada jadwal akhir. Hasil evaluasi menunjukkan bahwa semua konflik berhasil dieliminasi.

Tabel 4. Evaluasi Conflict Rate

Metrik	Formula	Hasil
Instructor Conflict Rate	$0 / 432 \times 100\%$	Terpenuhi
Room Conflict Rate	$0 / 312 \times 100\%$	Terpenuhi
Student Conflict Rate	$0 / 432 \times 100\%$	Terpenuhi

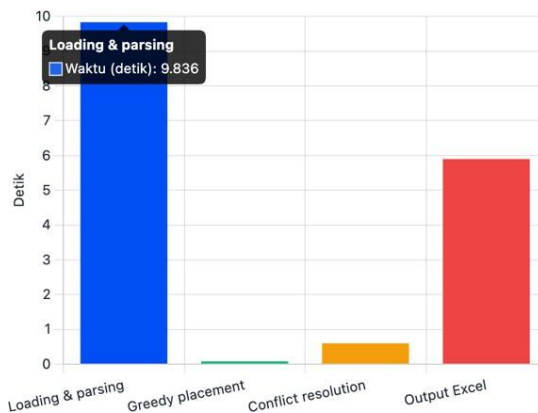
Jadwal akhir dapat dikategorikan sebagai *conflict-free schedule*. Hal ini membuktikan bahwa kombinasi *Greedy* dan CSP mampu menghasilkan jadwal yang tidak hanya cepat, tetapi juga bebas konflik.

6. Evaluasi Waktu Komputasi

Waktu komputasi diukur pada setiap tahap proses. Hasilnya menunjukkan bahwa proses algoritmik berjalan sangat cepat.

Tabel 5. Waktu Komputasi

Tahap	Waktu (detik)	Persentase
Loading dan parsing data	9.836	59.9%
Penempatan awal (Greedy)	0.081	0.5%
Iterative conflict resolution	0.598	3.6%
Output dan penulisan Excel	5.898	36.0%
Total	16.413	100%



Proses inti algoritma hanya membutuhkan 0,679 detik. Waktu terbesar terdapat pada proses input dan output file Excel. Dengan demikian, beban utama sistem tidak berada pada algoritma penjadwalan, tetapi pada proses pembacaan dan penulisan data.

7. Evaluasi Pemanfaatan Sumber Daya

Evaluasi *resource utilization* digunakan untuk menilai tingkat penggunaan ruang dan slot waktu.

Tabel 6. Pemanfaatan Sumber Daya

Metrik	Terpakai	Tersedia	Tingkat Pemanfaatan
Slot ruang (room-day-sesi)	309	560	55.18%
Slot waktu (day-sesi)	35	35	100.00%
Ruang unik	15	16	93.75%

Hasil ini menunjukkan bahwa seluruh slot waktu telah digunakan. Namun, pemanfaatan slot ruang masih sebesar 55,18%. Artinya, sistem masih memiliki kapasitas cadangan untuk menampung tambahan mata kuliah. Pemanfaatan ruang unik sebesar 93,75% menunjukkan bahwa hampir seluruh ruang tersedia telah digunakan.

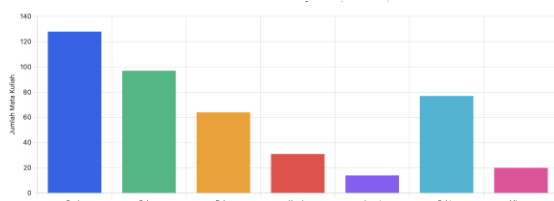
8. Distribusi Jadwal

Distribusi jadwal menunjukkan konsentrasi mata kuliah pada awal minggu dan sesi pagi. Hari Senin memiliki jumlah mata kuliah tertinggi, yaitu 128 MK atau 29,6%. Sesi pertama juga menjadi sesi paling banyak digunakan, yaitu 111 MK atau 25,7%.

Tabel 7. Distribusi Mata Kuliah per Hari

Hari	Jumlah MK	Persentase
Senin	128	29.6%
Selasa	97	22.5%
Rabu	64	14.8%
Kamis	31	7.2%
Jumat	14	3.2%
Sabtu	77	17.8%
Minggu	20	4.6%
Total	432	100%

Pola ini muncul karena *strategi Greedy first-fit* memprioritaskan slot yang tersedia paling awal. Akibatnya, jadwal cenderung terkonsentrasi pada hari Senin, Selasa, dan sesi pagi. Pola ini tidak melanggar *hard constraint*, tetapi menunjukkan bahwa pemerataan jadwal masih dapat ditingkatkan melalui mekanisme *load balancing*.



9. Perbandingan Metode

Perbandingan dilakukan terhadap tiga skenario, yaitu *Greedy* murni, CSP dengan *backtracking*, dan *hibrida Greedy + CSP*.

Tabel 8. Perbandingan Metode

Metrik	Greedy Murni	CSP + Backtracking	Hibrida Greedy
Total MK terjadwal	431	432	0
Konflik ruang	9	9	0
Konflik dosen	0	0	0
Total konflik	9	9	0
Bebas konflik?	Tidak	Tidak	Ya
Waktu komputasi (detik)	0.081	0.077	0.670

Metode hibrida menghasilkan performa terbaik dari sisi validitas jadwal. *Greedy* murni sangat cepat, tetapi masih menghasilkan konflik. CSP dengan *backtracking* juga belum mampu menghilangkan konflik pada dataset ini. Sebaliknya, *Greedy + CSP* mampu menghasilkan 432 jadwal bebas konflik, meskipun waktu komputasinya lebih tinggi.

D. KESIMPULAN

1. Sistem penjadwalan kuliah otomatis berbasis integrasi Algoritma Greedy dan Constraint Satisfaction Problem (CSP) berhasil dirancang dan diimplementasikan menggunakan bahasa pemrograman Python. Sistem ini mampu menghasilkan jadwal kuliah yang sepenuhnya bebas konflik (*conflict-free schedule*) dengan *Constraint Satisfaction Rate* sebesar 100% untuk seluruh 7 *hard constraint* yang telah ditentukan, meliputi tidak adanya bentrok dosen, bentrok ruang, bentrok mahasiswa, serta pemenuhan aturan khusus untuk semester awal (Zoom), mata kuliah *non-reguler*, MKDU, dan jadwal tetap PWK.
2. Efektivitas sistem dalam menangani kompleksitas *constraint* Fakultas Teknik terbukti melalui kemampuannya menjadwalkan 432 mata kuliah dari 6 program studi dengan waktu komputasi inti hanya 0.679 detik dan konvergensi dalam 2 iterasi. Hasil perbandingan dengan metode *Greedy* murni dan CSP *Backtracking* menunjukkan bahwa pendekatan hibrida merupakan satu-satunya metode yang berhasil menghasilkan jadwal tanpa konflik (0 konflik ruang dan 0 konflik dosen), sedangkan kedua metode pembanding masing-masing masih menyisakan 9 konflik ruang.

E. DAFTAR PUSTAKA

- [1] M. A. Andriansah, S. Praja, and T. Afandy, "Implementasi algoritma greedy dan string matching penukaran uang menjadi koin, pencarian pola dalam teks yang berisi nama 'Indomaret'," Jurnal Multimedia dan Teknologi Informasi (Jatilima), vol. 6, no. 2, pp. 123–133, 2024.

- [2] Y. Eri, "Penerapan algoritma genetika dan algoritma greedy dalam menentukan rute terpendek pendistribusian air mineral kemasan di Bandar Lampung," Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Lampung, 2025.
- [3] M. & B. Pramono, "Implementasi metode pewarnaan graf menggunakan algoritma greedy untuk menentukan jadwal mata pelajaran pada SMKN 4 Kendari," J. Informatika Ilmu Komputer dan Sist. Inf., vol. 1, no. 2, 2023.
- [4] N. W. Filkaromah, "Kalender akademik dalam manajemen strategi pendidikan Islam," J. Inov. dan Kolabor. Nusantara, vol. 6, no. 2, 2025.
- [5] A. Harfani, "Penerapan metode constraint satisfaction problem pada sistem pakar diagnosa penyakit shigellosis," J. Informatics Manag. Inf. Technol., vol. 2, no. 4, pp. 145–150, 2022.
- [6] A. Ihtiara and M. Abdul Muthalib, "Penjadwalan mata kuliah menggunakan algoritma greedy dengan mempertimbangkan preferensi dosen," J. Sains dan Teknol. 4.0, vol. 1, no. 2, pp. 37–46, 2024.
- [7] M. K. Lubis, Sistem pakar diagnosa penyakit skizofrenia dengan menerapkan kombinasi metode constraint satisfaction problem (CSP) dan metode certainty factor, 2023.
- [8] U. S. Mustakim, "Hubungan manajemen waktu perkuliahan dan efektivitas pembelajaran dengan ruang kelas terbatas," J. Pendidik. dan Penelit. Semesta Mendidik, vol. 2, no. 2, pp. 169–178, 2025.
- [9] V. Nurcahyawati, M. Risqiwahid, and A. Arrosyidi, "Optimasi penjadwalan mata pelajaran menggunakan constraint programming," J. Informatika dan Teknol. Inf. (JUTEK), vol. 1, no. 3, pp. 118–124, 2023.
- [10] L. A. Pangestu, S. H. Suryawan, and A. J. Latipah, "Penerapan algoritma genetika dalam penjadwalan mata pelajaran," J. Informatika, vol. 10, no. 2, pp. 194–205, 2023.
- [11] H. D. Pitaloka and R. A. N. Pamudji, "Implementasi forward chaining dan CSP untuk diagnosa penyakit lambung pada manusia," IKRAM: J. Ilmu Komputer Al Muslim, vol. 4, no. 2, pp. 70–77, 2025.
- [12] E. W. Pratiwi and M. Z. Siambaton, "Aplikasi penjadwalan dokter pada Rumah Sakit Umum Kota Pinang dengan menggunakan algoritma greedy," Hello World J. Ilmu Komputer, vol. 1, no. 1, pp. 1–9, 2022, doi: 10.56211/helloworld.v1i1.4.
- [13] I. A. H. Rahman, "Implementasi integer linear programming dalam sistem penjadwalan perkuliahan berbasis web pada Jurusan Ilmu Komputer Universitas Lampung (Skripsi)," Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Lampung, 2025.
- [14] I. A. H. Rahman, "Optimasi

- penjadwalan perkuliahan menggunakan pemrograman linear: Studi kasus Program Studi Informatika Universitas Baturaja,” 2026.
- [15] A. Roihan, K. Nasution, and M. Z. Siambaton, “Implementasi algoritma greedy kombinasi dengan perulangan pada aplikasi penjadwalan praktikum,” *Sudo J. Teknik Informatika*, vol. 1, no. 2, pp. 42–50, 2022, doi: 10.56211/sudo.v1i2.8.
- [16] M. A. Rohim, F. Wiranto, and D. A. Fauziah, “Optimasi pembuatan jadwal perkuliahan menggunakan algoritma genetika berbasis pendekatan multivariat,” *BIOS: J. Teknol. Inform. dan Rekayasa Komputer*, vol. 6, no. 1, pp. 31–38, 2025, doi: 10.37148/bios.v6i1.160.
- [17] R. M. Rohmawati, M. I. A. Fathoni, and I. Ismanto, “Penerapan algoritma Welch-Powell pada penyusunan jadwal perkuliahan di program studi pendidikan matematika,” *Euler: J. Ilmiah Matematika, Sains dan Teknologi*, vol. 10, no. 2, pp. 200–210, 2022.
- [18] S. Serliana, R. Rahman, H. Hastuti, F. Yusuf, and A. M. Abidin, “Penerapan algoritma Particle Swarm Optimization (PSO) pada sistem penjadwalan kegiatan mentoring baca tulis Al-Qur’an,” *Switch: J. Sains dan Teknol. Inf.*, vol. 3, no. 5, pp. 55–71, 2025, doi: 10.62951/switch.v3i1.337.
- [19] N. Siregar and H. Jumianto, “Penerapan algoritma backtracking dalam penyelesaian masalah,” *Jusinfo: J. Sains dan Informatika*, vol. 1, no. 1, pp. 1–7, 2025.
- [20] N. Sri Dewi and A. Hasan, “Perbandingan genetic algorithm dan ant colony optimization dalam optimasi penjadwalan perawat di rumah sakit,” *J. Nasional Teknol. Komputer*, vol. 6, no. 1, pp. 85–93, 2026, doi: 10.61306/jnastek.v6i1.351.
- [21] V. Suhandi, V. Arisandhy, and D. T. Liputra, “Penjadwalan mata kuliah dengan mempertimbangkan ketersediaan waktu pengajar dan satuan kredit semester yang tidak terpisah menggunakan integer linear programming,” *J. Integrated System*, vol. 6, no. 1, pp. 73–86, 2023.
- [22] A. F. Syuhada, R. Mardhiyyah, and F. I. Sanjaya, “Integrasi model algoritma genetika dan constraint satisfaction problem pada optimasi penjadwalan shift karyawan UMKM kuliner,” *Bull. Comput. Sci. Res.*, vol. 6, no. 1, pp. 458–468, 2025.
- [23] T. Triyono and K. Kusriani, “Algoritma genetika dalam penjadwalan mata kuliah: Eksplorasi metode crossover, mutasi, dan seleksi terbaik,” *Duta.com*, vol. 17, no. 2, pp. 126–134, 2024.
- [24] S. A. Windari, “Aplikasi penjadwalan dan laporan kegiatan patroli Sat Samapta berbasis website pada Polres Ogan Komering Ilir (OKI) (Diploma thesis),” *Politeknik Negeri Sriwijaya*, 2024.

- [25] [30] E. Yulia Sari, T. Rahmawati, and A. Priyanto, "Pengaruh hybrid adaptive repairing genetic algorithm dan VIKOR-based fitness dalam optimasi penjadwalan perkuliahan," *J. Media Informatika*, vol. 7, no. 1, pp. 16–25, 2026, doi: 10.55338/jumin.v7i1.7676.
- [26] R. G. Zahidah and R. A. Putri, "Implementasi algoritma constraint satisfaction problems dan backtracking pada penjadwalan kegiatan belajar mengajar," *METIK Jurnal*, vol. 9, no. 1, pp. 82–91, 2025.
- [27] Y. X. G. H. Zega and G. E. Kurniawati, "Pentingnya manajemen waktu bagi mahasiswa dalam meningkatkan prestasi belajar di Sekolah Tinggi Teologi Duta Panisal Jember," *Metanoia*, vol. 4, no. 1, pp. 58–70, 2022.