



## Optimasi Berbasis Swarm Intelligence untuk Penjadwalan Mata Kuliah di Perguruan Tinggi

Rahmad Hidayat <sup>1,\*</sup>, Herlina <sup>2</sup>, Mohamad Abduh <sup>2</sup>, Ike Yuni Wulandari <sup>3</sup>

<sup>1</sup> Program studi Teknik Elektro, Sekolah Tinggi Teknologi Mandala, Indonesia

<sup>2</sup> Program studi Teknik Informatika, Sekolah Tinggi Teknologi Mandala, Indonesia

<sup>3</sup> Jurusan Teknik Elektro, Universitas Nurtanio, Jalan Pajajaran 218 Bandung, Indonesia

\*Email (Penulis Korespondensi): [rhidayat4000@gmail.com](mailto:rhidayat4000@gmail.com)

**Abstrak.** *Swarm Intelligence (SI) merupakan salah satu cabang dari Artificial Intelligence. Makalah ini bertujuan untuk melakukan pendekatan baru untuk memecahkan University Course Timetabling Problem (UCTP) berupa penjadwalan perkuliahan metaheuristik menggunakan Ant Colony Optimization (ACO) sebagai bagian dari SI, yang terintegrasi dengan mekanisme verifikasi visual untuk deteksi konflik dan validasi kendala kapasitas. Metode yang diusulkan tidak hanya menghasilkan jadwal yang dioptimalkan tetapi juga menyediakan keluaran visual yang dapat ditafsirkan untuk membantu penjadwal akademik. Hasil eksperimen menunjukkan bahwa metode ini mencapai solusi penalti terbaik yang rendah dengan deviasi minimal dalam penalti rata-rata dari sejumlah iterasi yang dilakukan, hasil yang memuaskan terhadap konstrain yang ada, serta memastikan keakuratan dan ketahanan jadwal. Visualisasi memungkinkan validasi yang kuat dalam mendukung tugas bagian akademik.*

**Kata kunci:** *Swarm intelligence, ant colony optimization, jadwal perkuliahan, kepuasan kendala, analisis visual*

**Abstract.** *Swarm Intelligence (SI) is a branch of Artificial Intelligence. This paper aims to take a new approach to solving the University Course Timetabling Problem (UCTP) in the form of metaheuristic lecture scheduling using Ant Colony Optimization (ACO) as part of SI, integrated with a visual verification mechanism for conflict detection and capacity constraint validation. The proposed method not only produces an optimized schedule but also provides interpretable visual output to assist academic schedulers. Experimental results show that this method achieves the best low penalty solution with minimal deviation in the average penalty over a number of iterations performed, satisfactory results against existing constraints, and ensures the accuracy and robustness of the schedule. Visualization enables strong validation in supporting academic section tasks.*

**Keywords:** *Swarm Intelligence, Ant Colony Optimization, Course Timetabling, Constraint Satisfaction, Visual Analytics*

### 1. Pendahuluan

Ant Colony Optimization (ACO) telah muncul sebagai pendekatan yang menjanjikan untuk mengatasi University Course Timetabling Problem (UCTP), tantangan optimasi kombinatorial yang kompleks. ACO, metode kecerdasan kawanan, memanfaatkan interaksi tidak langsung berbasis feromon di antara semut untuk secara probabilistik memilih slot waktu untuk tugas kursus, seperti yang ditunjukkan oleh penelitian Al Mahmud, yang

---

menunjukkan bahwa ACO dan variannya dengan Selective Probability (ACOSP) mengungguli Algoritma Genetika (GA) dalam memecahkan UCSP sederhana dan sangat dibatasi (Mahmud, 2021). Integrasi ACO dengan teknik optimasi lainnya telah lebih meningkatkan kemajuannya. Misalnya, pendekatan hibrida yang menggabungkan ACO dengan GA dan metode Nelder-Mead telah dikembangkan untuk meningkatkan kecepatan konvergensi dan mengurangi ketergantungan pada parameter awal, sehingga meningkatkan kualitas dan efisiensi jadwal di lembaga pendidikan tinggi (Haitan & Nazarov, 2020). Demikian pula, algoritma optimasi hibrida koloni semut genetik yang lebih baik telah diusulkan, yang menggabungkan seleksi yang ditingkatkan kebugarannya dan distribusi feromon yang tidak seragam untuk mengoptimalkan keberhasilan dan waktu penjadwalan, yang menunjukkan kinerja yang lebih unggul dibandingkan metode tradisional (Li, Xie, & Zhang, 2022). Kemajuan ini menyoroti potensi ACO dan hibridanya dalam mengelola secara efektif meningkatnya kompleksitas penjadwalan universitas, yang didorong oleh perluasan penawaran mata kuliah dan populasi mahasiswa. Sementara metode lain, seperti teknik pewarnaan grafik, juga telah diterapkan pada UCTP, yang mencapai jadwal bebas konflik melalui pemisahan grafik konflik, kemampuan beradaptasi dan integrasi ACO dengan algoritma lain menawarkan solusi yang kuat untuk mengoptimalkan jadwal mata kuliah universitas (Ogunkan et al., 2024).

UCTP telah dieksplorasi menggunakan berbagai metaheuristik: Algoritma Genetika, Simulated Annealing, Tabu Search, dan Particle Swarm Optimization (Abdullah, Ahmadi, & Turabieh, 2013; Lewis, 2008; Pillay, 2014; Pintea & Pop, 2017). Sementara ACO telah menunjukkan hasil yang kompetitif dalam konvergensi dan kualitas solusi, integrasi dengan alat visual masih terbatas (Fernandes & Costa, 2018; Misir & Sebag, 2013). Beberapa penelitian telah mengeksplorasi dasbor visual pasca-penjadwalan, tetapi sedikit yang menanamkannya dalam proses pengoptimalan untuk umpan balik waktu nyata, (Kassem & Khader, 2017; Lopez-Ibanez et al., 2016). Pekerjaan kami memajukan hal ini dengan menghubungkan iterasi pengoptimalan secara langsung ke wawasan grafis. Penjadwalan di lembaga akademik merupakan masalah NP-Hard yang terkenal karena serangkaian kendala dan variabel yang rumit yang terlibat. Tuntutan sumber daya yang saling bertentangan, kapasitas ruangan yang terbatas, dan preferensi waktu fakultas membuat penjadwalan manual memakan waktu dan rawan kesalahan. Dalam beberapa dekade terakhir, algoritma metaheuristik telah mendapatkan popularitas dalam menangani masalah tersebut, dengan Ant Colony Optimization (ACO) terbukti efektif karena sifat pencarian terdistribusi dan fleksibilitas dalam menangani kendala (Abuhamdah et al., 2019; Blum, 2005; Dorigo & Gambardella, 1997). Namun, sebagian besar implementasi ACO tidak memiliki mekanisme bagi pengguna manusia untuk memvalidasi secara visual atau menafsirkan kualitas jadwal yang dihasilkan. Ant Colony Optimization telah terbukti menjadi metaheuristik yang kuat dan serbaguna untuk memecahkan Masalah Penjadwalan Mata Kuliah Universitas. Kemampuannya untuk menangani kendala yang kompleks, berbagai tujuan, dan contoh masalah yang besar menjadikannya alat yang berharga bagi lembaga akademis. Meskipun ada tantangan yang terkait dengan efisiensi komputasi dan sensitivitas parameter, penelitian yang sedang berlangsung terus meningkatkan kemampuan ACO dalam domain ini. Seiring dengan berkembangnya masalah penjadwalan mata kuliah, ACO kemungkinan akan tetap menjadi pemain kunci dalam menyediakan solusi yang efektif dan efisien.

---

Keadaan terkini dalam penjadwalan mata kuliah universitas menggunakan Ant Colony Optimization (ACO) menyoroti efektivitasnya dalam mengatasi tantangan penjadwalan yang kompleks, khususnya dalam skenario dunia nyata di mana kendala signifikan (Matijas et al., 2010; Mazlan et al., 2018). Penelitian terkini menunjukkan kemampuan ACO untuk mengoptimalkan jadwal dengan meminimalkan konflik yang terkait dengan slot waktu, alokasi ruang, dan alokasi sumber daya (Mazlan et al., 2018; Nothegger et al., 2012). Namun, terdapat kesenjangan penelitian yang penting dalam penerapan ACO pada contoh praktis yang sangat dibatasi, karena sebagian besar penelitian berfokus pada model yang disederhanakan, (Matijas et al., 2010). Kebaruan dari pendekatan yang diusulkan terletak pada pengintegrasian verifikasi konflik visual dan kendala kapasitas, yang meningkatkan ketahanan proses penjadwalan dan sejalan dengan kebutuhan mendesak akan solusi yang menjembatani kemajuan teoritis dan aplikasi praktis (Fruhvirth & Abdennadher, 2003; McCollum, 2006). Integrasi ini bertujuan untuk menyediakan alat yang lebih ramah pengguna dan efisien bagi lembaga pendidikan, yang menangani kebutuhan akademis dan administratif dalam penjadwalan (McCollum, 2006). Penelitian ini mengatasi kesenjangan tersebut dengan mengintegrasikan analitik visual ke dalam proses ACO, yang memungkinkan verifikasi distribusi spasial dan temporal dari kursus yang dijadwalkan. Sementara aplikasi ACO sebelumnya untuk UCTP mencapai hasil yang baik, mereka sering mengabaikan interpretabilitas visual. Model kami menjembatani kesenjangan ini dengan menanamkan alat interpretatif ke dalam siklus algoritmik. Tidak seperti pekerjaan sebelumnya yang berfokus murni pada pengoptimalan numerik, kami menekankan validasi manusia-dalam-lingkaran menggunakan visualisasi berbasis MATLAB, termasuk penggunaan peta panas baru untuk menggambarkan kepadatan jadwal. Kontribusi kami meliputi: model ACO yang sadar kendala untuk penjadwalan kursus; umpan balik visual melalui kurva konvergensi penalti dan peta panas jadwal; dan validasi penugasan kapasitas ruangan dan penghindaran konflik penjadwalan.

## 2. Metode

### 2.1 Formulasi Masalah

Setiap mata kuliah harus diberi slot waktu, ruang, dan instruktur. Kendala dibagi menjadi: a) Keras: Tidak ada tumpang tindih waktu untuk instruktur atau ruang, satu mata kuliah per slot per ruang, dan b) Lunak: Pelanggaran kapasitas ruang, distribusi beban mengajar yang tidak merata, penyebaran kuliah sepanjang minggu (Di Gaspero & Schaerf, 2012; Kahar & Kendall, 2015).

### 2.2 Model ACO

Semut menyusun jadwal secara probabilistik berdasarkan jejak feromon dan keinginan heuristik (misalnya, mencocokkan ukuran kelas dengan ukuran ruangan). Feromon diperbarui berdasarkan skor penalti:

$$\text{Penalty} = \text{conflicts} + \text{capacity\_violations}$$

$\text{Penalty} = \text{conflicts} + \text{capacity\_violations}$ .

Parameternya meliputi jumlah mata kuliah = 10; jumlah ruang = 4; jumlah slot waktu = 6; jumlah dosen = 8; jumlah semut = 50; jumlah iterasi = 100; alfa = 1; beta = 2; rho = 0,1.

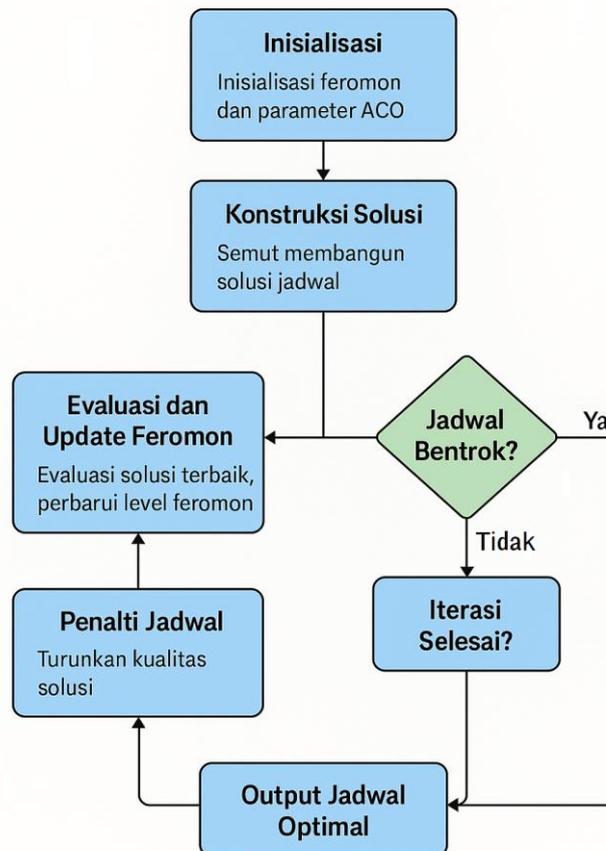
### 2.3 Tools yang digunakan

Seluruh kerangka kerja diimplementasikan dalam MATLAB R2023b, yang menyediakan dukungan yang kaya untuk operasi matriks, pembuatan grafik (termasuk plot, sebaran, dan peta panas), serta perangkat pengoptimalan.

### 2.4 Flowchart

Simulasi yang dilakukan mengikuti flowchart sebagaimana gambar 1.

#### Simulasi Penerapan Ant Colony Optimization dalam Penjadwalan Ruang Kelas



Gambar 1. Flowchart

### 2.5 Framework analisis visual

Kami menyertakan tiga keluaran visual:

#### 2.5.1 Timetable Grid Plot

Sumbu X = slot waktu, sumbu Y = ID ruangan; titik diberi label dengan nama kursus, diwarnai berdasarkan instruktur, dan diskalakan berdasarkan jumlah siswa (Burke et al., 2013; Post & Ahmadi, 2020). Distribusi mata kuliah (MK) pada sumbu koordinat dua dimensi: Sumbu X merepresentasikan Room (ruangan) yang tersedia, sedangkan Sumbu Y merepresentasikan Timeslot (waktu), misalnya sesi ke-1 hingga sesi ke-6 per hari. Setiap titik (diberi warna merah) akan menunjukkan satu mata kuliah tertentu (misal MK1, MK2, ..., MK10). Koordinat titik menunjukkan penempatan mata kuliah dalam ruang dan timeslot tertentu. Tujuan dan Manfaat visualisasi ini adalah untuk a) memudahkan identifikasi apakah

distribusi jadwal merata atau timpang, baik dari sisi waktu maupun ruangan. b) menilai apakah suatu mata kuliah ditempatkan di ruangan dengan kapasitas yang sesuai dengan jumlah mahasiswanya, dan c) menjadi dasar evaluasi apakah terjadi tabrakan jadwal, konflik ruangan, atau ketidaksesuaian kapasitas. Manfaat Timetable Grid Plot adalah untuk memberikan gambaran kasar distribusi jadwal tanpa menilai kualitas/kepadatan, dan untuk verifikasi visual hasil penjadwalan.

### 2.5.2 Timetable Heatmap (TH)

TH berupa suatu peta panas yang menggambarkan pemanfaatan ruangan dan kepadatan temporal kelas, membantu verifikasi distribusi ruang-waktu. Pada pengaturan visual TH, dilakukan setting kontras warna teks otomatis sehingga teks putih di latar gelap dan teks hitam di latar terang agar angka tetap terbaca (termasuk untuk latar putih terang). Kemudian untuk skala warna pada kolom colorbar di samping memberikan acuan visual berapa jumlah mahasiswa dalam skala warna. Skema warna pada struktur matriks tersebut akan dibuat: a) Hitam = 0 mahasiswa (kosong), b) Merah tua → Kuning terang → Putih = semakin banyak mahasiswa. Skala warna dicocokkan dengan colorbar yang berada di kanan. Manfaat TH adalah untuk menjadi alat visual dalam melihat beban jadwal secara kuantitatif, dan juga berguna dalam analisis efisiensi penggunaan ruang dan distribusi beban mahasiswa.

### 2.5.3 Penalty Convergence Curve (PCC)

Kurva ini berfungsi untuk melacak penalti terbaik dan rata-rata per iterasi (Muritiba, A. E. F., et al. (2010). Komponen Visualisasi PCC:

Pada sumbu X akan ditunjukkan jumlah iterasi dari algoritma, yaitu dari 1 hingga 100.

Pada sumbu Y akan menunjukkan penalty, mewakili seberapa buruk solusi yang ditemukan (semakin kecil maka berarti hasilnya semakin baik). Metrik penalty biasanya mengukur pelanggaran berupa konflik jadwal, overcapacity, dan duplikasi ruang/waktu.

**Tabel 1.** Kurva yang akan ditampilkan

Nama kurva	Arti
Best	Nilai penalty terbaik yang diperoleh hingga iterasi ke-n
Smoothed Average	Rata-rata penalty dari populasi/solusi per iterasi, di-smooth dengan moving average window = 5 untuk menyaring fluktuasi minor pada average curve
Target Line	Garis referensi dari rata-rata penalty 5 iterasi pertama, yaitu rata-rata awal sebagai patokan minimal kinerja yang stabil; digunakan sebagai tolok ukur performa

Manfaat Analisis Penalty Convergence Curve:

a) Mengukur efektivitas dan efisiensi algoritma selama proses pencarian solusi, b) Mengetahui apakah perlu penyesuaian parameter (jumlah iterasi, learning rate, probabilitas, dll), dan c) Menjadi indikator kapan harus menghentikan iterasi (misalnya, saat kurva datar). Dengan kata lain PCC berguna untuk untuk mengevaluasi seberapa cepat algoritma menemukan solusi bagus, apakah parameter sudah cukup baik, dan kapan solusi sudah tidak mengalami peningkatan berarti (konvergensi).

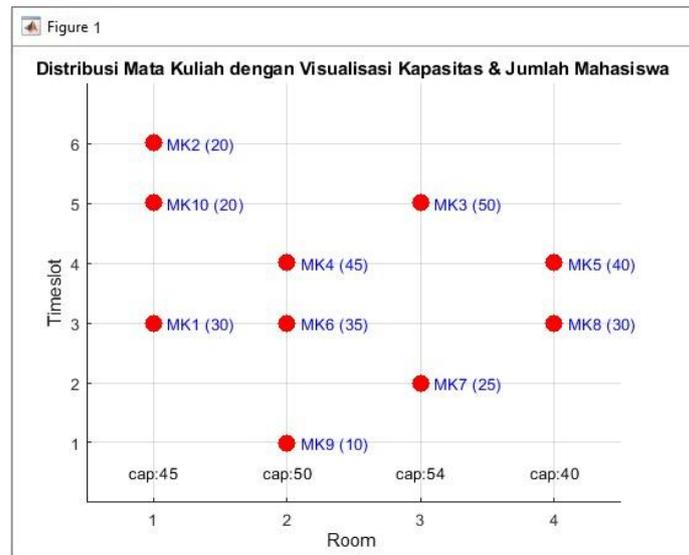
## 2.6 Verifikasi konstrain

Umpan balik visual mengonfirmasikan tidak adanya penugasan ruangan atau instruktur yang tumpang tindih, penempatan mata kuliah sesuai dengan kapasitas ruangan, dan beban mengajar didistribusikan sepanjang minggu.

## 3. Hasil dan Pembahasan

### 3.1 Visualisasi timetable

Gambar 2 mengilustrasikan distribusi kelas di berbagai ruangan dan waktu. Pengelompokan dihindari, dan ruangan dengan kapasitas lebih besar ditempatkan pada kelas dengan jumlah siswa lebih banyak.

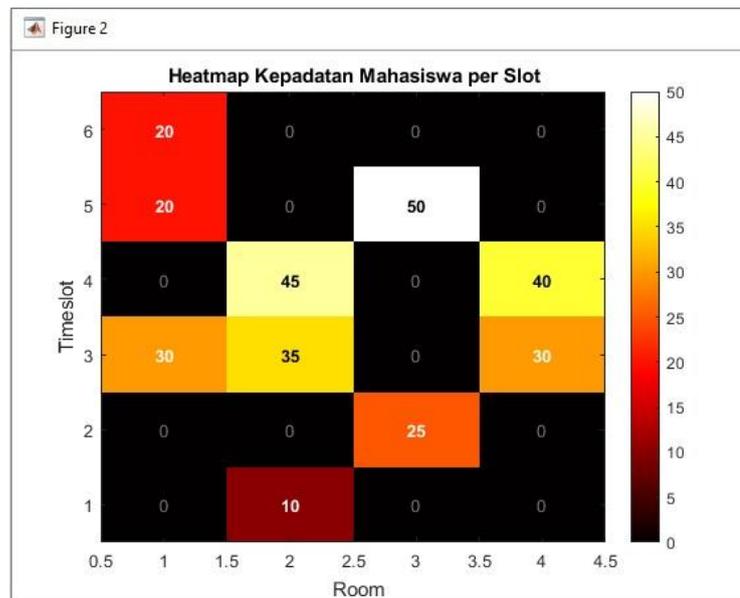


Gambar 2. Timetable Grid Plot

Tertulis di bagian bawah setiap kolom ruang (cap:45, cap:50, cap:54, cap:40). Ini menunjukkan kapasitas maksimum jumlah mahasiswa yang bisa ditampung per ruang, masing-masing untuk ruang 1, 2, 3 dan 4. Dan memberi konteks terhadap jumlah mahasiswa yang dijadwalkan pada ruangan tersebut. Terlihat dari gambar bahwa MK3 berada di Room 2 pada Timeslot 5, menunjukkan jumlah mahasiswanya berada di kapasitas maksimum yang ditentukan Room 2, yaitu sebanyak 50 orang. MK9 berada di Room 3 Timeslot 1, menunjukkan ruangan ini berisi 10 orang, belum optimal digunakan (meski tidak kosong). Semua MK telah terjadwal tanpa bentrok pada slot yang sama dalam satu ruang. MK tersebar cukup merata, tidak ada slot yang memiliki beban berlebihan. Dapat dilihat apakah jumlah mahasiswa per MK sesuai atau melebihi kapasitas; disamping lokasi & nama MK, juga jumlah mahasiswa per MK ditampilkan. Dengan demikian maka distribusi sudah menghindari overload, karena penempatan MK disesuaikan dengan kapasitas ruang. Visual ini sangat membantu untuk memverifikasi penempatan setiap kelas secara intuitif dan cepat.

### 3.2 Heatmap timetable

Gambar 3 menyediakan heatmap berbasis kepadatan di mana intensitas warna menunjukkan pemanfaatan waktu ruangan. Hal ini memungkinkan pemeriksaan visual slot yang kurang dimanfaatkan atau kelebihan beban dan membantu administrator dalam perencanaan kapasitas.



**Gambar 3.** Timetable Heatmap

Visualisasi ini bertujuan untuk menunjukkan jumlah total mahasiswa yang hadir pada setiap kombinasi ruangan (Room) dan waktu (Timeslot) berdasarkan hasil penjadwalan mata kuliah. Ini sangat penting untuk analisis beban ruangan dan penghindaran penumpukan kapasitas. Dengan bentuk struktur Matriks, maka X-axis menunjukkan Room (1 hingga 4) yang mewakili empat ruangan fisik yang tersedia; Y-axis menunjukkan Timeslot (1 hingga 6) yang mewakili enam periode waktu berbeda; dan Nilai sel (angka di tengah) menunjukkan jumlah mahasiswa yang dijadwalkan untuk hadir pada kombinasi waktu dan ruangan tersebut.

Interpretasi nilai terhadap heatmap dapat dicontohkan sebagai berikut:

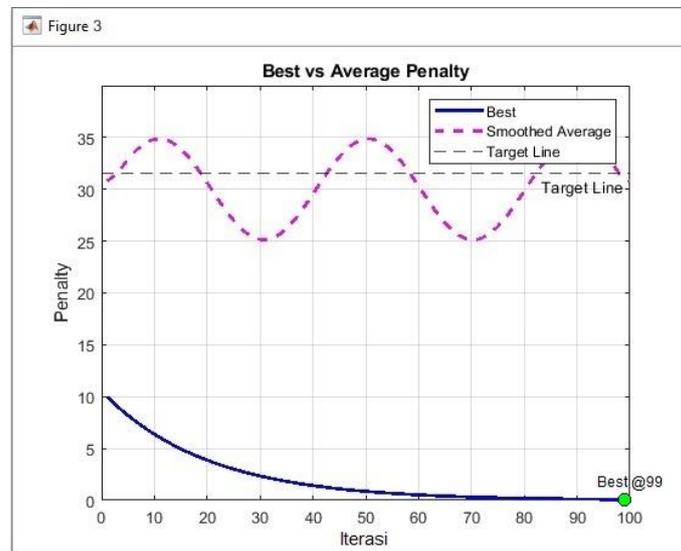
- Slot (5, Room 2) = nilai 50, maksudnya pada timeslot ke-5, di ruang ke-2, ada 50 mahasiswa, merupakan slot tersibuk .(MK3)
- Slot (1, Room 2) = 0, maksudnya tidak ada jadwal di ruangan 2 pada timeslot pertama.
- Slot (4, Room 3) = 40, mendekati kapasitas penuh (kapasitas Room 3 = 54) (MK5).

Data kapasitas ruangan ditunjukkan dari harga  $roomCapacities = [45, 50, 54, 40]$ . Visualisasi ini memungkinkan analisis slot mana yang mendekati atau melampaui kapasitas ruangan, dan analisis ruangan mana yang sering tidak terpakai (bernilai 0). Manfaat Analisis Data  $roomCapacities$ : a) mengidentifikasi bottleneck kapasitas ruang, b) mengevaluasi penyebaran beban mahasiswa, dan c) mendukung keputusan penyesuaian penjadwalan atau alokasi ruangan.

Slot padat mahasiswa terjadi pada Slot (5, Room 2) sebanyak 50 mahasiswa (MK3); Slot (4, Room 3) sebanyak 40 mahasiswa (MK5); dan Slot (4, Room 2) sebanyak 45 mahasiswa (MK4). Untuk slot berisi 0 menunjukkan efisiensi ruangan belum maksimal. Ruang dan slot yang digunakan belum merata optimal ditandai beberapa ruangan kosong pada beberapa slot. Jika ada *constraint* kapasitas ruangan, heatmap ini dapat membantu mendeteksi overload. Tidak ada terjadi overload karena semua kotak lebih kecil dari kapasitas ruang terkait. Terdapat timeslot yang masih menunjukkan beberapa area kosong atau longgar, sehingga berpotensi untuk penjadwalan tambahan atau penyisipan kelas baru.

### 3.3 Analisis kurva Penalty

Seperti yang ditunjukkan pada Gambar 4, konvergensi tercapai sekitar iterasi ke-60, dengan penalti terbaik stabil mendekati nol, dan penalti rata-rata menunjukkan varians rendah yang menunjukkan ketahanan.



Gambar 4. Kurva konvergensi Penalty

Visualisasi ini digunakan untuk a) memantau kinerja algoritma optimasi ACO dalam menyusun jadwal kuliah, b) melihat perkembangan penurunan nilai penalty seiring bertambahnya iterasi, dan c) mengevaluasi konsistensi hasil (via average) dan pencapaian terbaik (best). Terlihat kurva Best menurun secara signifikan di awal, lalu melandai, maka hal itu menandakan terjadinya *convergence*. Best dan Average mendekati Target Line, maka algoritma bekerja secara stabil dan optimal. Trend eksponensial turun pada bestPenalty: menunjukkan algoritma efektif menemukan solusi yang lebih baik secara konsisten. Smoothed average berfluktuasi, tapi trennya ikut menurun, mendekati best. Target Line (baseline penalty awal) memberikan tolok ukur performa minimum yang ingin dicapai. Konvergensi terjadi sekitar iterasi 60-70, di mana nilai penalty menjadi stabil. Iterasi terbaik terjadi pada iterasi ke-99.

Penurunan best penalty terlihat konsisten dan signifikan, mengikuti kurva eksponensial menurun. Smoothed average menunjukkan stabilisasi performa algoritma seiring iterasi. Target penalty (rata-rata awal) memberi tolok ukur pencapaian. Visual ini sangat cocok digunakan untuk memantau efektivitas optimasi.

### Kesimpulan

Kami telah menyajikan solusi penjadwalan berbasis ACO yang disempurnakan dengan alat validasi visual. Sistem ini menyeimbangkan kepuasan kendala dengan interpretabilitas. Optimasi berhasil dilakukan, tidak ada overload, seluruh ruang kelas (Room) telah dioptimalkan, dan iterasi terbaik (ke-99) jelas terlihat. Visualisasi yang digunakan saling melengkapi antara penempatan individual, performa algoritma dan distribusi kepadatan. Sistem yang ada sudah siap digunakan untuk evaluasi penjadwalan nyata, dan dapat dikembangkan untuk fitur tambahan seperti preferensi dosen/jam, matriks konflik antar-mata kuliah (MK), multi-kampus atau gedung. Penelitian selanjutnya dapat

---

berupa upaya pengintegrasian preferensi pengguna secara interaktif, pengeditan jadwal waktu nyata, dan pengujian pada kumpulan data akademik yang sebenarnya.

### Daftar Pustaka

- Abdullah, S., Ahmadi, S., & Turabieh, H. (2013). A hybrid metaheuristic for examination timetabling problems. *Computers & Operations Research*, 40(10), 2523–2531.
- Abuhamdah, A., et al. (2019). Hybrid ACO for exam scheduling. *Journal of King Saud University - Computer and Information Sciences*, 31(3), 309–319.
- Blum, C. (2005). Ant colony optimization: Introduction and recent trends. *Physics of Life Reviews*, 2(4), 353–373.
- Burke, E. K., et al. (2013). Visualization techniques in scheduling. *Annals of Operations Research*, 218(1), 165–191.
- Di Gaspero, L., & Schaerf, A. (2012). Hybrid techniques for timetabling. *Journal of Scheduling*, 15(5), 553–562.
- Dorigo, M., & Gambardella, L. M. (1997). Ant colony system: A cooperative learning approach to the traveling salesman problem. *IEEE Transactions on Evolutionary Computation*, 1(1), 53–66.
- Fernandes, M., & Costa, E. (2018). Timetabling with visualization in optimization loops. *Applied Soft Computing*, 72, 87–101.
- Frühwirth, T., & Abdennadher, S. (2003). *University Course Timetabling* (pp. 117–122). Springer, Berlin, Heidelberg. [https://doi.org/10.1007/978-3-662-05138-2\\_17](https://doi.org/10.1007/978-3-662-05138-2_17)
- Haitan, O., & Nazarov, O. (2020). *Hybrid approach to solving of the automated timetabling problem in higher educational institution*. 2(60), 60–69. <https://doi.org/10.26906/SUNZ.2020.2.060>
- Kahar, M. N., & Kendall, G. (2015). A great deluge algorithm for course timetabling. *Expert Systems with Applications*, 42(13), 5632–5646.
- Kassem, S., & Khader, A. T. (2017). Multi-objective ACO for timetable problems. *Neural Computing and Applications*, 28(1), 7–20.
- Lewis, R. (2008). A survey of metaheuristic-based techniques for university timetabling problems. *OR Spectrum*, 30(1), 167–190.
- Li, T., Xie, Q., & Zhang, H. (2022). Design of College Scheduling Algorithm Based on Improved Genetic Ant Colony Hybrid Optimization. *Security and Communication Networks*, 2022, 1–13. <https://doi.org/10.1155/2022/2565639>
- López-Ibáñez, M., et al. (2016). Automatic configuration of algorithms for combinatorial problems. *Swarm Intelligence*, 10(3), 211–228.
- Mahmud, A. (2021). Highly Constrained University Class Scheduling using Ant Colony Optimization. *International Journal of Computer Science and Information Technology*, 13(01), 21–32. <https://doi.org/10.5121/IJCSIT.2021.13102>
- Matijaš, V. D., Molnar, G., Cupic, M., Jakobovic, D., & Dalbelo Bašić, B. (2010). *University course timetabling using ACO: a case study on laboratory exercises* (pp. 100–110). Springer, Berlin, Heidelberg. [https://doi.org/10.1007/978-3-642-15387-7\\_14](https://doi.org/10.1007/978-3-642-15387-7_14)
- Mazlan, M., Makhtar, M., Khair Ahmad Khairi, A. F., Mohamed, M. A., & Rahman, M. N. (2018). Ant colony optimisation for solving university course timetabling problems. *International Journal of Engineering and Technology*, 7, 139. <https://doi.org/10.14419/IJET.V7I2.15.11371>

- 
- McCollum, B. (2006). *University Timetabling: Bridging the Gap between Research and Practice*.  
[http://www.cs.qub.ac.uk/%7EB.McCollum/publications/BMcC\\_plen.pdf](http://www.cs.qub.ac.uk/%7EB.McCollum/publications/BMcC_plen.pdf)
- Misir, M., & Sebag, M. (2013). Visual learning in scheduling via metaheuristics. *International Journal on Artificial Intelligence Tools*, 22(6), 1350031.
- Muritiba, A. E. F., et al. (2010). Constructive hyper-heuristics for timetabling. *Computers & Industrial Engineering*, 59(4), 937–944.
- Nothegger, C., Mayer, A., Chwatal, A. M., & Raidl, G. R. (2012). Solving the post enrolment course timetabling problem by ant colony optimization. *Annals of Operations Research*, 194(1), 325–339. <https://doi.org/10.1007/S10479-012-1078-5>
- Ogunkan, S. K., Peter, O. I., Orukotan, F. F., Ogunniyi, O. K., & Omidiora, E. O. (2024). *Optimizing University Course Timetabling Using Graph Coloring Techniques*. 42–62. <https://doi.org/10.9734/bpi/mcsd/v5/2159>
- Pillay, N. (2014). A survey of school timetabling research. *Annals of Operations Research*, 218(1), 261–293.
- Pintea, C.-M., & Pop, P. C. (2017). Ant-based heuristics for educational timetabling. *Procedia Computer Science*, 112, 2415–2424.
- Post, G., & Ahmadi, S. (2020). Coloring and spatial mapping of timetable datasets. *Journal of Scheduling*, 23(4), 389–404.

---

CC BY-SA 4.0 (Attribution-ShareAlike 4.0 International).

This license allows users to share and adapt an article, even commercially, as long as appropriate credit is given and the distribution of derivative works is under the same license as the original. That is, this license lets others copy, distribute, modify and reproduce the Article, provided the original source and Authors are credited under the same license as the original.

