

Penjadwalan Mata Kuliah Otomatis Menggunakan Algoritma Late Acceptance *Hill-Climbing Hyper-Heuristics* dengan Domain Permasalahan ITC

Automated Course Timetabling Using Late Acceptance Hill Climbing Hyper-Heuristics Algorithm with Problem Domain from International Timetabling Competition

Cut Alna Fadhillah

Pogram Studi Informatika, Fakultas Sains dan Teknologi, Universitas Samudra

E-mail: cutalnafadhillah@unsam.ac.id

Abstrak

Permasalahan penjadwalan mata kuliah merupakan topik yang sangat menarik untuk diselesaikan dikarenakan termasuk salah satu permasalahan NP-hard, dimana belum ada algoritma konvensional eksak yang mampu menyelesaikannya dalam waktu polinomial. ITC adalah kompetisi yang diadakan khusus untuk permasalahan penjadwalan mata kuliah. Kompetisi yang sudah diadakan keempat kalinya ini terus memberikan tantangan yang berbeda dalam penyelesaian masalah untuk mendapatkan solusinya. Tujuan permasalahan penjadwalan mata kuliah pada International Timetabling Competition adalah untuk meminimalkan biaya yang dikeluarkan pada semua konten permasalahan. Terdapat dua permasalahan mata kuliah secara umum yang akan diselesaikan. Pertama adalah menjadwalkan mata kuliah pada waktu dan ruang yang telah disediakan dan kedua adalah membagi mahasiswa kepada kelas-kelas yang telah terjadwal. Selain itu terdapat pula batasan yang harus dipenuhi akan menjadi tantangan dalam menyelesaikan masalah ini. Beberapa penelitian untuk melakukan penyelesaian permasalahan penjadwalan mata kuliah telah dilakukan dengan menggunakan berbagai macam algoritma. Terdapat beberapa algoritma yang dapat digunakan untuk menyelesaikan permasalahan penjadwalan mata kuliah, salah satunya adalah algoritma Late acceptance hill climbing dengan menggunakan pendekatan Hyper-heuristics. Algoritma ini yang akan dipilih untuk menyelesaikan permasalahan penjadwalan mata kuliah menggunakan dataset dari International Timetabling Competition. Hasil luaran yang diharapkan dari pengerjaan penelitian ini adalah daftar jadwal mata kuliah dan daftar mahasiswa yang mengambil mata kuliah tersebut dengan memenuhi batasan-batasan yang telah ditetapkan sehingga hasil dari luaran tersebut dapat menjadi solusi untuk penyelesaian permasalahan penjadwalan mata kuliah dari domain permasalahan International Timetabling Competition yang kompetitif dengan hasil dari algoritma benchmark.

Kata kunci: penjadwalan mata kuliah, international timetabling competition, algoritma late acceptance hill climbing, hyper-heuristics

Abstract

The problem of scheduling courses is a very interesting topic to solve because it is one of the NP-hard problems, where there is no exact conventional algorithm that can solve it in polynomial time. ITC is a competition held specifically for the problem of scheduling courses. The competition, which has been held for the fourth time, continues to provide different challenges in solving problems to get solutions. The purpose of the problem of scheduling courses in the International Timetabling Competition is to minimize the costs incurred on all problem content. There are two general course problems that will be solved. The first is to schedule courses at the time and space that have been provided and the second is to divide students into classes that have been scheduled. In addition, there are also limitations that must be met which will be a challenge in solving this problem. Several studies to solve the problem

of scheduling courses have been carried out using various algorithms. There are several algorithms that can be used to solve the problem of scheduling courses, one of which is the Late acceptance hill climbing algorithm using the Hyper-heuristics approach. This algorithm will be selected to solve the problem of scheduling courses using the dataset from the International Timetabling Competition. The expected output from this research is a list of course schedules and a list of students taking the courses by fulfilling the established limitations so that the results of the output can be a solution to solving the problem of scheduling courses from the competitive International Timetabling Competition problem domain with the results of the benchmark algorithm.

Keywords: course scheduling, international timetabling competition, late acceptance hill climbing algorithm, hyper-heuristics

I. PENDAHULUAN

Penjadwalan mata kuliah adalah salah satu permasalahan *NP-hard*, yang berarti sulit untuk diselesaikan menggunakan metode yang konvensional dan waktu komputasi yang dibutuhkan untuk mendapatkan solusi yang optimal dengan meningkatnya eksponensial seiring besarnya permasalahan [1]. Dengan kata lain, berdasarkan sudut pandang optimasi kombinatorial, penjadwalan merupakan permasalahan *NP-hard*, yang artinya belum ada algoritma konvensional eksak yang dapat menyelesaikan permasalahan tersebut dalam waktu polinomial [2]. Hal tersebut yang menjadi alasan penjadwalan menarik untuk menjadi topik penelitian.

Permasalahan penjadwalan yang disediakan oleh *International Timetabling Competition 2019* adalah kompetisi keempat kalinya yang diselenggarakan. Permasalahan penjadwalan yang disediakan adalah kategori penjadwalan mata kuliah pada perguruan tinggi. Mata kuliah harus menemukan waktu dan ruang yang tepat dengan mempertimbangkan kondisi *student* dan *teacher* untuk masing-masing mata kuliah tersebut. Tujuan utamanya adalah meminimalkan biaya yang harus dikeluarkan pada semua *resource* yang tersedia serta memenuhi batasan yang telah ditentukan [3]. Selain itu hasil akhir juga harus meminimalkan adanya tumpang tindih dan kejanggalan yang terjadi pada *resources*. Dengan kata lain solusi yang dihasilkan harus *feasible* dengan memenuhi *hard constraint* dan memaksimalkan untuk memenuhi *soft constraint* [4].

Berdasarkan beberapa penelitian yang telah dilakukan sebelumnya, maka *framework Hyper-heuristics* dan algoritma *Late acceptance hill climbing* akan digunakan untuk menyelesaikan permasalahan penjadwalan mata kuliah [5]. *Hyper-Heuristic* memiliki keunggulan yaitu tidak diperlukannya parameter *tuning* dan tidak bergantung pada *problem domain*. Sedangkan LAHC dipilih karena merupakan algoritma *Hyper-Heuristic* baru yang terbukti memiliki performa lebih baik dibandingkan beberapa algoritma *meta-Heuristics* [6]. Pemilihan algoritma tersebut dikarenakan LAHC berhasil menyelesaikan permasalahan *exam timetabling* yang dilakukan sebagai uji coba dan menempati posisi kedua sebagai pemenang pada *ITC 2007* [7].

Penjadwalan adalah suatu kegiatan atau aktivitas pengalokasian, sumber daya dalam suatu ruang dan waktu, berdasarkan aturan dan batasan tertentu untuk mencapai tujuan optimal [8]. Penjadwalan mata kuliah adalah kegiatan pengalokasian kumpulan mata kuliah terhadap ruang dan jangka waktu tertentu,

pada saat yang sama, siswa dan guru juga ditugaskan ke mata kuliah sehingga pertemuan dapat berlangsung [9].

International Timetabling Competition 2019 Dataset yang selanjutnya akan disebut ITC 2019 Dataset adalah kumpulan data yang dikeluarkan oleh Timetabling Competition yang diselenggarakan keempat kalinya [10]. *International Timetabling Competition* pertama diselenggarakan pada tahun 2002, kedua pada tahun 2007, ketiga pada tahun 2011 dan keempat pada tahun 2019. Dataset ini berisi tentang ITC 2019 Dataset yang terbagi menjadi tiga grup data Instances yang akan dikeluarkan yaitu Early, middle, dan late Instances. Pada pengerjaan penelitian ini Dataset yang digunakan adalah Early Dataset yang memiliki 10 Instances. Jumlah course adalah total jumlah mata kuliah yang harus tersedia dan diambil mahasiswa, jumlah *class* adalah total jumlah kelas yang harus dijadwalkan pada waktu dan ruang yang telah tersedia [11].

2. METODOLOGI PENELITIAN

Metodologi dari penelitian yang dapat dilihat pada Gambar 1 yang terdiri dari langkah-langkah sebagai berikut.

A. Identifikasi Permasalahan

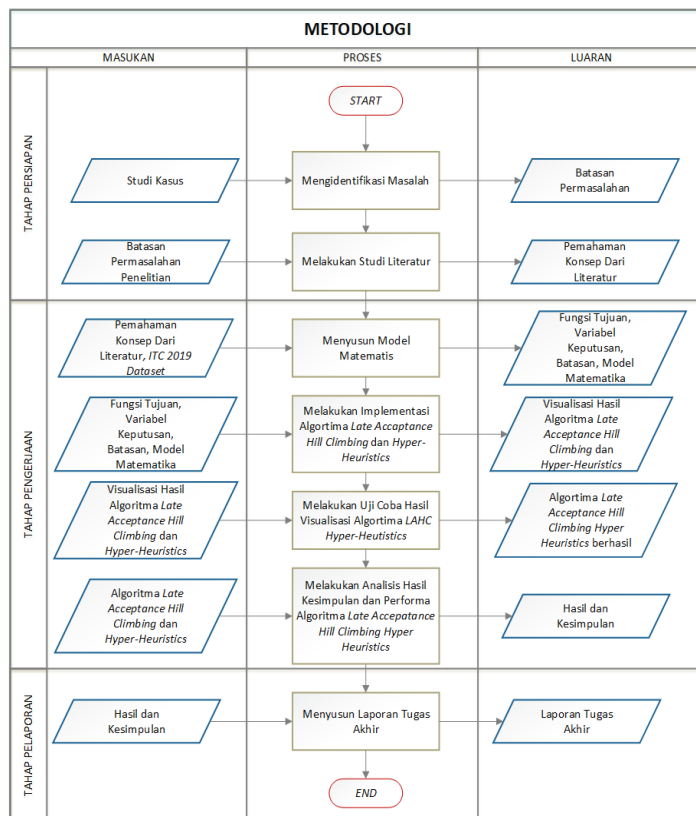
Tahapan ini merupakan langkah awal yang dilakukan dalam pengerjaan penelitian ini. Pada tahapan ini dilakukan identifikasi permasalahan yang ada pada *Course Timetabling Problem* menggunakan data yang disediakan oleh *International Timetabling Competition 2019* [11]. Identifikasi masalah dilakukan dengan mengetahui proses bisnis, batasan, ruang lingkup dan hal terkait lainnya. Hasil yang akan ditemukan adalah permasalahan penjadwalan mata kuliah pada studi kasus tersebut.

B. Studi Literatur

Pada tahapan ini dilakukan pengumpulan berbagai sumber informasi yang memiliki keterkaitan dengan konsep yang akan digunakan dalam pengerjaan penelitian ini serta mengkaji pustaka yang terkait. Pustaka yang digunakan yaitu dapat berupa *paper*, jurnal, laporan penelitian, ataupun penelitian yang terkait dengan permasalahan yang diangkat pada penelitian ini.

C. Pembuatan Model Matematis

Model matematis diperlukan untuk mendapatkan menemukan penyelesaian permasalahan dari data-data yang sudah diperoleh. Tujuan dari pemodelan ini adalah untuk menentukan fungsi tujuan, variabel keputusan dan juga batasan dari permasalahan yang didapatkan. Berikut keterangan untuk masing-masing model:



Gambar 1. Alur Pengerjaan Penelitian

1. Variabel Keputusan

Variabel keputusan merupakan seperangkat variabel yang belum diketahui nilainya yang kemudian akan dicari dalam permasalahan yang didapatkan. Dalam pengerjaan penelitian ini, didapatkan beberapa variabel yang akan dicari hasilnya. Permasalahan ITC 2019 [11] secara umum memiliki dua tujuan yaitu menjadwalkan kelas pada waktu dan ruang yang telah disediakan dan mahasiswa pada kelas yang telah terjadwal. Konten-konten yang kemudian akan menjadi variabel keputusan adalah jumlah times, jumlah courses, jumlah rooms, dan jumlah students.

2. Fungsi Batasan

Batasan merupakan beberapa fungsi yang digunakan untuk memberi batasan terhadap pencapaian fungsi tujuan oleh variabel keputusan. Batasan permasalahan pada ITC 2019 Dataset terbagi menjadi 19 bagian dengan definisi dan ketentuan berbeda-beda untuk masing-masing batasan tersebut. Kemudian batasan tersebut dapat berlaku menjadi Hard constraint dengan diikuti tanda penulisan required yang berarti batasan harus dipenuhi dan bersifat wajib dan menjadi Soft Constraint dengan diberi penalty.

D. Implementasi Algoritma LAHC dan Hyper - Heuristic

Pada tahap ini dilakukan pembuatan *Algoritma Late acceptance hill climbing* dan *Hyper-heuristics* menggunakan bahasa pemrograman Java. Sehingga dapat dijalankan dan diketahui performa dari algoritma tersebut. Dari hasil *generate code* tersebut, akan muncul perhitungan komputasi yang menjawab permasalahan dalam penelitian ini. Proses implementasi diterapkan pada *ITC 2019 Dataset*. Program

akan membaca masukan berupa *file dataset* dengan ekstensi *.xml* yang kemudian diproses menggunakan algoritma *sequential heuristics* untuk menentukan solusi awal. Kemudian solusi awal tersebut akan dioptimasi dengan menggunakan algoritma *Hyper-heuristics*. Dalam kelompok *dataset early* terdapat 10 *instances* yang tersimpan dalam 10 *files* yang berbeda. Kemudian algoritma akan mencari *initial solution* untuk masing-masing *instances* yang menjadi masukan dan selanjutnya akan disimpan pada *fitness array*.

Kemudian dilakukan iterasi, untuk setiap iterasinya algoritma akan menghasilkan hasil yang berbeda-beda. Nilai hasil solusi tersebut yang kemudian akan dibandingkan. Terdapat perbedaan pada algoritma *Greedy Hill Climbing* yaitu pada perbandingan kandidat solusi tepat pada solusi terakhir yang muncul, LAHC melakukan perbandingan kandidat solusi dengan solusi yang telah dilewati pada iterasi sebelumnya.

Tahap selanjutnya adalah menentukan *acceptance criteria* yang akan diterapkan untuk kemudian menentukan apakah solusi pada iterasi tertentu akan diterima atau ditolak. Pada LAHC, solusi yang akan diterima adalah solusi yang memiliki nilai lebih baik atau dibandingkan solusi pembanding sebelumnya. Kemudian juga diperlukan untuk menentukan *stopping criteria* untuk menentukan hasil terbaik untuk iterasi tersebut. Seperti jumlah iterasi ataupun lama waktu yang dihabiskan. Hasil luaran yang didapatkan adalah model solusi untuk penjadwalan mata kuliah beserta mahasiswa yang mengambil mata kuliah tersebut.

E. Uji Coba

Pada tahapan ini adalah tahap dimana dilakukan uji coba algoritma yang telah diimplementasikan pada tahap sebelumnya untuk dilakukan validasi. Pengujian dilakukan pada *ITC 2019 Dataset* yang terdiri dari sepuluh *Instances*. Uji coba dilakukan untuk menunjukkan apakah pemodelan algoritma tersebut sudah dapat memenuhi *hard constraint* dan *soft constraint*, serta apakah hasil yang didapatkan sudah dapat memenuhi fungsi tujuan yang ingin dicapai. Jika ternyata terdapat algoritma yang diimplementasikan tidak sesuai dengan batasan dan ketentuan yang berlaku, akan dilakukan evaluasi kesalahan dan perbaikan terhadap algoritma untuk kemudian dilakukan uji coba kembali sebagai validasi.

Tabel 1. Daftar Batasan *Dataset*

Batasan	Definisi
<i>SameStart</i>	Kelas dimulai pada waktu yang sama
<i>SameTime</i>	Kelas berlangsung dengan rentang waktu yang sama
<i>SameDays</i>	Kelas dijadwalkan pada hari yang sama
<i>SameWeeks</i>	Kelas dijadwalkan pada minggu yang sama
<i>SameRoom</i>	Kelas dijadwalkan pada ruang yang sama
<i>DifferentTime</i>	Kelas dijadwalkan dengan durasi yang berbeda
<i>DifferentDays</i>	Kelas dijadwalkan pada hari yang berbeda
<i>DifferentWeeks</i>	Kelas dijadwalkan pada minggu yang berbeda
<i>DifferentRoom</i>	Kelas dijadwalkan pada ruang yang berbeda
<i>Overlap</i>	Kelas dijadwalkan saling tumpang tindih untuk durasi, hari dan minggu
<i>NotOverlap</i>	Kelas dijadwalkan tidak saling tumpang tindih untuk durasi, hari dan minggu
<i>SameAttendees</i>	Kelas yang dijadwalkan memenuhi kriteria dengan mempertimbangkan waktu perjalanan antara lokasi kelas
<i>Precedence</i>	Kelas harus dijadwalkan pada waktu tertentu yaitu diawal waktu pada setiap minggu, hari, dan slot Dan sebagai berikut.

Batasan	Definisi
<i>WorkDay</i>	Kelas dijadwalkan tidak boleh lebih dari jam kerja
<i>MinGap</i>	Kelas dijadwalkan dengan memiliki minimal waktu selisih antara satu dengan yang lainnya
<i>MaxDays</i>	Kelas dijadwalkan tidak boleh lebih dari hari yang kerja telah ditentukan dalam seminggu
<i>MaxDayLoad</i>	Jumlah durasi kelas tidak boleh lebih dari maksimal slot yang telah ditentukan dalam sehari
<i>MaxBreaks</i>	Jumlah waktu istirahat tidak boleh lebih dari yang telah ditentukan
<i>MaxBlock</i>	Kelas dijadwalkan pada blok-blok tertentu sesuai dengan urutan yang ditentukan

F. Analisis Hasil Performa Algoritma dan Kesimpulan

Pada tahap ini dilakukan analisis hasil performa algoritma. Analisis hasil performa dilakukan dengan mengamati dan membandingkan luaran dari setiap iterasi yang dilakukan untuk setiap *Instances* yang dijadikan sebagai masukan. Analisis juga dilakukan terhadap kecepatan algoritma dalam menyelesaikan permasalahan dan kemampuan algoritma dalam mencapai fungsi tujuan yaitu mencari biaya terendah untuk setiap *resources* yang digunakan. Dari analisis yang dilakukan dapat diketahui kelebihan dan kekurangan dari algoritma yang digunakan untuk menyelesaikan permasalahan penjadwalan mata kuliah.

```
public final class InitSol {
    ArrayList<Integer> weeks, days;
    int start, length, penalty_ts,
    idRoom, pen_rom;
    int[][][] unRoom;
    ArrayList<Kelas> listKelas;
    ArrayList<Minggu> timeslot;
    ArrayList<Integer> klsNoTS;
    InitSol (ArrayList<Kelas> listKls, ArrayList<Minggu> timeslt, int[][][] unR, int[][] travel){
    this.listKelas = listKls;
    this.timeslot = timeslt;
    this.unRoom = unR;
}
```

Gambar 2. Kode Kelas *Initial Solution*

3. HASIL DAN PEMBAHASAN

Hasil implementasi pada dua dataset yang dapat digunakan menghasilkan hasil solusi yang berbeda antara inisial solusi yang dibangun dengan hasil yang didapatkan dari solusi yang telah dioptimasi. Tabel 2 menunjukkan perbandingan data nilai penalti yang didapatkan untuk *dataset tiny*. Rata-rata dari 11 percobaan yang telah dilakukan adalah 8. Sehingga dapat dilihat bahwa terdapat penurunan nilai pinalti jika dibandingkan dengan nilai pinalti manual dan solusi awal. Berdasarkan hasil tersebut terdapat rentang perbedaan nilai sebanyak 45 poin. Hal ini menjadi alasan bahwasannya nilai penalti yang didapatkan dari hasil solusi awal tidak menunjukkan solusi terbaik. Terbukti dengan dilakukannya optimasi pada proses penjadwalan dari solusi awal dan menghasilkan nilai penalti yang jauh berbeda. Begitu pula dengan hasil optimasi pada *dataset* jenis *small 2* yang dijelaskan pada Tabel 2.

Tabel 2 Hasil Nilai Penalti *Dataset Tiny*

Jadwal	Solusi Awal	Rata-rata Iterasi 1.000.000
Nilai Pinalti	53	8

Tabel 3 Hasil Nilai Penalti *Dataset Small 2*

Jadwal	Solusi Awal	Rata-rata Iterasi 1.000.000
Nilai Pinalti	1536	865

Rata-rata dari 11 percobaan tersebut adalah 865. Sehingga dapat dilihat bahwa terdapat penurunan nilai pinalti jika dibandingkan dengan nilai penalti solusi awal sebanyak 671. Terdapatnya perbedaan nilai penalti antara solusi awal dan solusi akhir yang berarti telah dilakukan optimasi dikarenakan teknik yang berbeda disaat pemilihan jadwal hari, waktu dan ruang tidak memperhatikan nilai penalti yang melekat pada pilihan jadwal tersebut. Solusi awal hanya mempertimbangkan apakah batasan wajib telah terpenuhi tanpa memperdulikan nilai penalti yang terdapat sehingga berapapun nilai penalti yang dimiliki oleh pilihan jadwal tersebut dapat diterima. Sedangkan untuk solusi akhir yang didapatkan berasal dari hasil pertimbangan dengan melihat nilai penalti yang berlaku. Hal inilah yang menjadi alasan mengapa solusi awal tidak dapat dikatakan solusi terbaik yang dihasilkan sehingga perlu dilakukan optimasi untuk menghasilkan nilai akhir yang jauh lebih baik dari nilai solusi awal.

4. KESIMPULAN

Kesimpulan yang dapat diambil dari penelitian ini antara lain adalah:

1. Penerapan Algoritma *Late acceptance Hill climbing* berbasis *Hyper-heuristics* mampu menyelesaikan permasalahan penjadwalan mata kuliah (*University Course Timetabling Problem*).
2. Struktur data yang efisien digunakan sebagai struktur data *hyper-heuristic* adalah struktur data berbasis *arraylist* dan pemrograman berbasis objek karena waktu komputasi yang relatif lebih cepat dan struktur data yang sangat kompleks.
3. Pada algoritma *Late acceptance Hill climbing Hyper-heuristics* tanpa pengembangan, jumlah *low level heuristics* dan jumlah iterasi berpengaruh pada solusi yang dihasilkan. Penggunaan dua jenis *low level heuristics* menghasilkan solusi yang semakin optimum daripada enam jenis *low level heuristic*. Sementara semakin banyak iterasi, solusi yang dihasilkan semakin optimum.
4. Pada algoritma *Late acceptance Hill climbing Hyper-heuristics* dengan pengembangan, beberapa parameter mempengaruhi hasil.
 - a. Jumlah iterasi yang menjadi kriteria batasan akan menghasilkan nilai penalti yang semakin baik dengan jumlah iterasi ditingkatkan lebih banyak.

- b. Perubahan kriteria yang menjadi batasan iterasi *idle* akan mempengaruhi nilai dengan meminimalkan nilai *idle* tersebut.
5. Algoritma LAHC mampu memberikan performa yang baik dalam penyelesaian masalah timetabling bila dibandingkan dengan algoritma pembanding *Hill climbing*. Kemudian juga mampu melakukan optimasi terhadap nilai penalti yang dikenakan pada masing-masing konten.

Berdasarkan hasil dan kesimpulan diatas, saran yang dapat diberikan untuk penelitian selanjutnya adalah sebagai berikut:

1. Penelitian ini hanya dapat menjalankan 2 *dataset* dari total 15 *dataset* yang sudah dikeluarkan oleh pihak ITC 2019. Harapannya penelitian selanjutnya dapat menyelesaikan lanjutan dari *dataset* yang belum berhasil diselesaikan hingga menemukan solusi inisial.
2. Penelitian hanya melakukan eksperimen sebanyak 11 kali iterasi untuk tiap-tiap skenario sehingga iterasi yang dilakukan tidak terlalu banyak. Harapannya dapat lebih banyak menghasilkan variasi.
3. Penelitian ini hanya menggunakan *low level heuristics* berupa *move*. Untuk mendapatkan hasil yang lebih variatif, *low level heuristics* dapat ditambah, sehingga memungkinkan ditemukan solusi yang lebih optimum. Eksperimen total *low level heuristics* yang menghasilkan solusi lebih optimum juga dapat dilakukan pada penelitian selanjutnya.

DAFTAR PUSTAKA

- [1] S. Putra, M. Iqbal, A. Putera, and U. Siahaan, "Analysis Of Course Distribution Scheduling For Lecturers Using Genetic Algorithms And Analysis Of Course Distribution Scheduling For Lecturers Using Genetic Algorithms And Constraint Satisfaction Methods At Batam University," *Informatika dan Sains*, vol. 14, no. 04, p. 2024, doi: 10.54209/infosains.v14i04.
- [2] A. Muklason, A. Marom, and I. G. A. Premananda, "Automated Course Timetabling Optimization Using Tabu-Simulated Annealing *Hyper-Heuristics* Algorithm," *Khazanah Informatika : Jurnal Ilmu Komputer dan Informatika*, vol. 10, no. 1, pp. 15–21, Apr. 2024, doi: 10.23917/khif.v10i1.4835.
- [3] A. Naskar, R. Pramanik, S. K. S. Hossain, S. Mirjalili, and R. Sarkar, "Late acceptance hill climbing aided chaotic harmony search for feature selection: An empirical analysis on medical data," *Expert Syst Appl*, vol. 221, p. 119745, Jul. 2023, doi: 10.1016/j.eswa.2023.119745.
- [4] R. R. Hutama, "Pembentukan Solusi Awal International Timetabling Competition 2021," *JATISI (Jurnal Teknik Informatika dan Sistem Informasi)*, vol. 8, no. 4, pp. 1939–1944, Dec. 2021, doi: 10.35957/jatisi.v8i4.1155.
- [5] E. K. Burke *et al.*, "Hyper-heuristics: a survey of the state of the art," *Journal of the Operational Research Society*, vol. 64, no. 12, pp. 1695–1724, Dec. 2013, doi: 10.1057/jors.2013.71.
- [6] G. Nagarajan and L. D. Dhinesh Babu, "A hybrid of whale optimization and late acceptance hill climbing based imputation to enhance classification performance in electronic health records," *J Biomed Inform*, vol. 94, p. 103190, Jun. 2019, doi: 10.1016/j.jbi.2019.103190.

- [7] J. H. Drake, E. Özcan, and E. K. Burke, "A Case Study of Controlling Crossover in a Selection Hyper-heuristic Framework Using the Multidimensional Knapsack Problem," *Evol Comput*, vol. 24, no. 1, pp. 113–141, Mar. 2016, doi: 10.1162/EVCO_a_00145.
- [8] D. Romaguera, J. Plender-Nabas, J. Matias, and L. Austero, "Development of a Web-based Course Timetabling System based on an Enhanced Genetic Algorithm," *Procedia Comput Sci*, vol. 234, pp. 1714–1721, 2024, doi: 10.1016/j.procs.2024.03.177.
- [9] M. Kumar Sharma and B. Kumar Chauhan, "Timetable rationalization & Operational improvements by human intervention in an urban rail transit system: An exploratory study," *Transp Res Interdiscip Perspect*, vol. 13, p. 100526, Mar. 2022, doi: 10.1016/j.trip.2021.100526.
- [10] S. Abdipoor, R. Yaakob, S. L. Goh, and S. Abdullah, "Meta-heuristic approaches for the University Course Timetabling Problem," *Intelligent Systems with Applications*, vol. 19, p. 200253, Sep. 2023, doi: 10.1016/j.iswa.2023.200253.
- [11] T. Müller, H. Rudová, and Z. Müllerová, "Real-world university course timetabling at the International Timetabling Competition 2019," *Journal of Scheduling*, Apr. 2024, doi: 10.1007/s10951-023-00801-w.
- [12] J. H. Drake, A. Kheiri, E. Özcan, and E. K. Burke, "Recent advances in selection hyper-heuristics," *Eur J Oper Res*, vol. 285, no. 2, pp. 405–428, Sep. 2020, doi: 10.1016/j.ejor.2019.07.073.
- [13] A. Lamghari and R. Dimitrakopoulos, "Hyper-heuristic approaches for strategic mine planning under uncertainty," *Comput Oper Res*, vol. 115, p. 104590, Mar. 2020, doi: 10.1016/j.cor.2018.11.010.
- [14] M. Alzaqebah, S. Jawarneh, M. Alwohaibi, M. K. Alsmadi, I. Almarashdeh, and R. Mustafa A. Mohammad, "Hybrid Brain Storm Optimization algorithm and Late Acceptance Hill Climbing to solve the Flexible Job-Shop Scheduling Problem," *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 6, pp. 2926–2937, Jun. 2022, doi: 10.1016/j.jksuci.2020.09.004.
- [15] J. Hall and K. M. Brown, "Creating feelings of inclusion in adventure tourism: Lessons from the gendered sensory and affective politics of professional mountaineering," *Ann Tour Res*, vol. 97, p. 103505, Nov. 2022, doi: 10.1016/j.annals.2022.103505.