

## Kajian Algoritma Optimasi Penjadwalan Mata Kuliah

Tri Handayani<sup>1</sup>; Dhomas Hatta Fudholi<sup>2</sup>; Septia Rani<sup>3</sup>

<sup>1, 2, 3</sup>Jurusan Informatika, Fakultas Teknologi Industri, Universitas Islam Indonesia

<sup>1</sup>17523188@students.uii.ac.id

<sup>2</sup>hatta.fudholi@uui.ac.id

<sup>3</sup>septia.rani@uui.ac.id

### ABSTRACT

Course scheduling is an important thing to do at the beginning of the academic semester. The process of arranging lecture schedules manually is often difficult because there are several constraints that require a long time to complete. This study aims to examine the algorithms that are appropriate to the subject matter of scheduling the course. Search and analysis are carried out on the literature relating to scheduling optimization. Literature search is performed on Google Scholar and Science Direct by entering the main keywords "course timetable", "university timetable problem", "school scheduling", and "scheduling algorithm". The results of the literature analysis include domain distribution, algorithm analysis and gaps from previous studies. In previous studies there are deficiencies such as algorithms that cannot produce optimal solutions. The domain distribution results obtained are universities and schools with a percentage of 88% and 12% of the total papers. The findings of 14 algorithm distributions can be classified into 3 methods, namely heuristic, metaheuristic, and hyper-heuristic. Based on the results of the analysis, several recommendations can be given. For fast optimization, Simulated Annealing (SA) can be a solution because it is able to produce solutions with the process time of 0.481-10.102s. For the best time and fitness value solutions, Genetic Algorithm (GA) can be a solution because it is able to produce solutions with the process time of 0.964-73.461s and fitness value to 1.

**Keywords:** optimization, scheduling, algorithm

### ABSTRAK

Penjadwalan mata kuliah merupakan hal penting yang dilakukan pada awal semester akademik. Proses penyusunan jadwal kuliah secara manual seringkali mengalami kesulitan karena terdapat beberapa konstrain sehingga membutuhkan waktu yang lama. Penelitian ini bertujuan mengkaji algoritma-algoritma yang sesuai dengan masalah penjadwalan mata kuliah. Pencarian dan analisis dilakukan terhadap literatur yang berkaitan dengan optimasi penjadwalan. Proses pencarian literatur dilakukan pada Google Scholar dan Science Direct dengan memasukkan kata kunci utama "course timetable", "university timetable problem", "school scheduling", dan "algoritma penjadwalan". Hasil analisis literatur meliputi sebaran domain, analisis algoritma serta gap dari penelitian sebelumnya. Pada penelitian sebelumnya terdapat kekurangan seperti algoritma yang tidak dapat menghasilkan solusi optimal. Hasil sebaran domain yang diperoleh ialah universitas dan sekolah dengan persentase 88% dan 12% dari keseluruhan makalah. Adanya temuan 14 sebaran algoritma dapat diklasifikasikan menjadi 3 metode, yaitu heuristic, metaheuristic, dan hyper-heuristic. Berdasarkan hasil analisis, dapat diberikan beberapa rekomendasi. Untuk optimasi yang cepat, Simulated Annealing (SA) dapat menjadi solusi karena mampu menghasilkan solusi dengan waktu 0.481-10.102s. Untuk solusi waktu dan nilai fitness terbaik, Genetic Algorithm (GA) dapat menjadi solusi karena mampu menghasilkan solusi dengan waktu 0.964-73.461s dan nilai fitness 1.

**Kata kunci:** optimasi, penjadwalan, algoritma

## 1. PENDAHULUAN

Penjadwalan mata kuliah merupakan hal penting yang dilakukan oleh program studi pada awal semester akademik agar dapat terlaksananya proses belajar mengajar yang baik. Dalam proses penyusunan jadwal perkuliahan terdapat beberapa hal yang kompleks dan sering mengalami kesulitan dalam penyusunannya, seperti pengalokasian mata kuliah, dosen dan ruangan yang akan digunakan agar tidak terjadi bentrok dengan jadwal mata kuliah [1].

Proses penyusunan penjadwalan, misalnya pada penjadwalan mata kuliah, dirasa cukup kompleks. Hal ini disebabkan karena terdapat beberapa hal/konstrain yang harus dipenuhi, antara lain: (a) dosen mengampu beberapa mata kuliah berbeda, (b) kebutuhan ruangan untuk mata kuliah tertentu, (c) keterbatasan jumlah ruangan yang dapat digunakan dalam waktu bersamaan, (d) permintaan dosen yang bersangkutan untuk tidak mengajar pada slot waktu tertentu. Proses penyusunan jadwal secara manual yang dilakukan dengan mengalokasikan mata kuliah dan dosen yang akan mengampu, mengalokasikan mata kuliah pada ruang yang terbatas dan sesi tertentu, membutuhkan waktu yang relatif lama.

Penelitian-penelitian sebelumnya [2-6] telah mencoba untuk menerapkan metode-metode optimasi dengan berbagai kasus seperti penjadwalan universitas dan sekolah dengan berbagai algoritma. Dari penelitian-penelitian sebelumnya, hasil optimal pun dapat diperoleh, missal pada penelitian [2]. Namun, terdapat juga penelitian yang tidak mampu menghasilkan solusi optimal karena jumlah data yang besar [3], sehingga membutuhkan waktu komputasi yang cukup lama.

Kajian ini bertujuan untuk mengetahui metode yang relevan untuk optimasi masalah penjadwalan mata kuliah pada saat ini. Kajian dilakukan dengan pencarian literatur yang berkaitan dengan optimasi penjadwalan. Pada literatur tersebut dilakukan analisis terhadap jenjang pendidikan, kinerja algoritma yang digunakan, serta hasil dari penelitian untuk mencari informasi berkaitan dengan masalah penjadwalan. Hasil dari kajian diharapkan dapat menjadi rujukan metode penjadwalan mata kuliah pada tahap selanjutnya dan menghasilkan rekomendasi metode yang dapat digunakan untuk optimasi penjadwalan mata kuliah.

## 2. METODE

Pada bagian ini akan dipaparkan metode yang digunakan untuk pencarian literatur serta metode analisis yang dilakukan pada literatur terpilih.

### 2.1. Metode Pencarian

Proses pencarian literatur dilakukan pada Google Scholar dan Science Direct dengan memasukkan kata kunci utama *course timetable*, *university timetable problem*, *school scheduling*, dan *algoritma penjadwalan*. Strategi pencarian dengan menelusuri daftar pustaka dari literatur yang ditemukan juga digunakan untuk mendapatkan literatur terkait. Dari hasil pencarian dilakukan pemilihan literatur berdasarkan kriteria berikut:

- a. Literatur yang dipilih merupakan publikasi ilmiah dengan rentang tahun 2015 hingga 2020, yang mengidikasikan relevansi kebaruan.
- b. Literatur membahas optimasi penjadwalan. Hal ini berkaitan dengan tujuan kajian makalah ini.
- c. Literatur memberikan detail implementasi algoritma pada kasus penjadwalan universitas dan sekolah.
- d. Literatur dapat dituliskan dalam Bahasa Indonesia maupun Bahasa Inggris.

## 2.2. Metode Analisis

Literatur yang terpilih pada proses pencarian kemudian akan dianalisis yang terlebih dahulu dipaparkan dalam bentuk tabel. Analisis-analisis yang dilakukan bertujuan untuk mendapatkan informasi-informasi yang dapat mendukung pencarian solusi optimasi penjadwalan mata kuliah. Analisis yang dilakukan menitikberatkan pada dua konteks utama, yaitu domain penelitian dan algoritma/metode yang digunakan. Pada konteks domain penelitian, dipaparkan analisis yang melihat karakteristik objek penelitian, termasuk di dalamnya jenjang pendidikan yaitu universitas atau sekolah. Disamping itu, analisis dari proses penjadwalan juga dilakukan dengan melihat data terkait, seperti mata kuliah/pelajaran, jumlah dosen/guru, ketersediaan ruangan, sesi perkuliahan dalam 1 (satu) hari dan jumlah hari belajar dalam seminggu. Pada konteks algoritma, dilakukan perbandingan kinerja algoritma pada kasus yang berbeda-beda dan kinerja antar algoritma penjadwalan. Analisis algoritma meliputi waktu *running time* dan nilai *fitness* terbaik.

## 3. HASIL DAN PEMBAHASAN

Berdasarkan pencarian literatur dengan kata kunci yang diberikan, ditemukan 55 makalah. Dari ke 55 makalah tersebut, terdapat 26 makalah yang memenuhi kriteria pencarian. Makalah-makalah yang tidak memenuhi kriteria, antara lain 14 makalah tidak memenuhi rentang tahun penulisan antara 2015 hingga 2020, 6 makalah tidak membahas optimasi penjadwalan, 4 makalah tidak memenuhi kriteria implementasi kasus penjadwalan universitas dan sekolah, dan 5 makalah tidak lengkap pembahasannya mengenai penjadwalan. Tabel 1 menunjukkan perbandingan antara makalah-makalah yang terpilih sesuai kriteria. Tabel 1 memberikan perbandingan langsung kriteria domain (termasuk di dalamnya jumlah matakuliah/pelajaran, jumlah pengajar, ruang, sesi dan hari) dan algoritma (*running time* dan keunggulan) pada masing-masing makalah.

Berdasarkan Tabel 1, secara umum, penjadwalan dapat dibedakan menjadi 2 tipe yaitu penjadwalan universitas dan penjadwalan sekolah. Penjadwalan universitas dan sekolah menggunakan data mata kuliah (MK) dan pelajaran dengan jumlah yang bervariasi. Di dalam makalah yang dianalisis ditemukan juga tenaga pengajar dosen pada universitas jumlahnya lebih banyak dibandingkan jumlah guru di sekolah. Terkait dengan waktu kegiatan belajar mengajar, secara umum kegiatan kuliah dan sekolah dilaksanakan 5-6 hari dalam seminggu dengan beberapa sesi per harinya.

Dari makalah tersebut, diperoleh 14 macam algoritma penjadwalan. Algoritma penjadwalan ini dapat dikelompokkan ke dalam tiga jenis metode yaitu *heuristic*, *metaheuristic* dan *hyper-heuristic*. Kinerja dari algoritma-algoritma ini dapat menghasilkan solusi optimal ketika mencapai nilai *fitness* 1. Dilihat dari konteks komputasi, *running time* yang dibutuhkan untuk memperoleh hasil optimal bervariasi dalam rentang waktu 0.964s hingga 11.1 jam. Hal ini berkaitan dengan banyaknya data yang diuji. Semakin banyak data yang diujikan, maka *running time* semakin lama. Berdasarkan hasil analisis yang dipaparkan pada Tabel 1, selanjutnya akan dilakukan analisis domain, analisis algoritma, dan analisis *gap* dari penelitian secara lebih detail.

### 3.1. Analisis Domain

Dari hasil kajian literatur, diperoleh domain yang digunakan oleh peneliti sebelumnya dengan persentase untuk domain universitas dan sekolah adalah 88% dan 12% dari keseluruhan makalah. Dalam proses penjadwalan di universitas dan sekolah, terdapat hal yang perlu diperhatikan yaitu jumlah mata kuliah/pelajaran, jumlah dosen/guru, jumlah ruangan yang tersedia, sesi kegiatan belajar mengajar dalam 1 (satu) hari dan jumlah hari kegiatan belajar mengajar dalam 1 (satu) minggu. Penjadwalan di universitas dilakukan oleh program studi ataupun fakultas.

Tabel 1. Hasil Kajian Literatur

Makalah	Tahun	Algoritma	Domain	Running Time (s)	Jumlah MK / Pelajaran	Jumlah Pengajar	Jumlah Ruang	Sesi	Hari	Hasil/Kelebihan
[1]	2019	Genetic Algorithm	Universitas	882	26	13	7	12	5	Menghasilkan kombinasi jadwal menggunakan sampel 6 dosen dengan waktu cepat.
[2]	2018	Graph Coloring	Universitas	N/A	39	18	8	3	5	Dapat menghasilkan jadwal kuliah dengan optimal.
[3]	2019	Greedy Algorithm	Universitas	6000	N/A	1200	94 & 21	12	5	Mampu meningkatkan 12% kualitas solusi tanpa menambahkan waktu CPU yang signifikan, namun belum dapat menemukan solusi optimal.
[4]	2019	Variable Neighborhood Search, Tabu Search	Universitas	120	32 & 36	N/A	11	15	5	Dataset meningkat 1855 dan 1110 dibanding manual, dataset manual 2675 dan 1984, sedangkan solusi optimal adalah 820 dan 874.
[5]	2020	Genetic Algorithm (Fitness Algorithm), Tabu Search Algorithm	Universitas	N/A	28	19	8	4	6	Nilai yang dihasilkan Fitness Algorithm adalah 40 sedangkan Tabu Search memperoleh nilai 15 sehingga Tabu Search lebih baik untuk menentukan jadwal.
[6]	2019	Genetic-Tabu Search, Ant Colony	Universitas	122.86 & 12.22	70	20	11	N/A	N/A	Di penggunaan memori dan waktu, Ant Colony lebih sedikit dibandingkan AG-TS. Namun, nilai <i>fitness</i> yang dihasilkan AG-TS lebih baik dibandingkan Ant Colony, sehingga bentrok lebih sedikit.
[7]	2019	Multi-Agent System	Universitas	N/A	N/A	N/A	N/A	N/A	N/A	Model agen mampu mencapai tujuan dan juga memenuhi batasan keras dan lunak yang ditentukan.
[8]	2015	Genetic Algorithm	Sekolah	N/A	N/A	N/A	N/A	N/A	6	Hasil didapatkan pada iterasi ke 80 dengan rata-rata <i>fitness</i> tertinggi.
[9]	2015	Firefly Algorithm	Universitas	N/A	Semua MK di FST	Sesuai MK	22	4	5	Nilai intensitas cahaya terbaik dicapai pada $\beta_0$ bernilai 1 dan $\gamma$ bernilai 10 dengan nilai intensitas cahaya 0.0003831.
[10]	2017	Graph Coloring	Universitas	N/A	30	16	N/A	4	6	Menempatkan setiap kejadian secara sekuensial dalam periode waktu yang valid sehingga tidak ada konflik antar kejadian.

**Tabel 1.** Hasil Kajian Literatur (lanjutan)

Makalah	Tahun	Algoritma	Domain	Running Time (s)	Jumlah MK / Pelajaran	Jumlah Pengajar	Jumlah Ruang	Sesi	Hari	Hasil/Kelebihan
[11]	2015	Particle Swarm Optimization	Universitas	N/A	N/A	N/A	10	9	6	Menghasilkan jadwal perkuliahan tercepat setelah 23 iterasi dan terlama 48 iterasi.
[12]	2017	Hill Climbing Algorithm	Universitas	N/A	63	125	N/A	N/A	N/A	Solusi jadwal diperiksa satu persatu sehingga proses cepat dan hasil mendekati optimal.
[13]	2016	Genetic Algorithm	Sekolah	N/A	N/A	N/A	9	8	6	Menghasilkan jadwal pelajaran yang optimal.
[14]	2015	Genetic Algorithm	Universitas	N/A	28	29	12	12	5	Dengan metode best <i>fitness</i> menghasilkan generasi selanjutnya lebih baik atau minimal sama dengan generasi sebelumnya.
[15]	2016	Genetic Algorithm	Universitas	16200 & 39960	150	131	32	10	5	GA menghasilkan waktu komputasi yang lebih lama untuk ukuran data yang besar.
[16]	2015	Ant Colony Optimization (Max-Min Ant Algorithm)	Universitas	N/A	N/A	N/A	N/A	7	5	Bersifat heuristic untuk menyelesaikan masalah optimasi dengan menghasilkan jalur terpendek dari sekian banyak jalur.
[17]	2017	Genetic Algorithm	Universitas	180-600	59	11	8	8	5	Proses penjadwalan lebih cepat dan tanpa bentrok.
[18]	2019	Particle Swarm Optimization	Universitas	214	26	24	8	3	5	Untuk mendapatkan nilai <i>fitness</i> yang baik membutuhkan waktu eksekusi yang besar. PSO dapat menyelesaikan penjadwalan tetapi belum menghasilkan solusi yang optimal.
[20]	2018	Genetic Algorithm (Tournament Selection, Truncation, Roulette Wheel)	Sekolah	N/A	N/A	61	27	N/A	N/A	Tournament Selection menghasilkan generasi terbanyak dengan nilai <i>fitness</i> terbaik 0.51. Truncation adalah metode dengan konvergensi tercepat, tetapi nilai <i>fitness</i> lebih rendah. Roulette Wheel menghasilkan konvergensi dan nilai <i>fitness</i> diantara dua metode lainnya.

**Tabel 1.** Hasil Kajian Literatur (lanjutan)

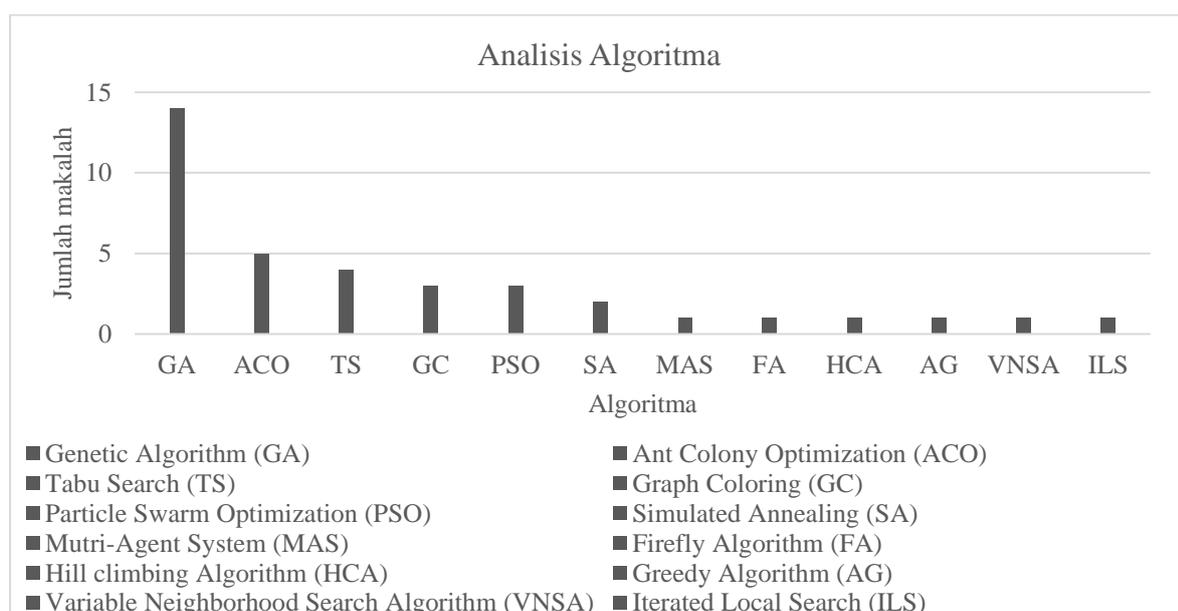
Makalah	Tahun	Algoritma	Domain	Running Time (s)	Jumlah MK / Pelajaran	Jumlah Pengajar	Jumlah Ruang	Sesi	Hari	Hasil/Kelebihan
[21]	2020	Genetic Algorithm, Ant Colony	Universitas	0.9856 & 6.6251	N/A	N/A	5	4	6	Dengan 10 tes, GA membutuhkan waktu 0.9856 s dengan rata-rata siklus 4.9. Ant Colony membutuhkan waktu 6.6251 s dengan rata-rata siklus 6.3.
[22]	2018	Ant Colony Optimization (Max-Min Ant System)	Universitas	30 - 360	N/A	N/A	N/A	N/A	N/A	Variasi Max-Min ACO menggunakan local search berkinerja lebih baik daripada variasi Rank-based.
[23]	2018	Genetic Algorithm, Simulated Annealing, Hybrid Algorithms	Universitas	0.964-73.461, 0.481-10.102, 0.709-44.042	N/A	N/A	N/A	N/A	N/A	GA berhasil meningkatkan nilai <i>fitness</i> pada ukuran data kecil hingga sedang. SA menghasilkan waktu eksekusi yang lebih baik. Hybrid Algorithm GA-SA berhasil mencapai nilai <i>fitness</i> nol untuk semua ukuran data.
[24]	2019	Iterated Local Search, Simulated Annealing, Tabu Search	Universitas	41.73, 2.59, 15.18	N/A	N/A	N/A	9	5	SA menemukan solusi valid tanpa pelanggaran konstrain lebih cepat daripada metode lainnya. Kinerja TS lebih lambat dari SA tetapi lebih cepat dari ILS. Sedangkan ILS membutuhkan waktu lebih lama untuk menemukan solusi.
[25]	2019	Genetic Algorithm, Particle Swarm Optimization	Universitas	9.36, 61.95	N/A	N/A	N/A	N/A	N/A	GA menghasilkan solusi lebih cepat tetapi nilai <i>fitness</i> yang dihasilkan lebih buruk yaitu 0.021. Sedangkan PSO menghasilkan nilai <i>fitness</i> 0.099 tetapi membutuhkan waktu lebih lama.
[26]	2016	Genetic Algorithm, Ant Colony Optimization	Universitas	21.26, 69.11	N/A	N/A	N/A	N/A	N/A	GA menghasilkan solusi terbaik pada percobaan ke-5 dengan nilai <i>fitness</i> 1. Sedangkan Ant Colony menghasilkan solusi terbaik pada percobaan ke-6 dengan nilai <i>fitness</i> 1000.

Berdasarkan mata kuliah/pelajaran, mata kuliah di program studi berjumlah dalam rentang 26-70 mata kuliah. Sedangkan untuk sebuah fakultas memiliki mata kuliah dengan berjumlah hingga 150 mata kuliah. Untuk jenjang pendidikan sekolah, tidak ditemukan informasi terkait jumlah mata pelajaran yang diuji. Berdasarkan jumlah dosen/guru, rata-rata setiap program studi memiliki dosen berjumlah 11-125 orang. Sedangkan jumlah dosen yang dimiliki fakultas berjumlah 131-1200 orang. Untuk sekolah hanya diperoleh informasi berjumlah 61 orang. Berdasarkan jumlah ruangan, di program studi tersedia 5-22 ruangan. Sedangkan di fakultas tersedia 32-55 ruangan dan di sekolah tersedia 9-61 ruangan. Berdasarkan sesi kegiatan belajar mengajar dalam satu hari, secara umum universitas berjumlah 3-15 sesi dan sekolah berjumlah 7-8 sesi. Berdasarkan kegiatan belajar mengajar dalam satu minggu, secara umum universitas dan sekolah dilaksanakan 5-6 hari.

### 3.2. Analisis Algoritma

Hasil analisis diperoleh 14 algoritma berbeda yang dapat digunakan untuk optimasi penjadwalan. Gambar 2 menunjukkan algoritma yang masih banyak digunakan dalam rentang tahun 2015-2020, di antaranya Genetic Algorithm (GA), Ant Colony Optimization (ACO), Tabu Search (TS), Graph Coloring (GC), Particle Swarm Optimization (PSO) dan beberapa algoritma penjadwalan lainnya. Analisis algoritma yang dilakukan didasarkan pada masing-masing penelitian terhadap algoritma yang digunakan. Berdasarkan hasil analisis yang ditunjukkan pada Gambar 2, dilakukan analisis pada algoritma-algoritma yang dominan.

Genetic Algorithm (GA) merupakan algoritma yang masih banyak digunakan sebagaimana ditunjukkan pada Gambar 2. Berdasarkan makalah [1], [14], [17], GA mampu menghasilkan solusi optimal penjadwalan sebuah program studi dengan rata-rata *running time* lebih cepat yaitu rentang waktu 3 sampai 14.7 menit. GA mampu mendapatkan solusi optimal banyak program studi yaitu tingkat fakultas. Hal tersebut terdapat pada makalah [15] bahwa GA menghasilkan solusi optimal dalam rentang waktu 4.5 hingga 11.1 jam yang relatif lebih lama karena *dataset* yang digunakan lebih besar. Selain penjadwalan mata kuliah, GA mampu menghasilkan solusi optimal pada penjadwalan mata pelajaran. Hal itu sesuai pada makalah [13], [20]. Namun pada makalah [8], GA belum mampu menunjukkan hasil optimal.



**Gambar 2.** Hasil Analisis Algoritma

Ant Colony Optimization (ACO) yang bersifat *heuristic* untuk menyelesaikan masalah optimasi dengan menggunakan jalur terpendek. Hal ini merupakan solusi yang dihasilkan pada makalah [16]. Pada makalah [22] dengan pemanfaatan *path attractives heuristic*, ACO dengan *local search* menghasilkan solusi optimal dengan kinerja lebih baik dibandingkan dengan *Rank-based*.

Algoritma Tabu Search (TS) pada makalah [4] adanya pengembangan TS dengan Variable Neighborhood Search Algorithm (VNSA) dan Hyper-Heuristic Algorithm (HHA). Dengan waktu kurang dari 120s mampu menghasilkan solusi optimal. *Dataset* yang menggunakan cara manual menghasilkan 1675 dan 1984 *dataset* mengalami peningkatan menjadi 820 dan 874 *dataset* setelah dilakukan optimasi.

Algoritma Graph Coloring (GC) mampu menghasilkan solusi optimal jadwal mata kuliah program studi. Hal ini sesuai dengan hasil penelitian pada makalah [2], [10]. Tetapi terdapat perbedaan di antara keduanya, yaitu pada jumlah mata kuliah yang diujikan, jumlah dosen, sesi perkuliahan satu hari serta hari kegiatan belajar dalam satu minggu. Adanya pengembangan dari GC yang digabungkan dengan Genetic Algorithm (GA) pada makalah [19]. GC mampu mengurangi biaya *penalty* rata-rata sekitar 70% dan nilai *fitness* hanya meningkat 10% dari populasi awal setelah 150 generasi.

Particle Swarm Optimization (PSO) mampu menghasilkan solusi optimal. Namun pada makalah [11] belum mampu menghasilkan penjadwalan pada waktu yang berkualitas untuk perkuliahan. Sedangkan pada makalah [18], PSO mampu menghasilkan solusi dengan waktu 214s dengan 26 mata kuliah, 24 dosen, 8 ruangan tersedia, 3 sesi perhari dan 5 hari kuliah. Apabila ukuran data semakin besar maka peluang mendapatkan nilai *fitness* yang baik semakin besar tetapi waktu eksekusinya semakin bertambah.

Berdasarkan hasil pada Tabel 1, terdapat penelitian tentang perbandingan terhadap algoritma-algoritma penjadwalan. Terdapat makalah perbandingan GA dan ACO yaitu makalah [6], [21] dan [26]. Berdasarkan makalah [21], [26], *running time* untuk memperoleh solusi optimal GA lebih cepat dibandingkan ACO dan memperoleh nilai *fitness* 1. Hal ini dikarenakan ACO memiliki kompleksitas yang lebih tinggi dibandingkan GA. Sedangkan pada makalah [6], GA dilakukan pengembangan dengan algoritma TS dibandingkan dengan ACO. GA-TS mampu memperoleh solusi optimal dengan nilai *fitness* yang lebih baik dibandingkan ACO. Namun, waktu yang dibutuhkan GA-TS lebih lama dibandingkan ACO yaitu 122.86s dan 12.22s. Pada penelitian lainnya dilakukan perbandingan GA dan PSO. Hal ini sesuai pada makalah [25], nilai kesesuaian PSO mengungguli GA sehingga PSO memiliki kecenderungan nilai *fitness* lebih baik dari GA. Tetapi kecepatan eksekusi GA memiliki waktu yang lebih cepat dibandingkan PSO.

Pada makalah [5] membandingkan Algoritma Fitness yang merupakan bagian dari GA dengan algoritma TS. Hasil perbandingan menunjukkan bahwa GA menghasilkan nilai *fitness* yang tidak konsisten dan nilai yang dihasilkan besar, sementara itu nilai *fitness* TS lebih kecil sehingga baik digunakan untuk menentukan penjadwalan. Algoritma TS juga dilakukan perbandingan dengan Iterated Local Search (ILS) dan Simulated Annealing (SA) pada makalah [24]. Hasil perbandingan menunjukkan ILS, SA dan TS memperoleh solusi optimal dengan waktu 41.73s, 2.59s, dan 15.18s. Dalam penelitian lainnya dilakukan perbandingan dengan GA dan Hybrid Algorithm (HA) yang menggabungkan dua algoritma yaitu GA dan SA. Penelitian ini menggunakan berbagai tingkatan data uji S hingga XXL. Berdasarkan makalah [23] diperoleh hasil bahwa komputasi GA dan SA berhenti pada data XL dan XXL. Sementara Hybrid Algoritma GA-SA menghasilkan kinerja yang baik dan mampu berjalan hingga data XXL. GA, SA, dan HA memperoleh solusi dengan waktu masing-masing 0.964-73.461s, 0.481-10.102s, dan 0.709-44.042s.

Berdasarkan hasil kajian literatur terdapat juga algoritma yang belum banyak digunakan untuk masalah optimasi penjadwalan. Sehingga hanya ditemukan sedikit literatur yang menggunakan algoritma tersebut, algoritma tersebut yaitu Multi-Agent System (MAS), Firefly Algorithm (FA), Hill Climbing Algorithm (HCA), Greedy Algorithm (AG), VNS Algorithm, dan Iterated Local Search (ILS). Algoritma-algoritma ini sudah mampu menyelesaikan masalah penjadwalan mata kuliah dan mata pelajaran. Namun, [3],[24], masih belum memperoleh solusi penjadwalan yang maksimal.

### **3.3. Analisis Gap Penelitian**

Penelitian sebelumnya terdapat 88% makalah dengan kasus penjadwalan universitas yang mampu mengoptimasi proses penjadwalan mata kuliah. Berdasarkan temuan, penelitian dengan penggunaan Genetic Algorithm (GA) berkinerja baik dalam waktu dan nilai *fitness*. Namun GA memiliki kekurangan apabila dilakukan pengembangan dengan algoritma Tabu Search (TS) dari sisi *running time* apabila dibandingkan dengan algoritma lainnya, seperti Ant Colony Optimization (ACO) dan Simulated Annealing (SA). Hal ini sebagaimana dijelaskan pada makalah [6] dan [23]. Selain itu terdapat Algoritma Greedy (AG) pada makalah [3] dengan kasus penjadwalan fakultas. Algoritma ini mampu mendapatkan solusi namun dengan waktu yang lama yaitu 6000s dan solusi yang dihasilkan hanya 2% dari optimal. Hal ini disebabkan data yang digunakan cukup besar.

Dari paparan di atas, setiap algoritma memiliki kelebihan masing-masing. Namun, ada beberapa hal yang menjadi *trade-off* saat algoritma diterapkan, seperti *running time* yang diharapkan dan juga kesesuaian algoritma dengan domain kasus. Kasus pada realita di lapangan juga memiliki kompleksitas yang berbeda-beda. Sebagai contohnya penjadwalan di Informatika UII masih menggunakan cara manual dengan waktu yang lama dan proses yang kompleks sehingga diperlukan optimasi untuk mengoptimalkan hasil penjadwalan mata kuliah. Kasus yang ada di Informatika UII memiliki beberapa batasan untuk melakukan penjadwalan. Dalam konteks mata kuliah, disusun secara paket dari program studi sehingga mata kuliah di paket yang sama tidak boleh tabrakan sesi dan hari pelaksanaannya. Dari sisi dosen yaitu dosen dengan lulusan S3 dapat mengampu maksimal 6 sks sedangkan dosen dengan lulusan S2 dapat mengampu 9 hingga 12 sks. Selain itu, dosen dengan jabatan struktural dapat memilih jadwal dan tidak tabrakan dengan kelas lain yang diampu. Untuk ruangan tersedia, Informatika memiliki jumlah ruang paralel 7 ruang kelas dan 3 ruang laboratorium. Untuk penggunaan laboratorium diutamakan untuk mata kuliah tertentu. Proses belajar mengajar di Informatika dilaksanakan dimulai pukul 07.00-18.00, kecuali hari Jumat perkuliahan dimulai pukul 06.45. Dalam satu minggu kegiatan belajar mengajar dilaksanakan 5 hari. Permasalahan penjadwalan ini perlu dimodelkan agar dapat diselesaikan menggunakan algoritma penjadwalan yang sesuai.

## **4. KESIMPULAN**

Proses penyusunan jadwal kuliah secara manual merupakan proses yang kompleks serta membutuhkan waktu yang lama, sehingga dilakukan pencarian metode optimasi untuk penjadwalan mata kuliah. Ditemukan 26 makalah, secara umum domain yang digunakan ialah universitas dan sekolah dengan persentase 88% dan 12% dari keseluruhan makalah. Berdasarkan hasil analisis, adanya temuan bahwa Algoritma Greedy belum mampu memperoleh solusi penjadwalan yang optimal. Hasil yang diperoleh hanya 2% dari nilai optimal. Hal ini dikarenakan jumlah data untuk pengujian yang cukup besar sehingga hasil yang diperoleh kurang maksimal dan membutuhkan waktu cukup lama. Dari hasil analisis, diketahui bahwa Simulated Annealing (SA) mampu menghasilkan solusi dengan waktu 0.481-10.102s. Namun, belum berhasil memperoleh nilai *fitness* 1. Algoritma ini dapat digunakan apabila mencari solusi optimasi dengan waktu cepat. Selain itu,

diketahui bahwa Genetic Algorithm (GA) dominan digunakan dalam beberapa tahun ke belakang karena mampu memperoleh solusi optimal dengan waktu cepat yaitu 0.964-73.461s dan nilai *fitness* 1. Oleh karena itu, algoritma ini dapat digunakan untuk menghasilkan solusi optimal penjadwalan dengan waktu tercepat dan nilai *fitness* terbaik. Adanya kajian pada makalah ini dapat memberikan rekomendasi dan memudahkan pemilihan algoritma penjadwalan yang sesuai dengan kriteria yang ditetapkan oleh pengguna.

#### DAFTAR PUSTAKA

- [1] Y. Sari, M. Alkaff, E. S. Wijaya, S. Soraya, and D. P. Kartikasari, "Optimasi Penjadwalan Mata Kuliah Menggunakan Metode Algoritma Genetika dengan Teknik Tournament Selection," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 6, no. 1, p. 85, 2019.
- [2] T. Sunarni, R. Bendi, and A. Alfian, "Penerapan Teknik Pewarnaan Simpul Graf Pada Permasalahan Penjadwalan Kuliah," *Pros. Ritektra*, vol. 8, no. 1, pp. 84–91, 2018.
- [3] A. Lemos, F. S. Melo, P. T. Monteiro, and I. Lynce, "Room usage optimization in timetabling: A case study at Universidade de Lisboa," *Oper. Res. Perspect.*, vol. 6, no. July 2018, p. 100092, 2019.
- [4] A. Muklason, R. G. Irianti, and A. Marom, "Automated course timetabling optimization using tabu-variable neighborhood search based hyper-heuristic algorithm," *Procedia Comput. Sci.*, vol. 161, pp. 656–664, 2019.
- [5] A. S. Laswi, "Perbandingan Algoritma Fitness of Spring dan Algoritma Tabu Search pada Kasus Penjadwalan Perkuliahan," *Ilk. J. Ilm.*, vol. 12, no. 1, pp. 39–46, 2020.
- [6] N. I. Kurniati, A. Rahmatulloh, and D. Rahmawati, "Perbandingan Performa Algoritma Koloni Semut Dengan Algoritma Genetika – Tabu Search Dalam Penjadwalan Kuliah," *Comput. Eng. Sci. Syst. J.*, vol. 4, no. 1, p. 17, 2019.
- [7] Z. Houhamdi, B. Athamena, R. Abuzaineddin, and M. Muhairat, "A multi-agent system for course timetable generation," *TEM J.*, vol. 8, no. 1, pp. 211–221, 2019.
- [8] D. D. P. Sari, W. F. Mahmudy, and D. E. Ratnawati, "Optimasi Penjadwalan Mata Pelajaran Menggunakan Algoritma Genetika ( Studi Kasus : SMPN 1 Gondang Mojokerto )," *Univ. Brawijaya*, vol. 5, no. 13, pp. 1–9, 2015.
- [9] H. Setiawan, L. H. Hanafi, and K. R. Prilianti, "Implementasi Algoritma Kunang-Kunang Untuk Penjadwalan Mata Kuliah di Universitas Ma Chung," *J. Buana Inform.*, vol. 6, no. 4, 2015.
- [10] Theresia Sunarni, R. Kristoforus Jawa Bendi, Achmad Alfian, "Optimasi Penjadwalan Mata Kuliah Menggunakan Pewarnaan Graf," *Pros. SNTI dan SATELIT*, no. January, pp. E48-53, 2017.
- [11] I. Kusmarna, L. K. Wardhani, and M. Safrizal, "Aplikasi Penjadwalan Mata Kuliah Menggunakan Algoritma Particle Swarm Optimization (Pso)," *J. Tek. Inform.*, vol. 8, no. 2, pp. 1–8, 2015.
- [12] A. H. Climbing, "ALGORITMA HILL CLIMBING LECTURE TIMETABLING USING HILL CLIMBING," vol. 1, pp. 98–105, 2017.
- [13] S. Ni Luh Gede Pivin, S. I Made, and D. Suta, "Penerapan Algoritma Genetika Untuk Penjadwalan Mata Pelajaran," *J. Appl. Intell. Syst.*, vol. 1, no. 3, pp. 220–233, 2016.
- [14] E. Suhartono, "Optimasi Penjadwalan Mata Kuliah dengan Algoritma Genetika ( Studi Kasus di AMIK JTC Semarang )," *Infokam*, vol. 2, pp. 132–146, 2015.
- [15] E. A. Abdelhalim and G. A. El Khayat, "A Utilization-based Genetic Algorithm for Solving the University Timetabling Problem (UGA)," *Alexandria Eng. J.*, vol. 55, no. 2, pp. 1395–

- 1409, 2016.
- [16] D. Wahyuningsih, "Rancangan Sistem Penjadwalan Akademik Menggunakan Algoritma Max Min Ant System (Studi Kasus: STMIK Atma Luhur Pangkalpinang)," *J. Edukasi dan Penelit. Inform.*, vol. 1, no. 2, pp. 1–4, 2015.
- [17] A. Josi, "Implementasi Algoritma Genetika Pada Aplikasi Penjadwalan Perkuliahan Berbasis Web Dengan Mengadopsi Model Waterfall," *J. Inform. J. Pengemb. IT*, vol. 02, no. 02, pp. 77–83, 2017.
- [18] D. W. Nugraha, A. Y. E. Dodu, and M. F. B. Paloloang, "Sistem Penjadwalan Perkuliahan Menggunakan Algoritma Particle Swarm Optimization Pada Jurusan Teknologi Informasi Fakultas Teknik Universitas Tadulako," *Sci. Comput. Sci. Informatics J.*, vol. 2, no. 1, p. 1, 2019.
- [19] M. Assi, B. Halawi, and R. A. Haraty, "Genetic Algorithm Analysis using the Graph Coloring Method for Solving the University Timetable Problem," *Procedia Comput. Sci.*, vol. 126, pp. 899–906, 2018.
- [20] N. G. A. P. H. Saptarini, P. I. Ciptayani, N. W. Wisswani, I. W. Suasnawa, and N. E. Indrayana, "Comparing Selection Method in Course Scheduling Using Genetic Algorithm," vol. 1, no. Icst, pp. 574–578, 2018.
- [21] S. Mauluddin, I. Ikbal, and A. Nursikuwagus, "Complexity and performance comparison of genetic algorithm and ant colony for best solution timetable class," *J. Eng. Sci. Technol.*, vol. 15, no. 1, pp. 276–290, 2020.
- [22] F. Broberg, E. Eriksson, and F. Broberg, "Comparing MAX-MIN and Rank-based Ant Colony Optimization Algorithms for solving the University Course Timetabling Problem Royal Institute of Technology Timetabling Problem Examiner : Orjan," 2018.
- [23] A. Salman and R. Hanna, "A Comparative Study between GeneticAlgorithm, Simulated Annealing and a Hybrid Algorithm for solving a University Course Timetabling Problem," *Degree Proj. Technol.*, 2018.
- [24] I. Andersson and C. P. Svensson, "Comparing Two-Phase Hybrid Metaheuristics for the University Course Timetabling Problem," 2019.
- [25] D. R. Ramdania, M. Irfan, F. Alfarisi, and D. Nuraiman, "Comparison of genetic algorithms and Particle Swarm Optimization (PSO) algorithms in course scheduling," *J. Phys. Conf. Ser.*, vol. 1402, no. 2, 2019.
- [26] I. A. Ashari, M. A. Muslim, and A. Alamsyah, "Comparison Performance of Genetic Algorithm and Ant Colony Optimization in Course Scheduling Optimizing," *Sci. J. Informatics*, vol. 3, no. 2, pp. 149–158, 2016.