



Penerapan Metode *Hybrid Genetic Algorithm (GA)* dan *Pattern Search (PS)* untuk Penjadwalan Mata Kuliah Universitas

Fani Puspitasari¹, Parwadi Moengin²

^{1,2} Fakultas Teknologi Industri, Jurusan Teknik Industri, Universitas Trisakti
Jl. Kyai Tapa No. 1, Jakarta 11440
E-mail: fani.puspitasari@trisakti.ac.id, parwadi@trisakti.ac.id

Abstract

The problem of university course scheduling is a complicated job to do because of the many constraints that must be considered, such as the number of courses, the number of rooms available, the number of students, lecturer preferences, and time slots. The more courses that will be scheduled, the scheduling problem becomes more complex to solve. Therefore, it is necessary to set an automatic course schedule based on optimization method. The aim of this research is to gain an optimal solution in the form of schedule in order to decrease the number of clashed courses, optimize room utilization and consider the preferences of lecturer-course. In this research, a hybridization method of Genetic Algorithm (GA) and Pattern Search (PS) is investigated for solving university course scheduling problems. The main algorithm is GA to find the global optimum solution, while the PS algorithm is used to find the local optimum solution that is difficult to obtain by the GA method. The simulation results with 93 courses show that the Hybrid GA-PS method works better than does the GA method without hybrid, as evidenced by the better fitness value of the hybrid GA-PS method which is -3528.62 and 99.24% of the solutions achieved. While the GA method without hybrid is only able to reach a solution of around 65% and has an average fitness value of -3100.76.

Keywords: *course scheduling, optimization, hybrid genetic algorithm methods, pattern search method*

Abstrak

Masalah penjadwalan mata kuliah universitas adalah sebuah pekerjaan yang rumit untuk dikerjakan karena banyaknya kendala yang harus diperhatikan seperti jumlah mata kuliah, jumlah ruangan yang tersedia, jumlah mahasiswa, preferensi dosen dan slot waktu. Semakin banyak jumlah mata kuliah yang akan dijadwalkan maka semakin rumit penyelesaian masalah penjadwalan tersebut, oleh karena itu diperlukan adanya pengaturan jadwal mata kuliah otomatis berdasarkan metode optimasi. Tujuan dari penelitian ini adalah untuk mendapatkan solusi yang optimal dalam bentuk jadwal untuk mengurangi jumlah mata kuliah yang bentrok, mengoptimalkan pemanfaatan ruang dan mempertimbangkan preferensi dosen. Pada penelitian ini, metode yang digunakan untuk menyelesaikan masalah penjadwalan mata kuliah universitas adalah metode *hybrid Genetic Algorithm (GA)* dan *Pattern Search (PS)*. Algoritma utama yang digunakan adalah GA untuk menemukan solusi optimum global sedangkan algoritma PS digunakan untuk mencari solusi optimum lokal yang sulit didapatkan oleh metode GA. Hasil simulasi dari 93 mata kuliah menunjukkan bahwa metode Hybrid GA-PS bekerja lebih baik dibandingkan metode GA tanpa *hybrid*, terbukti dari nilai rata-rata *fitness value* metode *hybrid* GA-PS yang lebih baik yaitu -3528,62 dan 99,24% solusi tercapai. Sedangkan metode GA tanpa *hybrid* hanya mampu mencapai solusi sekitar 65% dan memiliki nilai rata-rata *fitness value* -3100,76.

Kata kunci: *penjadwalan mata kuliah, optimasi, metode hybrid genetic algorithm, metode pattern search*

Pendahuluan

Penjadwalan mata kuliah merupakan kegiatan yang rutin dilakukan oleh setiap

institusi pendidikan tingkat universitas di setiap pergantian semester. Tujuan dari penjadwalan mata kuliah adalah agar proses belajar mengajar dapat berjalan dengan baik, tidak

hanya bagi mahasiswa yang mengambil mata kuliah tersebut, namun juga bagi dosen yang mengajar. Masalah penjadwalan tersebut menjadi pekerjaan yang rumit dikerjakan karena terdapat kendala yang harus diperhatikan. Kendala tersebut antara lain jumlah mahasiswa, jumlah ruangan yang tersedia, keterbatasan kapasitas ruangan dan keterbatasan jam mengajar dosen. Semakin banyak jumlah mata kuliah yang akan dijadwalkan maka semakin rumit penyelesaian masalah penjadwalan tersebut (Wati & Rochman, 2013).

Selama ini, penjadwalan mata kuliah di banyaknya universitas masih dilakukan secara manual dan hal ini memungkinkan terjadinya *human error* sehingga menyebabkan adanya bentrok antar mata kuliah. Oleh karena itu, diperlukan adanya pengaturan jadwal mata kuliah otomatis dengan pendekatan metode optimasi sehingga dapat memudahkan penjadwalan yang bersifat kompleks dan dapat meng-cover banyaknya *constraints* dalam melakukan penjadwalan.

Penelitian terkait penjadwalan mata kuliah dengan menggunakan metode *Genetic Algorithm* (GA) sudah banyak dilakukan, Priambodo, Nhita, & Aditsania (2016) dan Anamisa, Djunaidy (2014) menerapkan metode *hybrid* Algoritma Genetika dan Algoritma Koloni Semut. Metode *hybrid* dua metode tersebut menghasilkan solusi yang lebih baik dibandingkan dengan tanpa *hybrid*. Wen-jing (2018) menerapkan metode *improved adaptive genetic algorithm*. Hasil penelitian menunjukkan bahwa metode *improved adaptive genetic algorithm* konvergen lebih cepat ke solusi dibandingkan metode GA biasa. Balcilar (2020) menerapkan metode GA dengan menambahkan preferensi dosen ke dalam formulasi penjadwalannya. Jat (2012) menerapkan metode *Local search* dan *Guided Search* pada metode GA untuk menghasilkan solusi dengan kualitas yang lebih baik.

Pada penelitian ini, penyelesaian penjadwalan mata kuliah dilakukan dengan metode *Hybrid Genetic Algorithm* (GA) dan *Pattern Search* (PS). Pemilihan *Genetic Algorithm* (GA) dalam penelitian ini adalah karena GA mampu menyelesaikan permasalahan yang memiliki tingkat kesulitan yang tinggi sehingga cocok untuk menyelesaikan masalah penjadwalan mata kuliah yang memiliki banyak variabel kendala

(Jat, 2012). Kelebihan lainnya dari GA adalah perhitungan matematis yang dilakukan sedikit dan operator-operator evolusi yang digunakan membuat algoritma ini sangat efektif pada pencarian global, dan memiliki fleksibilitas yang tinggi untuk di-*hybrid*-kan dengan metode pencarian lainnya agar lebih efektif (Gen & Cheng, 1997).

Namun metode GA memiliki kelemahan yaitu tidak cukup efisien ketika menemukan nilai optimal pada proses mutasi iterasi-iterasi terakhir. Hal ini dikarenakan metode GA bekerja baik hanya di ruang pencarian solusi yang luas (*global search*), sedangkan tidak cukup baik untuk ruang solusi yang terbatas (*local search*) (Basak, Sanyal, Kumar, & Goswami, 2013). Oleh karena itu, metode *Pattern Search* (PS) di-*hybrid* kan ke dalam metode GA untuk mengatasi kelemahan metode GA tersebut. Metode PS dapat bekerja baik pada pencarian solusi dalam ruang solusi yang terbatas sehingga dapat menemukan nilai *local optimum*. Hal ini menjadi kelebihan metode *Hybrid* GA-PS jika dibandingkan dengan metode *Hybrid* GA lainnya karena dengan metode PS, masalah penjadwalan yang memiliki ruang solusi yg luas dapat ditemukan solusi *local* optimumnya.

Penelitian ini bertujuan menghasilkan solusi terbaik berupa sebuah jadwal yang mampu mengoptimalkan penggunaan berbagai sumber daya pembelajaran seperti ruang kelas, dosen, dan mahasiswa. Pada penelitian ini juga penjadwalan dilakukan dengan menambahkan preferensi dosen memilih jadwal mengajar pada *time-slot*, hari, dan kelas yang diinginkan.

Metodologi

Penjadwalan Mata Kuliah

Penjadwalan mata kuliah di universitas didefinisikan sebagai masalah penugasan multi-dimensi, dimana mahasiswa dan dosen di-*assign* ke dalam penyusunan daftar perkuliahan yang dimasukkan pada slot waktu dan ruang tertentu (Carter & Laporte, 1996). Dalam penjadwalan mata kuliah, terdapat batasan atau kendala, yang terdiri dari kendala mutlak (*hard constraint*) dan kendala lunak (*soft constraint*).

Kendala mutlak (*hard constraint*) adalah kendala yang mutlak diperlukan, bila salah satu kendala tidak dipenuhi, maka perkuliahan tidak akan berjalan. Sedangkan kendala lunak (*soft constraint*) adalah kendala yang tidak mutlak

perlu terpenuhi. Namun jika kendala mutlak dan kendala lunak dipenuhi, maka dapat dikatakan solusi penjadwalan telah optimal.

Solusi penjadwalan dapat dibagi ke dalam solusi layak, solusi tidak layak dan solusi optimal. Solusi layak adalah solusi yang memenuhi semua kendala mutlak (*hard constraint*) tetapi tidak harus semua memenuhi kendala lunak (*soft constraint*). Solusi tidak layak adalah solusi yang tidak berhasil memenuhi semua kendala, baik kendala mutlak maupun kendala lunak. Solusi optimal adalah solusi yang memenuhi semua persyaratan kendala dan tidak ada pelanggaran kendala mutlak dan lunak (Jat, 2012).

Metode Genetic Algorithm (GA)

Metode *Genetic Algorithm* (GA) pertama kali diperkenalkan oleh John Holland pada tahun 1960 berdasarkan konsep teori Evolusi Darwin, lalu dikembangkan oleh David E Goldberg pada 1989. Metode GA terinspirasi dari proses seleksi genetika makhluk hidup yang diterapkan menjadi algoritma komputasi dan dapat digunakan untuk mencari solusi masalah optimasi. Konsep yang diadopsi ke dalam metode GA yaitu bahwa makhluk hidup dapat mempertahankan kelangsungan hidupnya dengan melalui proses reproduksi, mutasi dan crossover (Sivanandam & Deepa, 2008).

GA memiliki beberapa kelebihan yaitu algoritma ini hanya melakukan sedikit perhitungan matematis yang berhubungan dengan masalah yang ingin diselesaikan. Kelebihan lain adalah metode GA dapat mencari solusi optimal secara global, artinya dapat mencari nilai optimal dari keseluruhan rentang input data (Erben & Kepler, 1995). Kemudian operator-operator evolusi yang digunakan membuat algoritma ini sangat efektif pada pencarian global, dan memiliki fleksibilitas yang tinggi untuk di-*hybrid*-kan dengan metode pencarian lainnya agar lebih efektif. GA juga memiliki kekurangan yaitu adanya kemungkinan terjadinya masalah dalam melakukan identifikasi fungsi *fitness*, dan dalam mendefinisikan suatu permasalahan, terjadinya konvergensi yang prematur, banyaknya parameter yang digunakan seperti jumlah populasi (*population size*), parameter mutasi, parameter *crossover*, dan yang terakhir tidak dapat digunakan untuk permasalahan *local optimal* (Sivanandam & Deepa, 2008).

Dalam metode *Genetic Algorithm* (GA), salah satu kemungkinan solusi optimal disebut sebagai individu. Kumpulan individu membentuk populasi yang merupakan sejumlah kemungkinan solusi. Tahap pertama dalam GA adalah representasi kromosom yaitu mengetahui tipe data seperti apa yang akan diteliti untuk selanjutnya diproses ke dalam skema pengkodean. Tipe data kromosom dapat berupa nilai biner, *float*, integer, karakter, maupun kombinatorial. Kumpulan kromosom yang dibangkitkan dengan GA akan melalui proses seleksi untuk memilih induk untuk proses *crossover* dan mutasi. Beberapa metode seleksi kromosom yaitu *rank-based final assignment*, *roulette wheel selection*, *stochastic universal sampling*, *local selection*, *truncation selection* (Saini, 2017).

Selanjutnya kromosom yang berhasil lulus seleksi akan melalui proses *crossover* dan dinamakan kromosom induk. *Crossover* adalah proses pindah silang nilai gen di antara kromosom induk. Kromosom hasil *crossover* disebut sebagai *offspring*. Setelah proses *crossover*, selanjutnya adalah proses mutasi kromosom *offspring* hasil *crossover*. Proses mutasi adalah mengganti satu gen yang terpilih secara acak dengan suatu nilai baru yang didapat secara acak. Setelah diperoleh kromosom baru hasil *crossover* dan mutasi, artinya telah dilakukan satu iterasi GA atau disebut satu generasi. Kromosom-kromosom lama akan digantikan dengan kromosom baru tanpa memperhatikan nilai *fitness*-nya. Proses selanjutnya adalah mengulang iterasi hingga sudah mencapai *stopping criteria*. Setelah melewati sekian iterasi (generasi) akan didapatkan individu terbaik.

Secara sederhana, berikut adalah pseudocode dari metode GA:

1. Definisikan fungsi fitness
 2. N=stopping criteria
 3. Representasi kromosom
 4. Inisialisasi kromosom dalam populasi secara acak
 5. Evaluasi kromosom dalam populasi
 6. t=0
- while t ≤ N
- {
- lakukan
- a. Seleksi kromosom
 - b. Crossover kromosom
 - c. Mutasi kromosom
 - d. General Replacement
- }

```
t=t+1
}
```

Metode *Pattern Search*

Pattern Search adalah metode optimasi yang memperbarui iterasi saat ini dengan mengambil sampel fungsi tujuan untuk menemukan penurunan nilai pada jumlah titik yang terbatas sepanjang ruang pencarian solusi (Yudong, Shuihua, Genlin, & Zhengchao, 2013). Metode *Pattern Search* bekerja dengan baik untuk mencari solusi *local optimum*, yaitu solusi dalam suatu rentang tertentu yang terbatas. Misal f menyatakan fungsi fitness, x_0 adalah titik tebakan awal, dan Δ_0 adalah panjang langkah awal, metode PS akan terus membangkitkan iterasi sehingga $f(x_{k+1}) \leq f(x_k)$. Setiap iterasi terdiri dari "*Search-Step*" dan "*Poll-Step*". *Search-step* membangkitkan sejumlah titik *trial* yang terbatas pada sebuah *Mesh* M_k , yang berpusat di x_k dan didefinisikan oleh himpunan terbatas *pattern* Γ yang dapat menjangkau ruang solusi. *Mesh* didefinisikan sebagai berikut:

$$M_k(i) = \{x_k + \Delta_k d_i, \quad i \in \Gamma\} \quad \text{Pers. 1}$$

Poll step adalah langkah untuk melakukan *polling* pada titik di *mesh* saat ini untuk menemukan titik terbaik. *Poll step* dikatakan lengkap bila semua titik dilakukan *polling*, dan dikatakan tidak lengkap bila algoritmanya berhenti ketika sudah menemukan titik yang nilai fungsi fitnessnya lebih kecil dibanding titik saat ini. *Poll step* yang lengkap memiliki kinerja yang lebih baik namun membutuhkan waktu *computing* yang lebih lama, sedangkan *poll step* yang tidak lengkap dapat menemukan *local optimum*.

Jika *poll step* menemukan titik yang lebih baik maka x_{k+1} sama dengan titik baru, dan perbarui panjang langkah dengan mengalikan Δ_k dengan faktor ekspansi α_e sehingga $\Delta_{k+1} > \Delta_k$ karena *pattern* saat ini adalah himpunan yang cocok untuk arah *poll*. Namun jika *poll step* tidak menemukan titik yang lebih baik maka $x_{k+1} = x_k$ dan kurangi panjang langkah dengan mengalikan Δ_k dengan faktor kontraksi α_c (< 1) sehingga $\Delta_{k+1} < \Delta_k$.

Berikut adalah *pseudocode* dari metode *Pattern Search* (PS) (Yudong, Shuihua, Genlin & Zhengchao, 2013).

1. Inisialisasi: bangkitkan *pattern* Γ dan panjang langkah Δ_0
2. Ulangi langkah berikut hingga iterasi maksimum
 - a. Bangkitkan *mesh* baru $\{M_k\}$
 - b. Lakukan *Poll step* pada titik-titik di *Mesh*. Jika berhasil maka perluas *Mesh*, bila tidak maka persempit *mesh*.

Formulasi Model Matematika

Dalam melakukan penjadwalan mata kuliah dengan metode GA-PS, terdapat asumsi-asumsi yang digunakan, di antaranya:

1. Mata kuliah dengan bobot 2 dan 3 sks dilaksanakan dalam 1 kali pertemuan/minggu.
2. Pelaksanaan mata kuliah wajib, diparalelkan menjadi 3 – 6 kelas
3. Pelaksanaan kuliah dilaksanakan dari Senin-Jumat dengan 10 slot waktu/sesi dengan masing-masing slot berjumlah 45 menit. Perkuliahan dimulai pukul 08:00-08.45 hingga 16:20-17:05, dan jam istirahat pukul 12:05-13:00. Pembagian slot waktu dapat dilihat pada Tabel 1.

Tabel 1. Pembagian slot waktu dalam 1 minggu

No	Slot waktu	Senin	...	Jumat
1	08:00-08:45			
2	08:50-09:35			
.				
.				
10	16:20-17:05			

4. Mata kuliah yang berbobot 2 sks mengisi 2 slot waktu (95 menit), sedangkan mata kuliah dengan bobot 3 sks mengisi 3 slot waktu (145 menit)
5. Diasumsikan jumlah peserta kelas sudah di-*assign* sebelum dilakukan penjadwalan
6. Kendala yang digunakan dalam penjadwalan adalah kendala mutlak (*hard constraint*), di antaranya:
 - a. Dosen tidak dapat mengajar dua mata kuliah di waktu yang sama
 - b. Satu ruangan hanya untuk satu mata kuliah
 - c. Mahasiswa angkatan berapa pun tidak dapat mengambil dua mata kuliah dengan kode MK berbeda di waktu yang sama. Namun dua

mata kuliah dengan kode MK yang sama dapat berjalan dalam waktu yang sama.

- d. Jumlah peserta kelas setiap mata kuliah yang di-assign ke dalam ruangan, harus lebih kecil atau sama dengan kapasitas ruangan
- e. Setiap mata kuliah harus berjalan secara kontinu tanpa ada *break*. Artinya mata kuliah dengan bobot 2 sks, harus mengisi 2 slot waktu secara kontinu, sedangkan untuk mata kuliah dengan bobot 3 sks, harus mengisi 3 slot waktu secara kontinu.

Fungsi *fitness* dari masalah penjadwalan yang dipakai dalam penelitian ini, yaitu memaksimalkan preferensi dosen mengajar pada jam, hari dan ruangan yang diinginkan, memaksimalkan kapasitas ruangan untuk setiap mata kuliah dan meminimumkan mata kuliah yang di-assign pada waktu yang tidak diinginkan (Balcilar, 2020). Selanjutnya, fungsi *fitness* dan kendala mutlak dimodelkan ke dalam persamaan matematika untuk memudahkan pengkodean ke dalam program GA-PS. Misal:

- n : Jumlah mata kuliah (termasuk yang diparalelkan)
- R : Himpunan bilangan real
- L : Himpunan dosen
- D : Himpunan hari (direpresentasikan ke dalam $\{1,2,3,4,5\}$)
- T : Himpunan *time slot* per hari (direpresentasikan ke dalam $\{1,2,\dots,10\}$)
- C : Himpunan ruangan (direpresentasikan ke dalam $\{1,2,\dots,12\}$)
- Y : Himpunan tahun angkatan mahasiswa (direpresentasikan ke dalam $\{1,2,3,4\}$)
- M : Himpunan kode mata kuliah (tipe data: char)
- S : Himpunan *id* mata kuliah (direpresentasikan ke dalam $\{1, \dots, n\}$)
- $Y_1, Y_2, \dots, Y_{|S|} \in Y$: Tahun angkatan mahasiswa
- $M_1, M_2, \dots, M_{|S|} \in M$: Kode mata kuliah
- $L_1, L_2, \dots, L_{|S|} \in L$: Dosen untuk *id* mata kuliah ke- s , $s \in S$
- $A_1, A_2, \dots, A_{|S|} \in R$: Jumlah peserta kelas
- $B_1, B_2, \dots, B_{|S|} \in R$: Total jam pelajaran

- $Cp_1, Cp_2, \dots, Cp_{|S|} \in C$: Ruang yang diinginkan dosen untuk *id* mata kuliah ke- s , $s \in S$
- $Dp_1, Dp_2, \dots, Dp_{|S|} \in D$: Hari yang diinginkan dosen untuk *id* mata kuliah ke- s , $s \in S$
- $Tp_1, Tp_2, \dots, Tp_{|S|} \in T$: Slot time yang diinginkan dosen untuk *id* mata kuliah ke- s , $s \in S$
- $Ep_1, Ep_2, \dots, Ep_{|S|} \in R$: *Gain* yang diperoleh mata kuliah bila di-assign pada ruangan, hari dan slot waktu yang diinginkan dosen
- $P_{d,t} \in R$: *Penalty* yang diperoleh mata kuliah bila di-assign pada hari ke- d dan slot waktu ke- t
- $Ct_1, Ct_2, \dots, Ct_{|C|} \in C$: Kapasitas ruangan ke- c , $c \in C$

Definisikan $X_{d,t,c,s} \in \{0,1\}$, dimana $X_{d,t,c,s} = 1$ jika *id* mata kuliah s di assign pada hari d , slot waktu t , dan ruangan c , dan $X_{d,t,c,s} = 0$ sebaliknya. Sehingga masalah penjadwalan mata kuliah dapat dimodelkan ke dalam persamaan matematis berikut:

$$\text{Min } Z = - \sum_{i=1}^{|S|} A_i \cdot Ep_i \cdot X_{Dp_i, Tp_i, Cp_i, i} - \sum_{d=1}^{|D|} \sum_{t=1}^{|T|} \sum_{c=1}^{|C|} \sum_{s=1}^{|S|} (Ct_c - A_s) \cdot X_{d,t,c,s} + \sum_{d=1}^{|D|} \sum_{t=1}^{|T|} \sum_{c=1}^{|C|} \sum_{s=1}^{|S|} P_{d,t} \cdot A_s \cdot X_{d,t,c,s} \quad (1)$$

Terhadap kendala mutlak :

- (2) $\sum_{c=1}^{|C|} \sum_{s=1}^{|S|} X_{d,t,c,s} = 1$, jika $L_s = L_l$, $\{\forall s \in S, \forall l \in L, \forall d \in D, \forall t \in T\}$
- (3) $\sum_{s=1}^{|S|} X_{d,t,c,s} = 1$, $\{\forall d \in D, \forall t \in T, \forall c \in C\}$
- (4) $\sum_{c=1}^{|C|} \sum_{s=1}^{|S|} X_{d,t,c,s} = 1$, jika M_i unik dan $Y_s = Y_y$, $\{\forall s \in S, \forall y \in Y, \forall d \in D, \forall t \in T\}$
- (5) $X_{d,t,c,s} \cdot A_s \leq Ct_c$, $\{\forall d \in D, \forall t \in T, \forall c \in C, \forall s \in S\}$
- (6) $\sum_{o=j+1}^{j+B_s-1} X_{d,t,c,s} = B_i - 1$,

jika $\left(\begin{matrix} j = \text{slot waktu dimulainya mata kuliah } s \\ d = \text{hari mata kuliah } s \\ c = \text{ruangan mata kuliah } s \end{matrix} \right)$,

$\{\forall s \in S\}$

Tabel 2. Daftar mata kuliah

Id MK	Tahun	Kelas Paralel	Kode MK	Mata Kuliah	sks	Dosen	Mhs	Preferensi Ruang (1,..12)	Preferensi Hari (1,..5)	Preferensi Time Slot (1,..10)	Gain (0,1]
1	1	A	IUM341	MK 1	3	AA	35	1	1	2	0.5
2	1	B	IUM341	MK 2	3	BB	23	3	4		0,3
3	1	C	IUM341	MK 3	3	CC	40		5	9	0,9
dan seterusnya											

Pengolahan Data

Data yang digunakan sebagai studi kasus untuk penelitian ini adalah data mata kuliah semester Genap 2019/2020 Jurusan Teknik Industri Universitas Trisakti. Jumlah mata kuliah yang perlu dijadwalkan adalah 93, dengan 12 jumlah ruangan dan 10 slot waktu dalam sehari. Tabel 2 adalah input data penjadwalan mata kuliah untuk pengkodean dengan metode GA-PS. Sedangkan Tabel 3 adalah data nama ruangan dan kapasitasnya.

Pada Tabel 2 terdapat kolom untuk mengisi preferensi ruangan, hari dan *time slot* yang diinginkan oleh dosen. Misal untuk mata kuliah 1, pengajar A memilih ruang 1, hari ke-1 (Senin) dan time slot 2 (mulai pukul 08.50), maka *gain* mata kuliah ini bila di-*assign* sesuai dengan preferensi akan bernilai 0.5. Nilai *Gain* bernilai random dalam selang (0,1], artinya nilai *gain* tidak boleh 0 bila seorang dosen memiliki preferensi (baik ruangan/hari/time slot), dan nilai maksimum *gain* adalah 1.

Selanjutnya, Tabel 4 dijadikan input data untuk menjelaskan nilai *penalty* untuk setiap slot waktu. Misal pada jam 08:00-08:45 diberikan nilai *penalty* 0,5 bila mata kuliah di-*assign* pada slot waktu tersebut, dan misal pada jam 16:20-17:05 diberikan nilai *penalty* 1 yang artinya sebisa mungkin mata kuliah tidak di-*assign* pada slot waktu terakhir tersebut.

Tabel 3. Data ruangan dan kapasitasnya

Kode Ruang	Kapasitas
615	25
616	42
dan seterusnya	

Tabel 4. Data *penalty* slot waktu

Slot Waktu	Senin	...	Jumat
08:00-08:45	0,5		0,5
08:50-09:35	0		0
...			
16:20-17:05	1		1

Metode Hybrid GA-PS

Pada penelitian ini, sebelum menerapkan metode *Hybrid GA-PS*, jumlah kemungkinan dari seluruh mata kuliah untuk di-*assign* pada hari, slot waktu dan ruangan yang memenuhi seluruh *hard constraint* akan dihitung terlebih dahulu. Selanjutnya metode GA-PS akan diterapkan untuk memilih peluang *assignment* terbaik yang akan di-*assign* ke mata kuliah tersebut. Gambar 1 menunjukkan *flowchart* dari metode *hybrid GA-PS* dalam penelitian ini.

Berikut adalah penjelasan dari penyusunan algoritma GA-PS untuk menyelesaikan masalah penjadwalan mata kuliah.

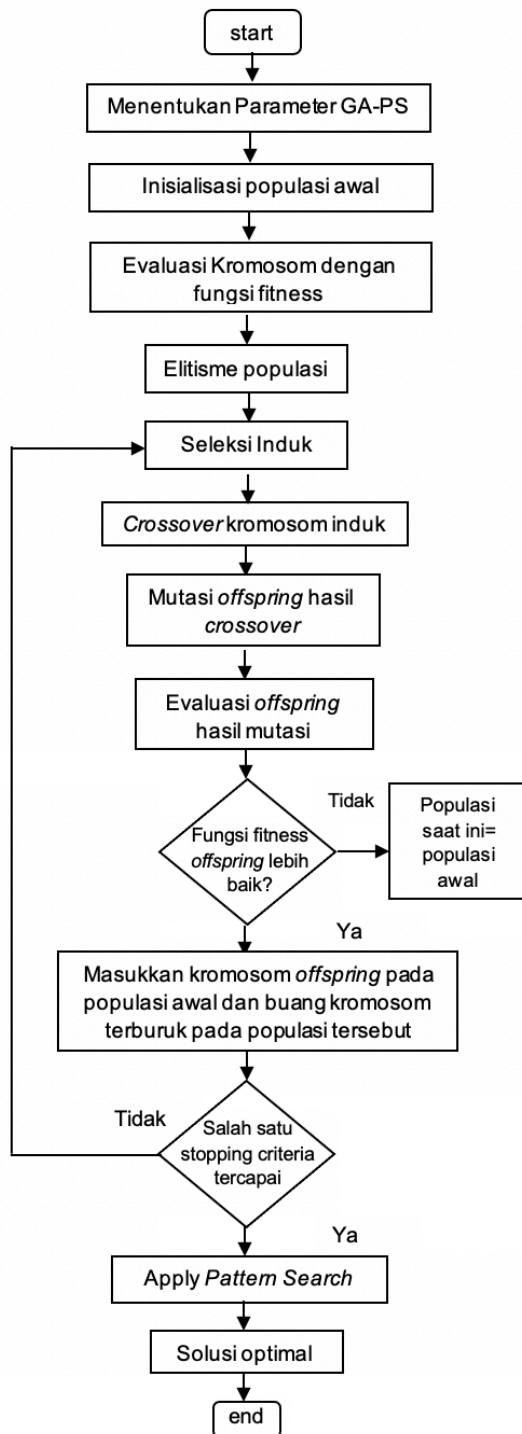
1. Menentukan parameter GA-PS

Pada penelitian ini, parameter metode *hybrid GA-PS* yang dipakai sesuai pada Tabel 5.

Tabel 5. Parameter metode GA-PS

Parameter GA	Nilai
Ukuran Populasi	500
Iterasi Maksimum	250
Crossover	Scattered
Probabilitas Mutasi (Pm)	0,01
Rate Elitisme	5%
Stall Generation	50
Selection	Stochastic Uniform
Parameter PS	Nilai
Initial Mesh Size	1
Max Function Evaluations	2000*numberOfVariables
Max Iterations	100*numberOfVariables
Mesh Contraction Factor	0,5
Mesh Expansion Factor	2

Dasar pemilihan parameter seperti tabel 5 adalah karena simulasi yang dilakukan sudah mencoba beberapa kombinasi nilai parameter, seperti mengubah nilai *pm* (0.01; 0.05; 0.1;



Gambar 1. Flowchart metode hybrid GA-PS

0,15) dan mengubah rate elitisme (5%; 10%), diperoleh kombinasi parameter yang memberikan solusi terbaik untuk masalah penjadwalan mata kuliah adalah nilai pm 0.01 dan rate elitisme 5%.

2. Representasi kromosom

Dalam penelitian ini, untuk mendekati masalah nyata sistem penjadwalan mata kuliah, tipe kromosom yang digunakan adalah suatu variabel yang mencakup ruang kuliah, *time slot*,

dosen, mata kuliah, definisikan $X_{d,t,c,s} \in \{0,1\}$, dimana $X_{d,t,c,s} = 1$ jika dan hanya jika id mata kuliah s di-assign pada hari d , slot waktu t , dan ruangan c , dan $X_{d,t,c,s} = 0$ sebaliknya jika dan hanya jika id mata kuliah s tidak di-assign pada hari d , slot waktu t , dan ruangan c .

3. Inisialisasi populasi

Matriks mata kuliah, matriks bentrok, matriks preferensi dosen, matriks slot waktu, matriks ruangan, dan matriks *assignment* awal digunakan sebagai input untuk menghitung jumlah kemungkinan dari seluruh mata kuliah untuk di-assign pada hari, slot waktu dan ruangan yang memenuhi seluruh *hard constraint*.

Matriks bentrok adalah $M_{n \times n}$ dimana n adalah jumlah mata kuliah yang harus dijadwalkan. Elemen matriks bentrok bernilai biner, bernilai 1 bila 'bentrok' dan bernilai 0 bila 'tidak bentrok'. Matriks *Assignment* $A_{|T| \times |C|}$ adalah representasi dari sebuah solusi metode GA-PS yaitu berupa jadwal yang optimal, dengan $|T|$ adalah slot waktu dalam seminggu dan $|C|$ adalah ruangan. Elemen dari matriks tersebut adalah id mata kuliah bila di-assign pada slot waktu i dan ruangan j , sedangkan bilangan 0 artinya pada slot waktu i dan ruangan j tidak di-assign mata kuliah. Sedangkan matriks *assignment* awal seluruh elemennya bernilai 0. Tabel 6 merupakan contoh bentuk representasi jadwal ke dalam bentuk matriks.

Tabel 6. Contoh representasi jadwal dalam bentuk matriks *assignment*

Waktu/Ruang	1	...	12
1	90		0
2	55		80
...			
...			
50	0		5

Setelah mendefinisikan matriks-matriks input tersebut, hitung seluruh kemungkinan *assign* untuk seluruh mata kuliah. Misal untuk mata kuliah MK 1 terdapat 100 kemungkinan *assign*, MK 2 terdapat 180 kemungkinan *assign*, dan seterusnya. Selanjutnya bangkitkan populasi awal dengan *uniform random*, kromosom awal ini bernilai [0 1] yang terdistribusi secara uniform. Setelah itu, kalikan setiap nilai kromosom awal tersebut pada jumlah

kemungkinan *assign* seluruh mata kuliah. Berikut adalah contoh kromosom awal:

gen 1	gen 93
0.199			0.739

Misal untuk mata kuliah pertama yaitu MK 1 terdapat 100 kemungkinan *assign*, karena nilai gen 1 bernilai 0.199 maka kita harus memilih kemungkinan *assign* ke $100 \times 0.199 = 19.9 \approx 20$, misal untuk mata kuliah ke 93 yaitu MK 93 terdapat 350 kemungkinan *assign*, karena nilai gen 93 bernilai 0.739 maka kita harus memilih kemungkinan *assign* ke $350 \times 0.739 = 258.65 \approx 259$, dan seterusnya untuk semua mata kuliah dilakukan perhitungan jumlah kemungkinan *assign*nya pada nilai gen pada kromosom. Bila nilai gen bernilai nol maka pilih kemungkinan *assign* yang pertama dari list, sedangkan bila gen bernilai 1 maka pilih kemungkinan *assign* yang paling terakhir dari list.

4. Evaluasi populasi awal

Setelah semua mata kuliah dihitung kemungkinan *assign*nya berdasarkan nilai kromosom populasi awal, selanjutnya adalah menghitung nilai fitness dari setiap kemungkinan *assign* yang dipilih tersebut dengan membuat nilai kromosom yang berkaitan bernilai 1 bila sesuai dengan preferensi, sedangkan yang tidak sesuai preferensi bernilai 0. Evaluasi populasi dihitung dengan fungsi fitness berdasarkan persamaan 1

5. Elitisme populasi

Proses elitisme dilakukan dengan meng-*copy* kromosom-kromosom terbaik agar tidak hilang saat proses *crossover* dan mutasi. *Rate* elitism yang dipakai sebesar 5% artinya 5 kromosom terbaik dipilih dari total N jumlah kromosom, dan di-*copy* ke dalam populasi.

6. Seleksi

Selanjutnya dilakukan seleksi induk yaitu dengan memilih dua kromosom dengan metode *default stochastic uniform*. Dua kromosom inilah yang akan menjadi induk untuk melakukan proses *crossover*.

7. *Crossover* dan mutasi

Crossover atau pindah silang dalam penelitian ini dilakukan dengan *crossover*

scattered. Hasil *crossover* gen-gen dari induk ini akan diturunkan pada keturunannya (*offspring*). Tahapan selanjutnya adalah melakukan proses mutasi untuk *offspring* yang diperoleh dari proses *crossover*. Proses mutasi adalah mengganti nilai gen dari suatu kromosom dengan suatu nilai yang diperoleh secara random. Dalam penelitian ini mutasi dilakukan dengan *uniform mutation* dengan nilai $pm=0.01$ artinya 1% gen *offspring* mengalami mutasi.

8. Evaluasi kromosom offspring

Setelah proses mutasi dilakukan, bila nilai fitness kromosom *offspring* hasil mutasi lebih baik, maka akan masuk pada populasi awal menggantikan kromosom terburuk dalam populasi tersebut, sedangkan bila kromosom hasil mutasi buruk, maka akan dibuang dan populasi saat ini tetaplah sama dengan populasi awal.

9. Pemeriksaan *stopping criteria*

Proses dari seleksi induk hingga mutasi disebut sebagai satu generasi, selanjutnya populasi akan terus melanjutkan generasi hingga salah satu *stopping criteria* tercapai, yaitu ketika jumlah iterasinya sudah maksimum atau ketika *stall generation* sudah tercapai.

10. *Apply* metode *Pattern Search* (PS)

Metode GA dapat dengan cepat mencapai nilai minimum lokal, tetapi membutuhkan banyak evaluasi fungsi untuk mencapai konvergensi. Agar mempercepat proses solusi tersebut, maka setelah metode GA menyelesaikan beberapa generasi untuk mendekati titik optimal, selanjutnya gunakan solusi GA sebagai titik awal untuk metode *Pattern Search* untuk melakukan pencarian lokal yang lebih cepat dan lebih efisien. Ketika metode GA berhenti karena telah mencapai *stopping criteria* (jumlah iterasi maksimum atau *stall generation* tercapai), selanjutnya perhitungan dilakukan oleh metode PS. Setelah kedua syarat *stopping criteria* tercapai maka akan dihasilkan output berupa kromosom optimal yang akan menghasilkan nilai fitness paling baik bila dikalikan dengan kemungkinan *assign* seluruh mata kuliah.

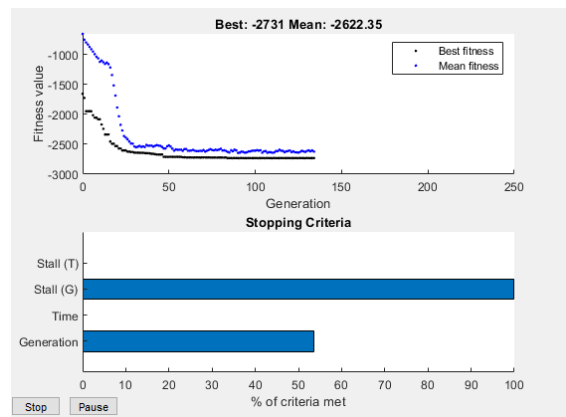
Hasil dan Pembahasan

Simulasi penyelesaian penjadwalan mata kuliah dengan metode *Hybrid GA-PS* dikodekan

pada Matlab. Pertama, metode GA biasa tanpa *hybrid* diterapkan pada permasalahan penjadwalan, selanjutnya metode Hybrid GA-PS diterapkan untuk menjadi perbandingan. Kedua metode ini dibandingkan dengan menggunakan parameter yang sama sesuai dengan Tabel 6. Setelah dilakukan sebanyak 10 kali simulasi untuk setiap metode, hasil dari *running* program dapat dilihat pada Tabel 7.

Dapat dilihat pada Tabel 7, setelah 10 kali simulasi dilakukan, metode *Hybrid* GA-PS bekerja lebih baik dibandingkan metode GA tanpa *hybrid*, hal ini terbukti dari nilai rata-ran *fitness value* metode *hybrid* GA-PS yang lebih baik yaitu -3528,62 dan 99,24% solusi tercapai. Persen solusi ini menyatakan jumlah mata kuliah yang di-*assign* pada hari, slot waktu dan ruangan, yang artinya 99,24% dari 93 jumlah mata kuliah berhasil di-*assign*, sedangkan metode GA tanpa *hybrid* hanya mampu mencapai solusi sekitar 65% dan rata-ran *fitness value* sebesar -3100,76. Fungsi *fitness* dari solusi bernilai negatif karena fungsi *fitness* dalam bentuk minimisasi, artinya bila nilai fungsi *fitness* semakin negatif maka semakin baik merepresentasikan solusi yang memenuhi setiap kendala.

Gambar 2 adalah salah satu hasil *running* metode GA tanpa *hybrid* dengan nilai *fitness* yang diperoleh adalah -2731. Hasil simulasi dari metode GA tanpa *hybrid* menghasilkan hanya 52% atau 49 dari 93 mata kuliah yang berhasil di *assign*.



Gambar 2. Hasil *running* metode GA tanpa *hybrid*

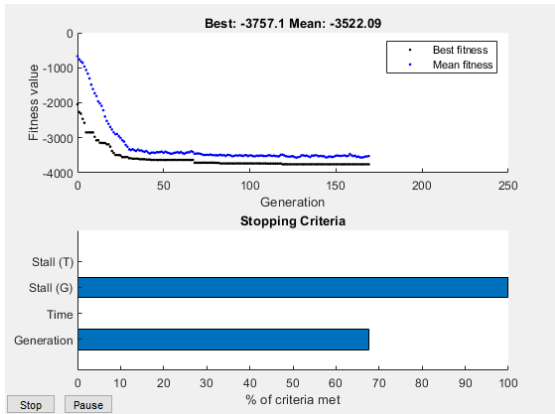
Hal ini menunjukkan bahwa metode GA tanpa *hybrid* kurang cocok untuk menyelesaikan masalah penjadwalan dengan fungsi tujuan yang multiobjektif. Ini dikarenakan metode GA biasa hanya bekerja baik untuk mencari solusi global optimum dan kurang baik mencari solusi local optimum. Cara lain agar metode GA tanpa *hybrid* menghasilkan solusi yang lebih akurat adalah dengan mengubah atau menambah *stopping criteria*, namun hal ini memerlukan banyak evaluasi fungsi *fitness* untuk mendekati global optimum, yang otomatis juga memerlukan waktu komputasi yang lebih lama.

Karena kekurangan metode GA tersebut, metode PS di *hybridkan* pada metode GA untuk dapat melakukan pencarian nilai local optimum. Metode PS diterapkan ketika metode GA berhenti karena salah satu *stopping criteria* tercapai.

Tabel 7. Tabel perbandingan metode GA tanpa *hybrid* dan metode *hybrid* GA-PS

Metode GA Tanpa Hybrid			Metode Hybrid GA-PS		
Simulasi ke-	Fitness Value	% Solusi	Simulasi ke-	Fitness Value	% Solusi
1	-3049	52,6882	1	-3757	100
2	-3860	97,8495	2	-3337,5	100
3	-2800	45,1613	3	-3539,5	98,9247
4	-3793,5	98,9247	4	-3452,5	98,9247
5	-2954,5	52,6882	5	-3552	98,9247
6	-3371,6	97,8495	6	-3630,5	97,8495
7	-2740	52,6882	7	-3658	100
8	-2864	52,6882	8	-3469	100
9	-2844	52,6882	9	-3448,6	98,9247
10	-2731	52,6882	10	-3441,6	98,9247
Rataan	-3100,76	65,5914	Rataan	-3528,62	99,2473

Gambar 3 menunjukkan hasil simulasi menggunakan metode hybrid GA-PS dengan fitness value sebesar -3757. Metode *Hybrid* GA-PS ini mampu memberikan solusi 100% atau 93 dari 93 mata kuliah berhasil di assign. Hal ini menunjukkan bahwa metode GA-PS mampu memberikan solusi yang lebih baik dan efisien.



Gambar 3. Hasil running metode Hybrid GA-PS

Matriks *Assignment* optimal dapat dilihat pada Tabel 8. Matriks *Assignment* $A_{50 \times 12}$ ini menjadi representasi dari sebuah solusi metode GA-PS yaitu berupa jadwal yang optimal. Dari matriks tersebut, 93 jumlah mata kuliah atau 100% mata kuliah berhasil di *assign* pada slot waktu dan ruangan masing-masing. Hal ini menunjukkan bahwa metode *hybrid* GA-PS telah berhasil menghasilkan output berupa jadwal yang optimal.

Tabel 9 adalah jadwal optimal hasil running simulasi metode GA-PS yang telah diubah ke dalam bentuk tabel penjadwalan. Dapat dilihat pada tabel tersebut bahwa pada slot waktu pada jam 08.00-08:45 yang diberikan nilai *penalty* 0,5, dan jam 16:20-17:05 yang diberikan nilai *penalty* 1 masih terdapat beberapa mata kuliah yang di *assign*. Begitu pun dengan masih terdapat beberapa mata kuliah yang di *assign* tidak sesuai dengan preferensi dosen. Faktor yang menyebabkan jadwal tidak ideal seperti yang diinginkan adalah karena banyaknya mata kuliah yang harus dijadwalkan, sehingga perlu melonggarkan beberapa kendala atau fungsi tujuan.

Kesimpulan

Berdasarkan hasil model penjadwalan menggunakan metode *Hybrid Genetic Algorithm Pattern Search* dengan bantuan

Tabel 8. Matriks *Assignment* Optima

2	32	55	0	34	1	0	82	31	0	0	0
2	32	55	0	34	1	0	82	31	0	0	0
2	36	55	39	0	1	0	82	0	0	0	0
4	36	37	39	35	86	0	57	0	0	56	0
4	36	37	39	35	86	0	57	0	0	56	0
4	0	37	0	35	86	0	57	58	0	56	0
0	7	8	0	40	6	0	43	58	0	87	0
0	7	8	0	40	6	0	43	58	0	87	0
0	7	8	0	40	6	0	43	0	0	87	69
0	0	0	0	0	0	0	0	0	0	0	69
47	11	45	49	46	60	0	62	0	0	0	9
47	11	45	49	46	60	0	62	0	0	0	9
0	11	0	0	50	60	0	62	52	0	89	9
13	14	0	16	50	65	66	68	52	0	89	0
13	14	15	16	50	65	66	68	52	0	89	0
13	14	15	16	0	65	66	68	0	0	0	33
0	90	15	0	0	0	0	0	0	0	0	33
0	90	17	0	0	0	0	44	71	0	70	0
0	90	17	0	0	0	0	44	71	0	70	0
0	0	17	0	0	0	0	44	0	0	0	0
0	0	0	30	0	0	0	76	0	0	91	0
0	0	3	30	0	0	0	76	0	0	91	0
0	54	3	0	0	0	0	63	0	0	91	0
0	54	3	0	0	0	0	63	0	0	83	0
0	54	0	0	0	0	0	63	0	0	83	0
0	41	61	0	0	0	0	0	0	0	83	0
0	41	61	0	59	88	0	12	0	0	42	0
0	41	61	0	59	88	0	12	0	0	42	0
0	0	0	0	59	88	0	12	0	0	42	0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	25	0	0	73	0	0	0	0
0	0	92	0	0	0	0	51	0	0	67	0
0	0	92	0	0	38	0	0	0	0	0	10
0	78	92	0	0	38	0	77	0	0	0	10
0	78	0	0	0	38	0	77	0	0	84	10
0	0	0	0	0	0	0	75	0	0	84	0
0	0	0	0	0	0	0	75	0	0	84	0
0	0	0	0	64	0	0	5	0	0	0	0
0	0	0	0	64	0	0	5	81	0	0	26
0	0	0	0	64	0	0	5	81	0	0	26
0	21	23	48	0	0	0	24	81	0	0	22
0	21	23	48	80	0	0	24	0	0	0	22
0	21	23	0	80	0	0	24	79	0	53	22
0	0	0	0	80	0	0	18	79	0	53	0
0	0	85	0	74	72	0	18	0	0	53	0
0	0	85	0	74	72	0	18	29	0	28	27
0	0	85	0	0	0	0	0	29	0	28	27

software Matlab, diperoleh kesimpulan bahwa hasil 10 kali simulasi menunjukkan bahwa metode *Hybrid* GA-PS bekerja lebih baik dibandingkan metode GA tanpa *hybrid* untuk masalah penjadwalan mata kuliah, hal ini terbukti dari nilai rata-rata *fitness value* metode *hybrid* GA-PS yang lebih minimum dan persen solusi yang mendekati 100% mata kuliah terjadwalkan. Sedangkan metode GA tanpa *hybrid* PS hanya mampu mencapai solusi sekitar 65% dan memiliki nilai rata-rata *fitness value* yang tidak lebih baik dibanding metode *Hybrid* GA-PS. Nilai fungsi *fitness* dalam penelitian ini bernilai negatif, dikarenakan fungsinya dalam bentuk minimasi, artinya semakin kecil nilai fungsi *fitness* maka semakin baik memperoleh solusi yang memenuhi fungsi tujuan dan kendala. Namun, terdapat beberapa mata kuliah yang tidak sesuai dengan preferensi dosen dan slot waktu yang dihindari. Faktor yang menyebabkan jadwal tidak ideal seperti yang diinginkan adalah karena banyaknya mata kuliah yang harus dijadwalkan, sehingga perlu melonggarkan beberapa kendala atau fungsi tujuan.

Tabel 9. Jadwal optimal hasil metode hybrid GA-PS

Slot waktu	Senin	Jumat
08:00-08:45	IUM341-B- Dr. Dadang ,M.T.-R615 IIS204-C- Prof Parwadi ,Ph.D.-R616 IIS321-A- Winnie,S.T., M.Si.-R618 IIS204-E- Ramly,--R620 IUM341-A- Fani,S.Si., M.Si.-R621 IIP326-- Dr. Ir. Iveline,M.T.-R713 IIS204-B- Linda,S.T., M.T.-R714			IUK301-B- Agung,S.T., M.T.-R620 IIG305-A- Novia,S.T., M.T.-R713
...				
15:30-16:15	IIG305-C- Indah,S.Pd., M.Si.-R616 IIG305-D- Nilla,S.T., M.T.-R618 IIP324-A- Ir. Wawan,M.T.-R620 IIG305-B- ARNES,S.T., M.Sc.-R621 IIP324-D- Dr. Ir. Tri,M.M.-R713 IIO305-- Dr. Dadan,--R716 IIU201-A- Dr. Tiena,-R717			IIO304-- Imam,-R618 IIP227-C- Linda,S.T., M.T.-R620 IIP227-A- Dr. Ir. Nora,M.T.-R621 IMG302-B- Nilla,S.T., M.T.-R713 IIG210-E- ARNES,S.T., M.Sc.-R714 IIG210-D- Winnie,S.T., M.Si.-R716 IIG210-C- Dian,S.T., M.T.-R717
16:20-17:05	IIU201-A- Dr. Tiena,-R717			IIO304-- Imam,-R618 IIG210-E- ARNES,S.T., M.Sc.-R714 IIG210-D- Winnie,S.T., M.Si.-R716 IIG210-C- Dian,S.T., M.T.-R717

Diharapkan untuk penelitian selanjutnya dapat menerapkan metode *hybrid* GA dengan metode metaheuristik lainnya yang dapat memberikan solusi yang lebih baik dan jadwal yang ideal sesuai dengan yang diinginkan.

Daftar Pustaka

- Anamisa, D. R., Djunaidy, A. (2014). Penyelesaian Penjadwalan Matakuliah Menggunakan Hibridisasi Algoritma Genetika dan Algoritma Semut. *Jurnal Ilmiah Teknologi Informasi*, 12(1), 15 – 20.
- Balcilar, M. (2020). *TimeTabling Genetic Algorithm*, [Online], Diakses dari <https://www.github.com/muhammetbalcilar/TimeTabling-GeneticAlgorithm>, [2020, Mei]
- Basak, R., Sanyal, A., Kumar, S., & Goswami, R. (2013). Comparative View of Genetic Algorithm and Pattern Search for Global Optimization. *International Journal of Engineering and Science*, 3(4), 09-12.
- Carter, M & Laporte, G. (1996). Recent developments in practical examination timetabling. *Proceedings of the 1st International Conference on the Practice and Theory of Automated Timetabling, Lecture Notes in Computer Science* 1153, 3–21.
- Erben, R. W., & Kepler, J. (1995). A Genetic Algorithm Solving a Weekly Course Timetabling Problem. *Proc. of the First Int. Conf. on the Practice and Theory of Automated Timetabling (ICPTAT)*, 21-32.
- Gen, M. & Cheng, R. (1997). *Genetic Algorithm and Engineering Design*. New York: John Wiley & Sons, Inc.
- Jat, S. N. (2012). *Genetic Algorithm for University Course Timetabling Problems*. Thesis. Department of Computer Science University of Leicester.
- Priambodo, M.A., Nhita, F., & Aditsania, A. (2016). Penjadwalan Mata Kuliah Menggunakan Metode *Hybrid* Algoritma Genetika dan Algoritma Koloni Semut. *E-Proceeding of Engineering*, 3(2), 3711-3720.
- Saini, N. (2017). Review of Selection Methods in Genetic Algorithms. *International Journal of Engineering and Computer Science ISSN:2319-7242*, 6(12), 23261-23263.
- Sivanandam, S.N., Deepa ,S.N. (2008). *Introduction to Genetic Algorithms*. Berlin: Springer-Verlag.
- Wati, D. A. N., & Rochman, Y. A. (2013). Model Penjadwalan Matakuliah Secara Otomatis Berbasis Algoritma Particle Swarm Optimization (PSO). *Jurnal Rekayasa Sistem Industri*, 2(1), 22-31.
- Wen-jing, W. (2018). Improved Adaptive Genetic Algorithm for Course Scheduling in Colleges and Universities. *International Journal of Emerging Technologies in Learning*, 13(6), 29-42.
- Yudong, Z., Shuihua, W., Genlin, J., & Zhengchao, D. (2013). Genetic Pattern Search and Its Application to Brain Image Classification. *Hindawi Publishing Corporation Mathematical Problems in Engineering*, Vol. 2013, 1-8.

Halaman ini sengaja dikosongkan.
This page is intentionally left blank.