

Algoritma *Ant Colony Optimization* untuk Optimasi Penjadwalan Mata Kuliah

Ant Colony Optimization Algorithm for Lecturing Schedule Optimization

Meiton Boru*

Program Studi Ilmu Komputer Fakultas Sains dan Teknik Universitas Nusa Cendana
Jl. Adisucipto, Lasiana, Kelapa Lima, Kupang, Nusa Tenggara Timur 85228.

*Email : meitonboru@staf.undana.ac.id

Abstrak – Tujuan dari penelitian ini adalah mengoptimalkan penjadwalan mata kuliah menggunakan algoritma *Ant Colony Optimization* (ACO). Metode yang dikembangkan akan mencari solusi optimal dari penjadwalan mata kuliah dimana terdapat beberapa batasan yang harus dipertimbangkan. Batasan-batasan tersebut diantaranya adalah seorang dosen atau suatu kelas hanya boleh dijadwalkan kuliah maksimal dua kali secara berturut-turut. Batasan lainnya adalah dua angkatan yang berbeda satu tahun, tidak boleh dijadwalkan pada waktu yang sama. Batasan ketiga adalah jangan sampai ada dosen atau kelas yang melakukan perkuliahan dengan frekuensi terlalu tinggi pada suatu hari. Dan batasan keempat adalah waktu alternatif dimana dosen dapat mengajar akan terbatas dikarenakan adanya aktivitas lain yang harus dilakukan oleh dosen tersebut. Agar kasus penjadwalan mata kuliah dapat diselesaikan menggunakan algoritma ACO, maka dibuat suatu graph dimana setiap nodenya adalah nama mata kuliah yang harus dijadwalkan. Jalur yang dibuat oleh semut dari node awal ke node akhir akan berisi urutan mata kuliah yang harus dilaksanakan pada satu minggunya. Berdasarkan hasil pengujian, algoritma ACO telah berhasil melakukan penjadwalan mata kuliah yang melibatkan 38 mata kuliah, 4 angkatan kelas, 6 lokasi perkuliahan dan 12 dosen pengampu mata kuliah. Solusi penjadwalan yang diperoleh memiliki nilai fitness 0,0092. Dimana tidak terdapat dosen yang memiliki frekuensi mengajar tinggi pada suatu hari, tetapi terdapat 12 jadwal kuliah yang menyebabkan suatu kelas mengikuti frekuensi perkuliahan yang tinggi. Serta terdapat 4 mata kuliah yang dijadwalkan berdekatan. Hasil penjadwalan dapat memiliki nilai fitness yang baik karena algoritma ACO menerapkan prinsip “pembelajaran dan perbaikan dari hasil sebelumnya”. Dapat disimpulkan bahwa algoritma ACO telah berhasil digunakan untuk mengoptimalkan penjadwalan mata kuliah.

Kata Kunci : *Ant colony optimization*, fitness, optimasi, penjadwalan

Abstract - *The purpose of this study is to optimize the the lecturing schedule using the Ant Colony Optimization (ACO) algorithm. The method developed will find an optimal solution of lecturing schedule where there are some limitations that must be considered. These limitations include being a lecturer or a class that can only be scheduled to lecture a maximum of two times in a row. Another limitation is that two adjacent level may not be scheduled at the same time. The third limitation is that there should be no lecturers or classes that conduct lectures with too high a frequency one day. And the fourth limitation is the alternative time where lecturers can teach will be limited due to other activities that must be carried out by the lecturer. So that the course scheduling case can be solved using the ACO algorithm, a graph is made where each node is the name of the course that must be scheduled. The path created by the ants from the initial node to the end node will contain the order of courses that must be carried out one week. Based on the test results, the ACO algorithm has succeeded in scheduling courses involving 38 subjects, 4 class forces, 6 recovery locations and 12 lecturers supporting the courses. The scheduling solution obtained has a fitness value of 0.0092. Where there are no lecturers who have a high teaching frequency one day, but there are 12 class schedules that cause a class to follow a high frequency of lectures. And there are 4 courses scheduled to be close together. The results of scheduling can have a good fitness value because the ACO algorithm applies the principle of "learning and improvement from previous results". It can be concluded that the ACO algorithm has been successfully used to optimize course scheduling.*

Keywords : *Ant colony optimization*, fitness, optimization, scheduling

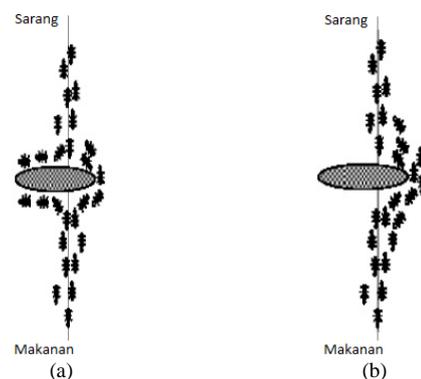
I. PENDAHULUAN

Penjadwalan kegiatan perkuliahan merupakan pekerjaan yang tidak sederhana. Terdapat beberapa batasan yang harus dipertimbangkan, diantaranya adalah sebagai berikut [1].

1. Tidak boleh ada mata kuliah pada suatu angkatan yang dijadwalkan beririsan dengan mata kuliah lain dari angkatan yang berbeda setahun di atasnya atau di bawahnya. Hal ini dikarenakan sering adanya mahasiswa yang mengulang mengambil mata kuliah di angkatan bawah atau mengambil mata kuliah di angkatan atas
2. Alternatif waktu kuliah yang bisa dilakukan oleh seorang dosen bisa jadi terbatas. Dosen sering memiliki aktivitas selain mengajar yang harus dilakukannya.
3. Distribusi jadwal mengajar diharapkan dapat merata untuk setiap kelas. Diharapkan tidak ada dosen atau kelas yang memiliki frekuensi kuliah terlalu tinggi pada satu harinya
4. Pekerjaan penjadwalan mata kuliah akan semakin rumit jika melibatkan semakin banyak mata kuliah, kelas maupun dosen.

Untuk mendapatkan jadwal yang optimal, maka harus dievaluasi banyak kemungkinan penjadwalan. Karena itu dibutuhkan metode optimasi yang dapat digunakan untuk mengerjakan penjadwalan mata kuliah dengan mempertimbangkan batasan-batasan di atas. Terdapat beberapa alternatif metode optimasi yang dapat digunakan untuk mengoptimalkan penjadwalan, seperti Algoritma Genetik (GA) [2], logika Fuzzy [3, 4], *Particle Swarm Optimization* (PSO) [5], maupun *Ant Colony Optimization* (ACO) [6]. Pendekatan menggunakan ACO menjadi pilihan dalam penelitian ini. ACO diinspirasi dari perilaku kelompok semut dalam mencari dan mengumpulkan makanan. Pada saat awal pencarian, kelompok semut akan menyebar secara acak hingga menemukan makanan. Jika makanan sudah ditemukan maka semut akan bolak-balik antara makanan dan sarang. Selama perjalanan bolak-balik tersebut, semut akan menebarkan pheromone. Jika terdapat semut yang melalui jalur yang pendek, maka semut itu akan bolak-balik lebih cepat sehingga pheromone yang disebar pada jalur tersebut akan lebih banyak. Sedangkan jika ada semut yang melalui jalur yang jauh, maka semut akan membutuhkan waktu bolak-balik lebih lama dan pheromone yang disebar pada jalur tersebut akan lebih sedikit. Karena semut akan tertarik pada jalur yang lebih banyak pheromone, maka jalur yang terpendek

akan dipilih oleh kelompok semut. Prinsip dari algoritma ini ditunjukkan pada **Gambar 1**. Algoritma ACO masih terbatas untuk menyelesaikan permasalahan optimasi dalam kasus diskrit [7]. Dalam permasalahan optimasi diskrit, algoritma ACO telah berhasil diimplementasikan dalam berbagai kasus, seperti masalah *travelling-salesman problem* (TSP) [8], penjadwalan pekerjaan [9] ataupun perutean kendaraan [10]. Penelitian ini mengimplementasikan algoritma ACO untuk mengoptimalkan jadwal perkuliahan dengan melibatkan berbagai batasan yang sering terjadi pada kasus penjadwalan perkuliahan.



Gambar 1. Prinsip dari algoritma ACO : (a) Kelompok semut secara acak mencari jalur menuju makanan, (b) jalur terpendek akan terdapat lebih banyak pheromone yang tersebar, maka kelompok semut akan cenderung melalui jalur tersebut

Terdapat beberapa publikasi mengenai penelitian optimasi penjadwalan menggunakan ACO. Deng [11] menggunakan algoritma ACO untuk penjadwalan awak pesawat. Chen [12] menggunakan ACO untuk penjadwalan staf pada suatu proyek berdasarkan suatu peristiwa. Zhou [13] menggunakan algoritma ACO untuk diimplementasikan pada kasus *dynamic job-shop problem*. Zhang [14] menggunakan algoritma ACO untuk pengendalian multi satelit. Perbedaan yang dilakukan pada penelitian ini adalah melibatkan batasan-batasan (*constraint*) yang harus dipertimbangkan dalam penyusunan jadwal kuliah. Batasan yang dilibatkan diantaranya adalah dosen dan suatu kelas tidak boleh dijadwalkan kuliah berturut-turut lebih dari dua kali mata kuliah, tidak boleh angkatan yang berdekatan dijadwalkan secara bersamaan agar mahasiswa tetap dapat mengulang suatu mata kuliah, tidak boleh ada dosen dan kelas yang memiliki frekuensi kuliah terlalu tinggi dalam satu hari, serta mengakomodasi terbatasnya waktu dosen yang dapat digunakan untuk perkuliahan.

Penelitian ini bertujuan untuk mengoptimalkan penjadwalan mata kuliah menggunakan algoritma ACO. Metode yang dikembangkan akan menemukan solusi optimal dari penjadwalan mata kuliah di mana ada beberapa batasan (*constraint*) yang perlu dipertimbangkan. Batasan-batasan itu adalah dosen atau suatu kelas hanya diijinkan untuk dijadwalkan kuliah berturut-turut maksimal dua mata kuliah. Batasan kedua adalah tidak boleh dua angkatan yang berbeda satu tahun dijadwalkan dalam waktu yang bersamaan, agar memungkinkan bagi mahasiswa untuk mengambil mata kuliah ke bawah atau ke atas. Batasan ketiga adalah tidak boleh ada jadwal kuliah dengan frekuensi yang terlalu tinggi dalam satu hari bagi dosen maupun kelas yang sama. Dan batasan keempat adalah harus dapat mengoptimalkan jadwal perkuliahan meskipun jadwal alternatif yang disediakan oleh dosen terbatas. Agar kasus penjadwalan mata kuliah ini dapat diselesaikan menggunakan algoritma ACO, maka dibangun grapp dengan setiap nodenya adalah nama mata kuliah yang harus dijadwalkan. Jalur yang dibuat oleh semut dari node awal ke node akhir akan berisi urutan mata kuliah yang harus dijadwalkan dalam satu minggunya.

II. METODE

Agar algoritma ACO dapat menyelesaikan kasus penjadwalan mata kuliah ini, maka digunakan grafik graph seperti yang ditunjukkan pada **Gambar 2**. Pada graph tersebut berisi node-node yang mewakili setiap mata kuliah yang harus dijadwalkan. Jalur yang dibuat oleh semut dari node awal ke node akhir akan berisi urutan mata kuliah yang harus dijadwalkan dalam satu

minggunya. Jika suatu node mata kuliah sudah dilalui oleh semut, maka node-node lain dengan nama mata kuliah yang sama akan dihilangkan sementara dari graph. Hal ini akan menjaga agar suatu mata kuliah yang akan dijadwalkan sekali saja. Graph ini digunakan untuk mentransformasikan masalah penjadwalan mata kuliah kedalam bentuk permasalahan optimasi kombinatorial. Algoritma ACO akan diterapkan melalui graph ini.

Pada setiap jalur yang dilaluinya, semut-semut akan meletakkan *pheromone*. Terdapat dua jenis pembaharuan (*update*) nilai *pheromone*, yaitu *update pheromone local*, dan *update pheromone global*. Setelah semut membangun jalur lengkap yang menghubungkan node awal ke node akhir, maka dilakukan *update pheromone lokal* yang mengikuti persamaan (1).

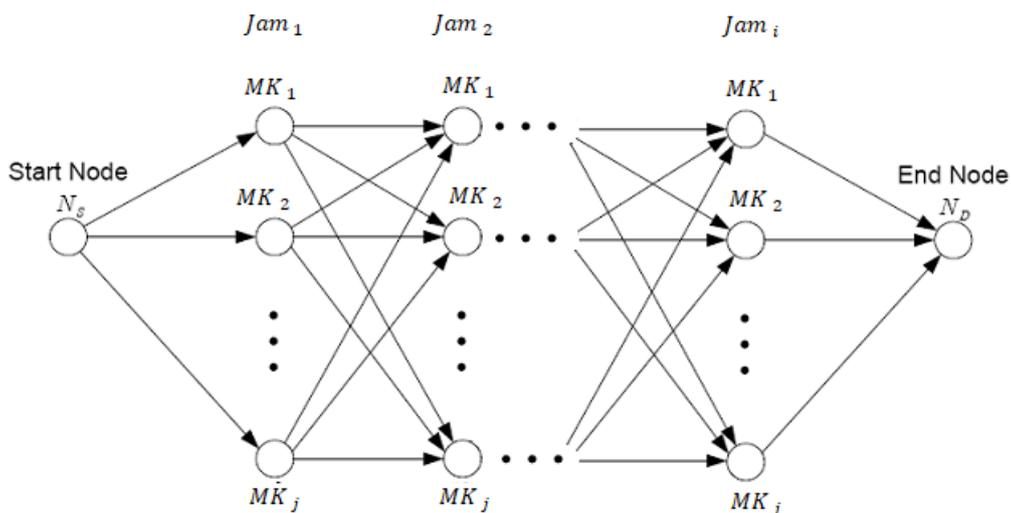
$$\tau_{ij}(t) = \tau_{ij}(t - 1) + C_{nn} \tag{1}$$

Dimana $C_{nn} = \frac{1}{n \times L_{nn}}$.

Variabel C_{nn} disebut variabel *peng-update-an pheromone*. n adalah jumlah jalur yang digunakan, sedangkan L_{nn} adalah nilai *fitness* dari kualitas jadwal yang diberikan oleh jalur tersebut.

Untuk *update pheromone global*, ada *update pheromone positif* (persamaan 2) dan ada *update pheromone negative* (persamaan 3). Persamaan yang digunakan adalah sebagai berikut.

$$\tau_{ij}^{best}(t) = \tau_{ij}^{best}(t - 1) + \frac{\rho}{L_{best}} \tag{2}$$



Gambar 2. Representasi graph permasalahan penjadwalan mata kuliah

$$\tau_{ij}^{worst}(t) = \tau_{ij}^{worst}(t - 1) - \frac{0.3\rho}{L_{worst}} \quad (3)$$

Dimana τ_{ij}^{best} adalah nilai *pheromone* dari tur yang memberikan nilai fitness terbaik (C_{best}) dalam satu iterasi. Adapun τ_{ij}^{worst} adalah *pheromone* dari tur yang memberikan nilai fitness terburuk (C_{worst}) dalam satu iterasi. *Pheromone* dari jalur yang memberikan nilai fitness terbaik akan terus bertambah. Adapun *pheromone* dari jalur yang memberikan nilai fitness terburuk akan terus berkurang selama iterasi.

Setelah satu iterasi selesai, beberapa *pheromone* akan menguap dari jalur. Penguapan *pheromone* menggunakan persamaan (4). Proses penguapan ini akan membuat algoritma ACO lebih terbuka untuk melakukan pelacakan pada arah baru.

$$\tau_{ij}(t) = \tau_{ij}(t)^\lambda + \Delta \quad (4)$$

Dengan λ adalah konstanta penguapan serta Δ adalah hasil penjumlahan persamaan (2) dan (3).

Evaluasi fitness dari suatu hasil penjadwalan dilakukan dengan menghitung faktor-faktor berikut.

1. Apakah ada mata kuliah yang dipecah (F_1) ?
 Jika ada mata kuliah dengan bobot 3 SKS, sistem dapat menjadwalkannya dengan cara dibagi dalam 2 kelompok jam kuliah. Pemecahan ini dilakukan jika tidak ditemukan alokasi waktu yang tersedia untuk menjadwalkan mata kuliah tersebut dalam satu waktu. Proses ini dilakukan untuk memperbesar kemungkinan tersusunnya penjadwalan berdasarkan masukan yang ada. Tetapi proses pemecahan mata kuliah ini akan memperkecil nilai fitness. Sehingga program akan cenderung membuat alternatif penjadwalan yang tidak terlalu banyak memiliki pemecahan mata kuliah.
2. Apakah ada pepadatan jadwal disuatu waktu (F_2) ?
 Agar produktivitas pemakaian ruangan meningkat, maka dikehendaki agar ruangan dapat beroperasi semenjak pagi hingga petang. Jika program menghasilkan jadwal dimana terdapat waktu kosong dimana ruangan tidak tergunakan, maka akan mengurangi nilai fitness.
3. Apakah ada frekuensi mengajar dosen yang terlalu tinggi (F_3) ?
 Diinginkan agar tugas mengajar dosen dapat terdistribusi merata tiap harinya. Hal ini agar

performansi mengajar dosen tetap optimal. Jika dosen mengajar terlalu banyak dalam suatu harinya, maka akan lelah dan tidak optimal. Maka nilai fitness akan berkurang jika dalam suatu jadwal yang dibuat terdapat frekuensi mengajar dosen yang terlalu tinggi.

4. Apakah ada frekuensi kuliah mahasiswa yang terlalu tinggi (F_4) ?
 Seperti halnya dosen, mahasiswa juga akan lelah dan susah konsentrasi jika mengikuti perkuliahan yang terlalu banyak pada suatu harinya. Maka nilai fitness akan berkurang jika dalam suatu jadwal yang dibuat terdapat frekuensi kuliah mahasiswa yang terlalu tinggi.
5. Apakah ada mata kuliah yang diikuti kelas yang sama yang berdekatan (F_5) ?
 Diinginkan agar mahasiswa memiliki waktu istirahat setelah selesai perkuliahan dan sebelum memulai perkuliahan selanjutnya. Maka jika terdapat dua mata kuliah yang diikuti kelas yang sama dan dijadwalkan berdekatan, maka akan mengurangi nilai fitness.
6. Apakah dalam satu hari ada dua mata kuliah yang diikuti kelas yang sama yang terpisah terlalu jauh waktunya (F_6) ?
 Walaupun dikehendaki ada jeda antar dua mata kuliah, tetapi diinginkan mahasiswa tidak perlu menunggu terlalu lama diantara dua mata kuliah tersebut.
 Maka rumus fitness yang digunakan ditunjukkan pada persamaan (5).

$$Fitness = \frac{1}{B_1 \times F_1} + \frac{1}{B_2 \times F_2} + \frac{1}{B_3 \times F_3} + \frac{1}{B_4 \times F_4} + \frac{1}{B_5 \times F_5} + \frac{1}{B_6 \times F_6} \quad (5)$$

dengan :

- F_1 = banyaknya pemecahan mata kuliah
- F_2 = banyaknya waktu yang kosong
- F_3 = banyaknya frekuensi mengajar dosen tinggi
- F_4 = banyaknya frekuensi kuliah mahasiswa tinggi
- F_5 = banyaknya mata kuliah berdekatan
- F_6 = banyaknya mata kuliah berjauhan
- B_1 = bobot pemecahan mata kuliah
- B_2 = bobot adanya waktu yang kosong
- B_3 = bobot frekuensi mengajar dosen tinggi
- B_4 = bobot frekuensi kuliah mahasiswa tinggi
- B_5 = bobot mata kuliah berdekatan
- B_6 = bobot mata kuliah berjauhan

III. HASIL DAN DISKUSI

Terdapat empat masukan data yang harus diberikan kepada sistem, yaitu tabel mata kuliah, tabel waktu kesediaan dosen untuk mengajar (tabel dosen), tabel kelas, dan tabel ruang. **Tabel I** adalah contoh dari tabel dosen. **Tabel II** adalah contoh dari tabel kelas. **Tabel III** adalah contoh dari tabel ruang. Dan **Tabel IV** adalah contoh dari tabel mata kuliah.

Tabel mata kuliah berisikan daftar seluruh mata kuliah yang hendak dijadwalkan. Data yang perlu dimasukkan pada tabel mata kuliah adalah nama kelas, jumlah sks, nama mata kuliah dan dosen pengajar mata kuliah tersebut. Tabel dosen, tabel kelas dan tabel ruang berisi informasi mengenai waktu yang dapat digunakan untuk mengalokasikan suatu mata kuliah. Angka 1 pada **Tabel I – Tabel III** menunjukkan bahwa pada waktu tersebut tidak dapat dialokasikan jadwal perkuliahan. Pada **Tabel I** merupakan contoh dimana dosen pada hari Senin hingga Rabu dapat dijadwalkan diantara jam 07.00 – 14.00. Tetapi pada hari Kamis hanya bisa dijadwalkan untuk mengajar diantara jam 07.00 – 12.00. **Tabel I - Tabel III** akan diupdate setiap suatu mata kuliah selesai dijadwalkan.

Tabel I. Contoh Tabel Dosen

Dosen A	Senin	Selasa	Rabu	Kamis	Jum'at	Sabtu
07.00 – 08.00					1	1
08.00 – 09.00					1	1
09.00 – 10.00					1	
10.00 – 11.00					1	
11.00 – 12.00						
12.00 – 13.00				1		
13.00 – 14.00				1		
14.00 – 15.00	1	1	1	1	1	1
15.00 – 16.00	1	1	1	1	1	1
16.00 – 17.00	1	1	1	1	1	1

Tabel II. Contoh Tabel Kelas

19TE-1	Senin	Selasa	Rabu	Kamis	Jum'at	Sabtu
07.00 – 08.00						1
08.00 – 09.00						1
09.00 – 10.00						1
10.00 – 11.00						1
11.00 – 12.00					1	1
12.00 – 13.00					1	1
13.00 – 14.00						1
14.00 – 15.00						1
15.00 – 16.00						1
16.00 – 17.00						1

Tabel III. Contoh Tabel Ruang

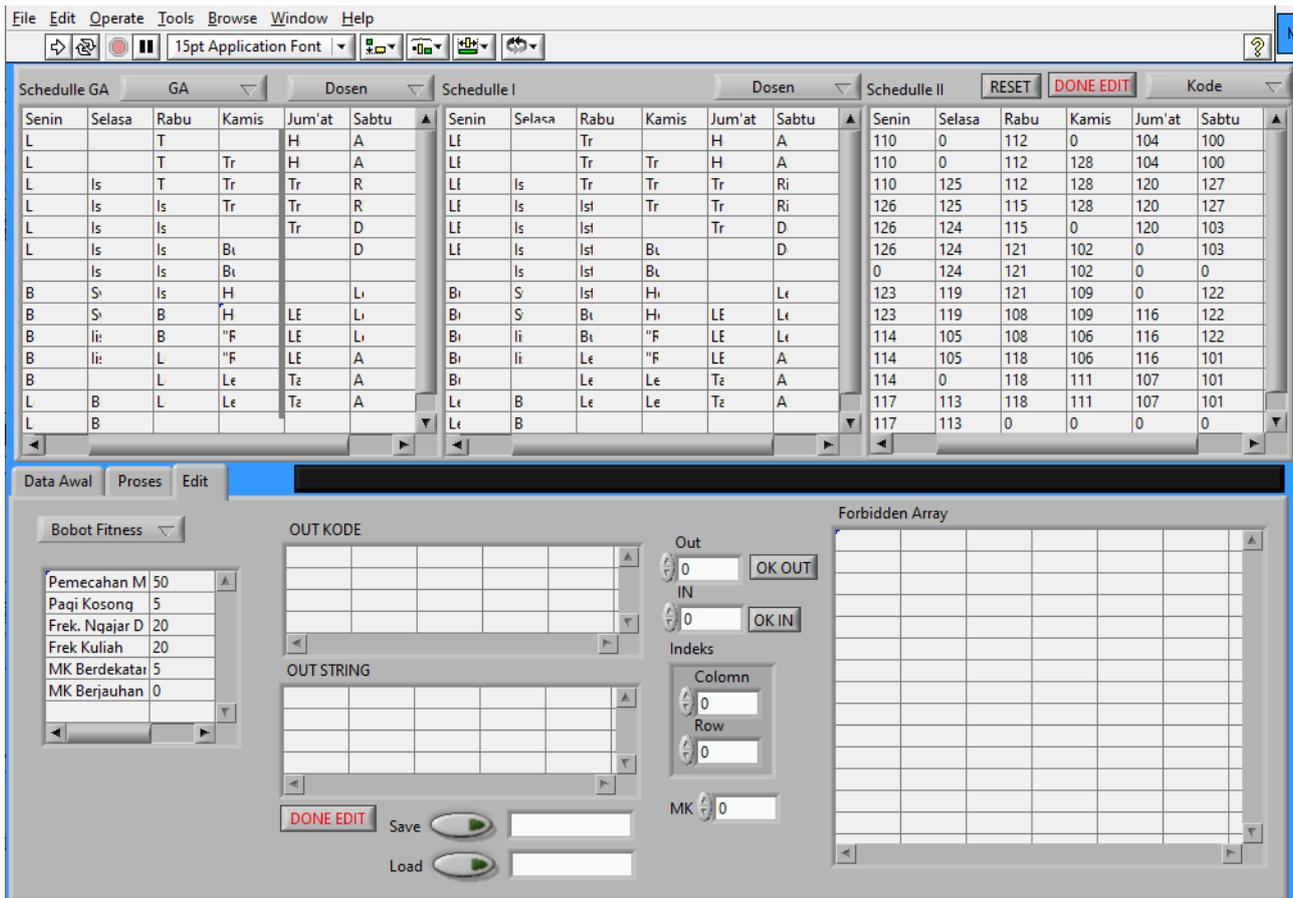
R. 4701	Senin	Selasa	Rabu	Kamis	Jum'at	Sabtu
07.00 – 08.00						
08.00 – 09.00						
09.00 – 10.00						
10.00 – 11.00						
11.00 – 12.00					1	
12.00 – 13.00					1	
13.00 – 14.00						
14.00 – 15.00						
15.00 – 16.00						
16.00 – 17.00						

Tabel IV Contoh Tabel Mata Kuliah

Kelas	Mata Kuliah	SKS	Dosen
19TE-1	Bahasa Inggris II	2	Dosen I
19TE-1	Fisika Dasar II	2	Dosen JU
19TE-1	Pancasila	2	Dosen OS
19TE-1	Pengukuran Listrik	2	Dosen BH
19TE-1	Agama	2	Dosen JA
18TE-1	Rangkaian Listrik II	2	Dosen TR
18TE-1	Matematika Teknik II	2	Dosen TR
18TE-1	Elektronika	2	Dosen BH
18TE-1	Elektromagnetika	2	Dosen MA
18TE-1	Telekomunikasi I	3	Dosen LO
18TE-1	Dasar Sistem Kontrol	3	Dosen IR
17TE-1	Mikrokontroler	3	Dosen S
17TE-1	Bahasa Indonesia	2	Dosen J
17TE-1	Metodologi Penelitian	2	Dosen YE
17TE-1	Teknik Tenaga Listrik	2	Dosen BH

Selanjutnya tabel mata kuliah, tabel dosen, tabel ruang dan tabel kelas akan digabungkan menjadi tabel prioritas mata kuliah. Tabel prioritas mata kuliah menginformasikan banyaknya alternatif waktu yang dapat dipilih untuk mengalokasikan suatu mata kuliah. Tabel prioritas mata kuliah mengurutkan mata kuliah dari yang paling sedikit alternatif waktu yang dapat dipilihnya, hingga mata kuliah yang paling banyak memiliki alternatif waktu pilihannya. Proses ini dilakukan agar tidak ada mata kuliah yang tidak dapat dialokasikan karena jadwal alternatif yang memungkinkan pelaksanaan mata kuliah tersebut sudah dijadwalkan untuk pelaksanaan mata kuliah lainnya.

Implementasi dari algoritma ini dilakukan pada program LabVIEW. Block diagram ditunjukkan pada **Gambar 3**. Tampilan program yang dibuat ditunjukkan pada **Gambar 4**.



Gambar 4. Tampilan front panel LabVIEW

Terdapat tiga kondisi berhenti dimana semut tidak perlu melanjutkan iterasinya lagi. Pertama adalah jika setelah beberapa iterasi berturut-turut tidak lagi terjadi perubahan nilai fitness. Kondisi berhenti yang kedua adalah jika jumlah iterasi maksimum sudah tercapai. Adapun kondisi berhenti yang ketiga adalah jika nilai fitness minimal yang dikehendaki sudah tercapai. Jika salah satu kondisi diatas sudah tercapai, maka algoritma ACO akan dihentikan, dan selama ketiga kondisi tersebut belum ada yang tercapai maka semut masih akan mengulangi proses pencarian.

Contoh hasil penjadwalan yang diberikan oleh program ditunjukkan pada Gambar 5. Terlihat bahwa hampir seluruh jadwal pemakaian ruang telah berhasil diisi oleh program. Hal ini menunjukkan kemampuan program untuk memanfaatkan setiap waktu yang tersedia. Bobot fitness yang digunakan adalah sebagai berikut.

- $B_1 = 20$
- $B_2 = 5$
- $B_3 = 20$
- $B_4 = 20$
- $B_5 = 5$
- $B_6 = 0$

Berdasarkan nilai bobot yang digunakan dapat terlihat bahwa prioritas utama dalam penjadwalan ini adalah agar tidak ada mata kuliah yang dipecah penjadwalannya serta agar tidak ada jadwal kuliah dosen dan mahasiswa yang terlalu sering pada suatu hari. Prioritas ini dapat terlihat dari nilai bobot yang dibuat lebih tinggi daripada bobot lainnya. Adapun jadwal yang dibuat tidak memperlmasalahkan jika ada dua mata kuliah dari satu kelas yang letaknya berjauhan ($B_6 = 0$).

Algoritma yang dirancang telah diujikan pada penjadwalan mata kuliah yang melibatkan 38 mata kuliah, 4 angkatan kelas, 6 ruang perkuliahan serta 12 dosen pengampu mata kuliah. Algoritma ACO berhasil menyusun jadwal final yang memiliki nilai fitness 0,0092. Pada jadwal tersebut tidak terdapat dosen yang memiliki frekuensi mengajar tinggi pada suatu harinya. Tetapi terdapat 12 jadwal mata kuliah yang membuat suatu kelas mengikuti perkuliahan dalam frekuensi yang tinggi. Serta terdapat 4 mata kuliah yang dijadwalkan berdekatan. Performansi akhir ini dinilai cukup baik dan menunjukkan bahwa proses optimalisasi penjadwalan mata kuliah yang memiliki beberapa *constraint* telah berhasil dilakukan menggunakan algoritma ACO.

	Senin	Selasa	Rabu	Kamis	Jum'at	Sabtu
07.00 - 07.45	Dasar Konversi Energi	Bahasa Inggris II	Kewirausahaan &	Komputer Aplikasi TE II	Elektronika II	Komputer Aplikasi IT II
	05TE-1	07TE-1	04TE-1	07TE-1	06TE-1	07TE-1
07.45 - 08.30	Dasar Konversi Energi	Bahasa Inggris II	Kewirausahaan &	Komputer Aplikasi TE II	Elektronika II	Komputer Aplikasi IT II
	05TE-1	07TE-1	04TE-1	07TE-1	06TE-1	07TE-1
08.30 - 09.15	Sistem Komunikasi	Antena dan Propagasi	Kewirausahaan &	Broad Band	Dasar Telekomunikasi	Rangkaian Listrik II
	04TE-1	04TE-1	04TE-1	04TE-1	06TE-1	06TE-1
09.15 - 10.00	Sistem Komunikasi	Antena dan Propagasi	Medan Elektromagnetik	Broad Band	Dasar Telekomunikasi	Rangkaian Listrik II
	04TE-1	04TE-1	06TE-1	04TE-1	06TE-1	06TE-1
10.00 - 10.45	Sistem Komunikasi	Antena dan Propagasi	Medan Elektromagnetik	Broad Band	Komunikasi Data	Rangkaian Listrik II
	04TE-1	04TE-1	06TE-1	04TE-1	04TE-1	06TE-1
10.45 - 11.30	Kendali Optimal	Dasar Sistem Kontrol	Medan Elektromagnetik	Elektronika	Komunikasi Data	Komputer Aplikasi TE VI
	05TE-1	06TE-1	06TE-1	05TE-1	04TE-1	05TE-1
11.30 - 12.15	Kendali Optimal	Dasar Sistem Kontrol	Sistem Komunikasi	Elektronika		Komputer Aplikasi TE VI
	05TE-1	06TE-1	05TE-1	05TE-1		05TE-1
12.15 - 13.00	Kendali Optimal	Dasar Sistem Kontrol	Sistem Komunikasi	Elektronika		Metodologi Penelitian
	05TE-1	06TE-1	05TE-1	05TE-1		05TE-1
13.00 - 13.45	Kendali Multivariabel	Instrumentasi Sistem	Sistem Komunikasi	Instrumentasi Berbasis	Kewarganegaraan	Metodologi Penelitian
	05TE-1	04TE-1	05TE-1	04TE-1	04TE-1	05TE-1
13.45 - 14.30	Kendali Multivariabel	Instrumentasi Sistem	Sistem Kendali	Instrumentasi Berbasis	Kewarganegaraan	Matematika Teknik II
	05TE-1	04TE-1	04TE-1	04TE-1	04TE-1	06TE-1
14.30 - 15.15	Kendali Multivariabel	Komputer Aplikasi TE IV	Sistem Kendali	Pengukuran Listrik		Matematika Teknik II
	05TE-1	06TE-1	04TE-1	07TE-1		06TE-1
15.15 - 16.00	Kalkulus II	Komputer Aplikasi TE IV	Sistem Kendali	Pengukuran Listrik	Sistem Embedded	Matematika Teknik II
	07TE-1	06TE-1	04TE-1	07TE-1	04TE-1	06TE-1
16.00 - 16.45	Kalkulus II	Agama	Mikroprosesor	Pancasila	Sistem Embedded	Fisika Dasar II
	07TE-1	07TE-1	05TE-1	07TE-1	04TE-1	07TE-1
16.45 - 17.30	Kalkulus II	Agama	Mikroprosesor	Pancasila	Sistem Embedded	Fisika Dasar II
	07TE-1	07TE-1	05TE-1	07TE-1	04TE-1	07TE-1

Gambar 4. Contoh hasil penjadwalan

Hasil penjadwalan dapat memiliki nilai fitness yang baik karena algoritma ACO menerapkan prinsip “pembelajaran dan perbaikan dari hasil sebelumnya”. Hasil fitness yang diperoleh dari penjadwalan menggunakan algoritma ACO ini, yaitu sebesar 0,0092, lebih baik dari hasil penjadwalan menggunakan algoritma GA yang dilakukan Aria pada [1], yaitu 0,0083. Walaupun pada hasil penjadwalan keduanya masih terdapat mata kuliah yang berdekatan.

IV. KESIMPULAN

Penelitian ini telah berhasil mengimplementasikan algoritma ACO untuk mengoptimalkan penjadwalan mata kuliah pada kasus multi angkatan dan multi ruangan. Program yang dibuat telah dapat mencari solusi penjadwalan pada waktu yang dapat digunakan oleh dosen, mahasiswa maupun ruangan. Program juga dapat meminimalkan mata kuliah yang harus dilaksanakan dengan dipecah serta meminimalkan jadwal kuliah dengan frekuensi yang tinggi baik bagi dosen maupun mahasiswa. Hasil

penjadwalan dapat memiliki nilai fitness yang baik karena algoritma ACO menggunakan prinsip “pembelajaran dan perbaikan dari hasil sebelumnya”. Dari salah satu hasil pengujian penjadwalan 38 mata kuliah, 4 angkatan kelas, 6 lokasi perkuliahan dan 12 dosen pengampu mata kuliah, algoritma yang diusulkan dapat memperoleh fitness 0,0092. Tidak terdapat dosen yang memiliki frekuensi mengajar tinggi pada suatu hari, tetapi terdapat 4 mata kuliah yang dijadwalkan berdekatan dan 12 jadwal kuliah yang menyebabkan suatu kelas mengikuti frekuensi perkuliahan yang tinggi.

REFERENCES

- [1] M . Aria, “Aplikasi algoritma genetik untuk optimasi penjadwalan mata kuliah,” *Majalah Ilmiah Unikom*, vol.6, hlm. 17-25, 2008
- [2] A. Jain, S Jain, dan P. K. Chande, “Formulation of genetic algorithm to generate good quality course timetable,” *International Journal of Innovation, Management and Technology*, vol. 1, no. 3, pp. 248 - 251,2010
- [3] A. Chaudhuri, dan K. De, “Fuzzy genetic heuristic for university course timetable problem,” *International Journal Advance. Soft Comput. Appl*, vol. 2, no. 1, pp. 100-123, 2010
- [4] Z. Jia, X. Lu, J. Yang, dan D. Jia, “Research on job-shop scheduling problem based on genetic algorithm,” *International Journal of Production Research*, vol. 49, no. 12, pp. 3585-3604, 2011
- [5] D. F. Shiau, “A hybrid particle swarm optimization for a university course scheduling problem with flexible preferences,” *Expert Systems with Applications*, vol. 38, no. 1, pp. 235-248, 2011
- [6] L. N. Xing, Y. W. Chen, P. Wang, Q. S. Zhao, dan J. Xiong, “A knowledge-based ant colony optimization for flexible job shop scheduling problems,” *Applied Soft Computing*, vol. 10, no. 3, pp. 888-896, 2010
- [7] B. Yu, Z. Z. Yang, dan B. Yao, “An improved ant colony optimization for vehicle routing problem,” *European journal of operational research*, vol. 196, no. 1, pp. 171-176, 2009
- [8] W. Junqiang, dan O. Aijia, “A hybrid algorithm of ACO and delete-cross method for TSP,” *2012 International Conference on Industrial Control and Electronics Engineering*, pp. 1694-1696, 2012
- [9] L. N. Xing, Y. W. Chen, P. Wang, Q. S. Zhao, dan J. Xiong, “A knowledge-based ant colony optimization for flexible job shop scheduling problems,” *Applied Soft Computing*, vol. 10, no. 3, pp. 888-896, 2010
- [10] J. Brito, F. J. Martínez, J. A. Moreno, dan J. L. Verdegay, “An ACO hybrid metaheuristic for close–open vehicle routing problems with time windows and fuzzy constraints,” *Applied Soft Computing*, vol. 32, pp. 154-163, 2015
- [11] G. F. Deng, dan W. T. Lin, “Ant colony optimization-based algorithm for airline crew scheduling problem,” *Expert Systems with Applications*, vol. 38, no. 5, pp. 5787-5793, 2011
- [12] W. N. Chen, dan J. Zhang, “Ant colony optimization for software project scheduling and staffing with an event-based scheduler,” *IEEE Transactions on Software Engineering*, vol. 39, no. 1, pp. 1-17, 2012
- [13] R. Zhou, A. Y. C. Nee, dan H. P. Lee, “Performance of an ant colony optimisation algorithm in dynamic job shop scheduling problems,” *International Journal of Production Research*, vol. 47, no. 11, pp. 2903-2920, 2009
- [14] Z. Zhang, N. Zhang, dan Z. Feng, “Multi-satellite control resource scheduling based on ant colony optimization,” *Expert Systems with Applications*, vol. 41, no. 6, pp. 2816-2823, 2014