

Course Scheduling System Using Genetic Algorithm

Harry Rahmat Gazali, Satria Perdana Arifin , Rika Perdana Sari

Politeknik Caltex Riau

Jl.Umbansari No.1 Rumbai - Riau, telp: (0761) 59393 / fax: (0761) 54224

e-mail: harryrahmatgazali@yahoo.com

Abstrak

Penjadwalan kuliah dalam suatu universitas bukanlah hal yang mudah untuk diselesaikan. Selain dilihat dari sisi banyaknya mata kuliah juga harus dilihat dari sisi dosen, yaitu kemungkinan-kemungkinan dosen akan mengajar lebih dari satu mata kuliah yang ada. Sebab ada kemungkinan jumlah mata kuliah dan jumlah dosen tidak sebanding. Selain itu, harus dipertimbangkan juga ketersediaan ruang kelas sehingga kegiatan belajar-mengajar dapat dilaksanakan. Penjadwalan kuliah pada Fakultas Keguruan dan Ilmu Pendidikan Universitas Lancang Kuning masih dilakukan secara manual, sehingga membutuhkan waktu yang relatif lama dalam menyusun jadwal perkuliahan dan sering terjadi human error berupa bentrokan jadwal dosen untuk mengajar, pemakaian ruang kuliah dan jadwal perkuliahan pada setiap kelas. Oleh karena itu, diperlukan sebuah solusi untuk masalah penjadwalan kuliah ini. Metode yang digunakan adalah pendekatan Algoritma Genetika yang merupakan pendekatan komputasional untuk menyelesaikan masalah yang dimodelkan dengan proses evolusi biologis. Namun, penggunaan Algoritma Genetika ini masih menimbulkan bentrokan pada sisi dosen (1,70 %), pemakaian ruang kuliah (19,03 %) dan jadwal perkuliahan pada setiap kelas (5,96%), sehingga hasil tersebut belum dapat dipergunakan untuk penjadwalan sesungguhnya dan masih memerlukan penelitian dan pengembangan lebih lanjut dengan mencoba metode seleksi, crossover dan mutasi yang berbeda.

Kata kunci: Penjadwalan kuliah, Algoritma Genetika

Abstract

Scheduling of courses in an university is not easy to solve. Beside viewing from the amount courses, we also need to view from lecturers perspective, namely possibilities of the lecturer that will teach more than one course. Because there is a possibility that the number of courses and the number of lecturers are not comparable. Moreover, classroom availability to manage teaching-learning activity should also be considered. Scheduling of courses at Fakultas Keguruan dan Ilmu Pendidikan Universitas Lancang Kuning still done manually, so that it requires a relatively long time in scheduling the courses and human errors often occurs in form of conflict on lecturer's teaching schedule, classrooms usage, and the courses schedule of each classes. Therefore, a solution is needed to solve the existing scheduling problems. The method used is a Genetic Algorithm which is a computational approach to solve modeled problems by the biological evolution process. However, the use of Genetic Algorithm is still causing a conflict on the lecturers (1,70 %), usage of the classrooms (19,03 %) and the schedule of each classes (5,96 %), so these results can not be used for a real scheduling and still need researches and further development by trying different methods such as selection, crossover and mutation.

Key words: Scheduling of courses, Genetic Algorithm

1. Introduction

Penjadwalan merupakan sebuah teknik dalam mengatur waktu. Salah satu contoh implementasi penjadwalan dalam bidang akademik adalah penjadwalan perkuliahan. Penjadwalan kuliah pada Fakultas Keguruan dan Ilmu Pendidikan Universitas Lancang Kuning masih dilakukan secara manual. Hal ini bukanlah sesuatu yang mudah dan sering terjadi human error berupa bentrokan pemakaian ruang kelas dan jadwal dosen untuk mengajar. Selain dilihat dari sisi banyaknya mata kuliah juga harus dilihat dari sisi dosen, yaitu kemungkinan-kemungkinan dosen akan mengajar lebih dari satu mata kuliah yang ada. Sebab ada kemungkinan jumlah mata kuliah dan jumlah dosen tidak sebanding.

Selain itu, harus dipertimbangkan juga ketersediaan ruang kelas sehingga kegiatan belajar-mengajar dapat dilaksanakan.

Penjadwalan kuliah secara manual membutuhkan waktu yang cukup lama dan konsentrasi agar kombinasi terbaik untuk pemakaian ruang kelas dan jadwal dosen mengajar diperoleh. Untuk mempermudah masalah penjadwalan kuliah ini, maka dibangunlah sebuah aplikasi yang diharapkan dapat menyelesaikan masalah penjadwalan kuliah pada Fakultas Keguruan dan Ilmu Pendidikan Universitas Lancang Kuning. Metode yang digunakan untuk menyelesaikan permasalahan tersebut adalah dengan pendekatan genetic algorithm.

Genetic algorithm sudah sering digunakan dalam permasalahan penjadwalan kuliah. Hal ini disebabkan karena genetic algorithm dapat menghasilkan solusi melalui sedemikian proses seperti proses seleksi, proses crossover dan proses mutasi yang bervariasi sesuai dengan nilai probabilitas yang ditetapkan untuk setiap proses yang akan dilakukan. Keberhasilan genetic algorithm dalam menyelesaikan masalah tergantung pada penggunaan metode dalam proses crossover dan mutasi, serta nilai probabilitas yang diberikan.

1.1 Sejarah Genetic algorithm

Genetic Algorithm (GA) adalah suatu algoritma pencarian yang berbasis pada mekanisme seleksi alam dan genetika. Genetic algorithm diperkenalkan pertama kali oleh John Holland (1975) dari Universitas Michigan. John Holland menyatakan bahwa setiap masalah dapat diformulasikan ke dalam terminologi genetika. Dari sinilah GA diterapkan dan dikembangkan oleh murid-murid John Holland. Salah satu muridnya menggunakan GA pada aplikasi simulasi untuk mencari strategi yang paling mudah pada permainan catur. Murid yang lain membuat sebuah disertasi yang menggunakan GA untuk simulasi fungsi-fungsi dari organisme sel tunggal sebagai simulasi biologi pertama yang menggunakan GA. Ilmuwan-ilmuwan yang selanjutnya mengembangkan GA adalah D.E. Goldberg, K.De Jong, J.J. Grefenstette, L. Davis, Muhleinbein dan inspirator-inspirator lainnya yang ikut menjadikan AG berkembang (Kuswadi, 2007).

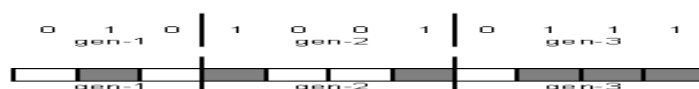
Genetic algorithm pada dasarnya adalah program komputer yang mensimulasikan proses evolusi. Dalam hal ini kromosom dihasilkan secara random dan memungkinkan untuk berkembang biak sesuai dengan hukum-hukum evolusi dengan harapan akan menghasilkan kromosom yang prima. Kromosom ini pada kenyataannya adalah kandidat penyelesaian dari masalah, sehingga bila kromosom yang baik berkembang, solusi yang baik terhadap masalah diharapkan akan dihasilkan (Sukmawan, 2003).

1.2 Komponen-Komponen Utama Model Genetika untuk Penjadwalan Kuliah

Komponen utama dalam model genetika penjadwalan kuliah, yaitu:

1. Teknik Penyandian/Representasi Kromosom

Teknik penyandian di sini meliputi penyandian gen dari kromosom. Gen merupakan bagian dari kromosom. Satu gen biasanya akan mewakili satu variabel. Gen dapat direpresentasikan dalam bentuk string bit, pohon, bilangan real, ataupun representasi lainnya (Kusumadewi, 2003).



Gambar 1.1 Representasi Biner

Dalam merepresentasikan kromosom untuk penjadwalan kuliah diperlukan jadwal kuliah kosong untuk setiap jam, setiap hari, dan setiap ruang. Jumlah slot pada jadwal kuliah tersebut merupakan hasil perkalian dari jumlah hari, jumlah ruangan, dan waktu kerja maksimum. Representasi kromosom dapat dilakukan dengan menggunakan gen yang bernilai integer untuk menyimpan indeks dari pertemuan kuliah (Suyanto, 2008). Misalkan, basis data pertemuan kuliah yang digunakan sebagai berikut:

Tabel 1.1 Basis Data Pertemuan Kuliah

Ukuran populasi tergantung pada masalah yang akan dipecahkan dan jenis operator genetika yang akan diimplementasikan. Setelah ukuran populasi ditentukan, kemudian harus dilakukan inialisasi terhadap kromosom yang terdapat pada populasi tersebut. Inialisasi kromosom dilakukan secara acak (Kusumadewi, 2003).

3. Fungsi Kebugaran (Fitness Function)

Seleksi alamiah di dunia nyata menyatakan bahwa hanya individu unggul (berkualitas saja) yang akan bertahan hidup. Sedangkan individu berkualitas rendah akan mati atau punah. Pada genetic algorithm, suatu individu dievaluasi berdasarkan suatu fungsi tertentu sebagai ukuran nilai kualitasnya. Fungsi ini dikenal dengan fungsi fitness. Terdapat banyak cara untuk membangun fungsi fitness (Suyanto, 2008).

Untuk membangun fungsi fitness yang baik pada penjadwalan kuliah harus memperhatikan pemakaian ruang kelas dan jadwal dosen. Jadwal kuliah yang memiliki bentrokan pemakaian ruang kelas dan jadwal dosen untuk mengajar bukanlah solusi yang baik. Maka, fungsi fitness untuk penjadwalan kuliah harus memperhatikan hal tersebut (Jankovic, 2008). Oleh karena itu, penghitungan nilai fitness dilakukan dengan:

- a. Penambahan nilai fitness jika perkuliahan ditempatkan pada ruang kuliah yang kosong.
- b. Penambahan nilai fitness jika dosen tidak mengajar lebih dari satu perkuliahan pada waktu yang sama.

Sehingga, rumus fungsi fitness-nya adalah:

$$f = \frac{Sk}{Ms} \dots\dots\dots(1)$$

Dimana,

Sk = Score penjadwalan kuliah

Ms = Score maksimum penjadwalan kuliah

Sedangkan,

$$Ms = S \times Ru \dots\dots\dots(2)$$

Dimana,

Sc = Jumlah kelas perkuliahan

Ru = Jumlah Syarat penjadwalan

Kromosom yang memiliki nilai fitness terbaik akan dilakukan dua proses genetika yaitu crossover dan mutasi dengan nilai yang telah ditentukan, kemudian dicari nilai fitness dari kromosom tersebut.

4. Seleksi

Seleksi akan menentukan individu-individu mana saja yang akan dipilih untuk dilakukan rekombinasi dan bagaimana offspring terbentuk dari individu-individu terpilih tersebut. Langkah pertama yang dilakukan dalam seleksi ini adalah pencarian nilai fitness. Masing-masing individu dalam suatu wadah seleksi akan menerima probabilitas reproduksi yang tergantung pada nilai objektif dirinya sendiri terhadap nilai objektif dari semua individu dalam wadah seleksi tersebut. Nilai fitness inilah yang nantinya akan digunakan untuk tahap-tahap seleksi berikutnya.

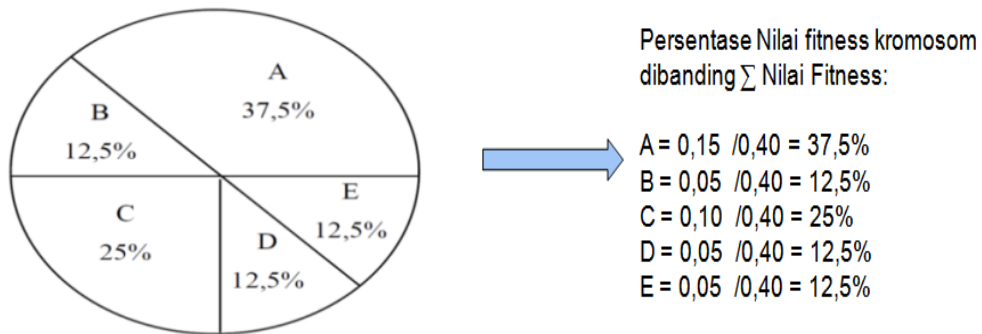
Ada beberapa metode seleksi, metode yang paling sederhana adalah Metode Roulette Wheel Selection dan sering juga dikenal dengan nama stochastic sampling with replacement. Pada metode seleksi ini, orangtua dipilih berdasarkan nilai fitnessnya, semakin baik nilai fitnessnya maka semakin besar kemungkinannya untuk terpilih (Suyanto, 2008). Diandaikan semua kromosom diletakkan pada sebuah roda roulette, besarnya kemungkinan bagi setiap kromosom adalah tergantung dari nilai fitnessnya seperti pada contoh berikut :

Kromosom	Nilai Fitness
A	0,15

Σ Nilai Fitnessnya :
 A + B + C + D + E
 = 0,15 + 0,05 + 0,10 + 0,05 + 0,05
 = 0,40

B	0,05
C	0,10
D	0,05
E	0,05

Gambar 1.4 Contoh Populasi dengan 5 Kromosom dengan Nilai Fitnessnya



Gambar 1.5 Probabilitas Terpilihnya Suatu Kromosom dalam Roda Roulette

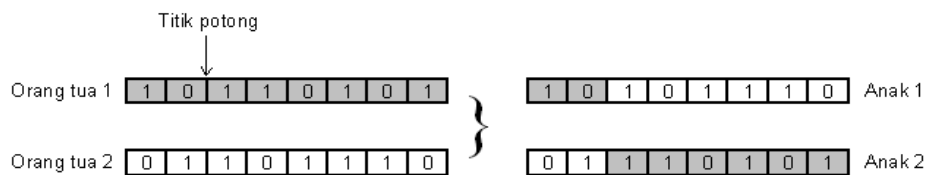
Probabilitas masing-masing individu merupakan hasil pembagian antara fitness masing-masing individu dengan total fitness dalam populasi. Dari contoh di atas, kromosom A memiliki probabilitas 37,5% untuk terpilih sebagai orangtua pembentuk keturunan baru pada setiap pemilihan kromosom (setiap roda diputar).

Tentu saja akan ada kromosom (individu) yang akan terpilih lebih dari sekali. Kromosom yang terbaik akan mampu berkembang biak lebih dari sekali, sedangkan yang terburuk akan mati (Kuswadi, 2007).

5. Operator Genetika

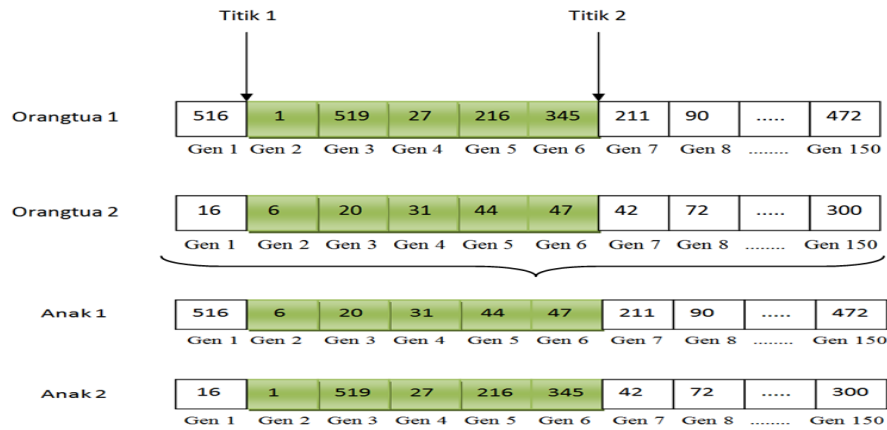
a) Rekombinasi/Crossover

Crossover (penyilangan) dilakukan atas dua kromosom untuk menghasilkan kromosom anak (offspring). Kromosom anak yang terbentuk akan mewarisi sebagian sifat kromosom induknya. Metode crossover yang paling sering digunakan pada genetic algorithm sederhana dengan kromosom berbentuk string biner adalah crossover satu titik (one-point crossover), yang dapat dilihat pada gambar berikut ini:



Gambar 1.6 Rekombinasi Satu Titik Potong pada Posisi 2 (Suyanto, 2008)

Operasi crossover pada penjadwalan kuliah dilakukan dengan pertukaran beberapa gen dari kedua induk, sehingga dihasilkan offspring yang mewarisi sebagian sifat dari kromosom induknya.



Gambar 1.7 Crossover Penjadwalan Kuliah (Suyanto, 2008)

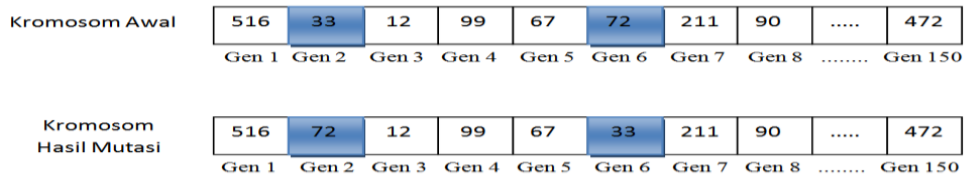
b) Mutasi

Mutasi yang digunakan pada genetic algorithm sederhana dengan kromosom biner, pada dasarnya akan mengubah secara acak nilai suatu bit pada posisi tertentu. Kemudian bit 1 diganti dengan 0, atau mengganti bit 0 dengan 1. Pada mutasi ini sangat dimungkinkan muncul kromosom baru yang semula belum muncul dalam populasi awal (Kusumadewi, 2003). Contoh mutasi dapat dilihat pada gambar berikut ini:



Gambar 1.8 Mutasi pada Gen ke-2 (Suyanto, 2008)

Operasi mutasi kromosom pada penjadwalan kuliah mengambil kuliah secara acak dan memindahkannya pada slot waktu secara acak, banyak kuliah yang akan dipindahkan pada sekali operasi ditentukan oleh ukuran mutasi.



Gambar 1.9 Mutasi Penjadwalan Kuliah (Suyanto, 2008)

6. Penentuan Parameter

Yang disebut dengan parameter di sini adalah parameter kontrol genetic algorithm, yaitu ukuran populasi (popsize), peluang crossover (p_c), dan peluang mutasi (p_m).

a) Ukuran Populasi (popsize)

Ukuran populasi mempengaruhi unjuk kerja yang baik dan keefektifan genetic algorithm. Genetic algorithm dengan populasi yang kecil biasanya unjuk kerjanya buruk karena populasi tidak menyediakan materi untuk mencakup ruang persoalan. Populasi yang lebih besar dibutuhkan untuk merepresentasikan keseluruhan ruang persoalan. Ukuran populasi sebaiknya tidak lebih kecil dari 30 untuk sembarang jenis permasalahan (Kuswadi, 2007).

b) Peluang Crossover (p_c)

Pada crossover ada satu parameter yang sangat penting yaitu peluang crossover (p_c). Peluang crossover menunjukkan rasio dari anak yang dihasilkan dalam setiap generasi dengan ukuran populasi. Misalkan ukuran populasi (popsize=100), sedangkan peluang crossover ($p_c=0,25$), berarti bahwa diharapkan ada 25 kromosom dari 100 kromosom yang ada pada populasi tersebut akan mengalami crossover. Nilai p_c yang sering dipakai biasanya berkisar 0,6 hingga 0,8 (Kuswadi, 2007).

c) Peluang Mutasi (p_m)

Pada mutasi ada satu parameter yang sangat penting yaitu peluang mutasi (p_m). Peluang mutasi menunjukkan presentase jumlah total gen pada populasi yang akan

mengalami mutasi. Untuk melakukan mutasi, terlebih dahulu harus menghitung jumlah total gen pada populasi tersebut. Kemudian bangkitkan bilangan random yang akan menentukan posisi mana yang akan dimutasi (gen keberapa pada kromosom keberapa). Misalkan ukuran populasi (popsize=100), setiap kromosom memiliki panjang 20 gen, maka total gen adalah $100 \times 20 = 2000$ gen. Jika peluang mutasi ($p_m = 0,01$), berarti bahwa diharapkan ada $(1/100) \times 2000 = 20$ gen akan mengalami mutasi. Nilai p_m yang sering dipakai berkisar $0,01 - 0,02$ (Kuswadi, 2007).

7. Kelebihan Algoritma Genetika

Algoritma genetika sangat cocok digunakan untuk menyelesaikan masalah optimasi dengan ruang lingkup yang besar, karena algoritma genetika selalu bergerak dengan mencari sejumlah solusi sekaligus, selama solusi tersebut masih bersifat feasible. Solusi dapat diperoleh setiap saat karena solusi dihasilkan pada generasi ke berapapun, namun dengan setting parameter yang tepat, diharapkan salah satu dari sekian banyak solusi yang dibangkitkan oleh algoritma genetika merupakan solusi optimum global. Solusi Optimum Global merupakan solusi yang terbaik diantara seluruh solusi yang dihasilkan.

8. Kelemahan Algoritma Genetika

Algoritma genetika ini juga masih memiliki kelemahan yaitu ketidakpastian untuk menghasilkan solusi optimum global, karena sebagian besar dari algoritma ini berhubungan dengan bilangan random yang bersifat probabilistik. Peranan parameter di sini adalah memaksimalkan probabilitas dalam menghasilkan solusi optimum global dengan cara membuat suatu skema pengolahan input argumen (fungsi fitness) dan setting parameter yang tepat.

2. Research Method

Genetic algorithm merupakan suatu pepograman komputer yang mensimulasikan proses evolusi biologi. Dalam menyelesaikan suatu masalah akan dihasilkan suatu individu yang telah melewati siklus evolusi biologi, sehingga individu tersebut memiliki nilai fitness yang baik dan dapat menjadi solusi untuk suatu permasalahan. Urutan proses yang dilakukan dalam genetic algorithm untuk penjadwalan kuliah ini adalah:

1. Buat suatu basis data yang menampung data perkuliahan, misalkan tabelnya:

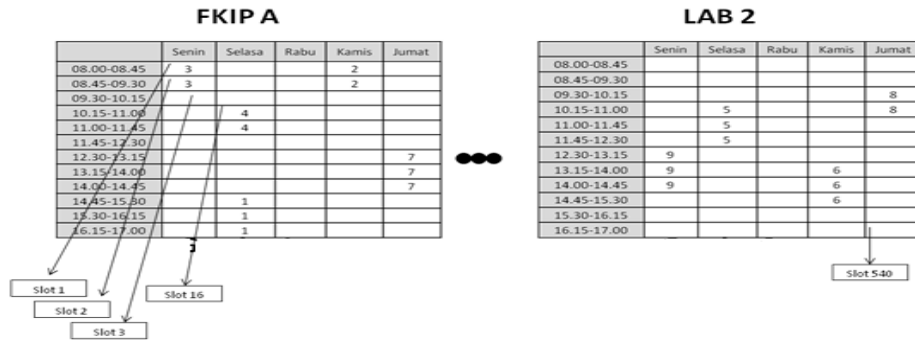
Tabel 2.1 Contoh Tabel Perkuliahan (Jurusan Biologi/Semester III/ Kelas A)

Id_Perkuliahan	Id_MK	Id_Dosen	Jumlah_SKS
1	PBIO 012	DS2	3
2	PBIO 013	DS12	2
3	FKIP 012	DS36	2
4	FKIP 011	DS13	2
5	PBIO 014	DS14	3
6	PBIO 015	DS3	3
7	PBIO 018	DS15	2
8	FKIP 009	DS16	2
9	PBIO 010	DS17	3

2. Buat Slot waktu perkuliahan dengan ukuran hasil kali jumlah jam kuliah, jumlah hari kuliah, dan jumlah ruangan.

$$\begin{aligned} \text{Slot waktu} &= \text{jumlah_jam} \times \text{jumlah_hari} \times \text{jumlah_ruangan} \\ &= 12 \times 5 \times 9 \\ &= 540 \text{ Slot} \end{aligned}$$

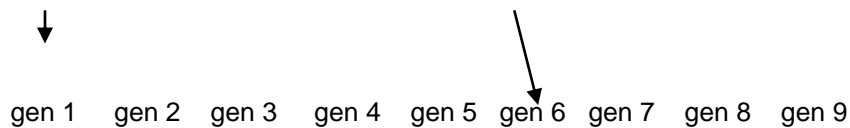
Kemudian ambil ID Perkuliahan dan masukkan kedalam Slot waktu secara acak. Hal ini dilakukan hingga jumlah populasi terpenuhi.



Gambar 2.1 Bentuk Representasi Kromosom 1

3. Masukkan nilai Slot waktu kedalam kromosom yang berukuran jumlah baris di tabel Perkuliahan. Hal ini dilakukan untuk mengefisienkan pemakaian slot pada kromosom. Kromosom dengan 9 gen :

22	37	1	16	495	523	56	530	483
----	----	---	----	-----	-----	----	-----	-----



Posisi gen menyatakan nomor urut pertemuan kuliah Nilai gen menyatakan nomor slot

Gambar 4.2 Bentuk Konversi Kromosom 1

4. Evaluasi nilai Fitness setiap kromosom didalam populasi. Kromosom yang tidak memiliki bentrokkan jadwal akan memiliki nilai fitness tertinggi. Contoh evaluasi nilai fitness pada kromosom diatas:

Tabel 2.2 Score Penempatan Perkuliahan

ID KelasPerkuliahan	Score Penempatan
1	2
2	2
3	2
4	1
5	1
6	2
7	2
8	2
9	2

$$f = \frac{Sk}{Ms}$$

$$= \frac{2+2+2+1+1+2+2+2+2}{9 \times 2}$$

$$= 16 / 18$$

$$= 0,89$$

Lakukan evaluasi fitness kepada setiap kromosom di dalam populasi. Misalkan jumlah populasi adalah 4 kromosom:
 Kromosom[1] = 22, 37, 1, 16, 495, 523, 56, 530, 483
 Kromosom[2] = 12, 33, 312, 18, 49, 53, 6, 50, 83
 Kromosom[3] = 42, 57, 31, 60, 45, 59, 88, 30, 73
 Kromosom[4] = 29, 7, 16, 19, 95, 93, 46, 60, 33

- a) Hasil penghitungan nilai fitness:
 Kromosom[1] = 0,89
 Kromosom[2] = 0,76
 Kromosom[3] = 0,46
 Kromosom[4] = 0,83 +
 Fitness = 2,94

Setelah evaluasi nilai fitness untuk setiap kromosom dilakukan, hitung jumlah nilai fitness seluruh kromosom di dalam populasi

b) Hitung Probabilitas Fitness

$$P[i] = \text{Fitness}[i] / \sum \text{Fitness}$$

$$P[1] = 0,89 / 2,94 = 0,34$$

$$P[2] = 0,76 / 2,94 = 0,26$$

$$P[3] = 0,46 / 2,94 = 0,16$$

$$P[4] = 0,83 / 2,94 = 0,24$$

Perhitungan Probabilitas *fitness* dilakukan untuk membandingkan nilai *fitness* suatu kromosom dengan kromosom lainnya di dalam populasi

c) Cari nilai kumulatif probability dari populasi

$$C1 = P1 = 0,34$$

$$C2 = P1 + P2 = 0,60$$

$$C3 = P1 + P2 + P3 = 0,76$$

$$C4 = P1 + P2 + P3 + P4 = 1$$

5. Seleksi menggunakan metode Roulette Wheel Selection. Langkah-langkah yang harus dikerjakan adalah:

a) Bangkitkan bilangan random sebanyak populasi dengan range 0 s/d 1

$$R[1] = 0,201$$

$$R[2] = 0,284$$

$$R[3] = 0,009$$

$$R[4] = 0,822$$

Perhitungan nilai kumulatif *probability* dari populasi dilakukan untuk mengetahui bahwa total *probability fitness* dari populasi adalah 1

b) Jika $R[k] < C[1]$ maka parent adalah kromosom[1]

Jika $C[k-1] < R[k] < C[k]$ maka parent adalah kromosom[k]

Tabel 2.3 Tabel Seleksi Kromosom untuk menjadi Parent

Kromosom	Nilai R	Range Probability(C)	Parent
1	0,201	- C1	1
2	0,284	- C1	1
3	0,009	- C1	1
4	0,822	C3 - C4	4

c) Hasil Seleksi adalah kromosom yang dipilih untuk menjadi parent, yaitu:

Kromosom[1] = 22, 37, 1, 16, 495, 523, 56, 530, 483
 Kromosom[2] = 22, 37, 1, 16, 495, 523, 56, 530, 483
 Kromosom[3] = 22, 37, 1, 16, 495, 523, 56, 530, 483
 Kromosom[4] = 29, 7, 16, 19, 95, 93, 46, 60, 33

6.) Melakukan crossover dengan nilai Probabilitas Crossover (P_c) yang telah ditentukan. Misalkan, $P_c = 25\% = 0,25$.

a) Bangkitkan bilangan random sebanyak populasi dengan range 0 s/d 1

$$Rc[1] = 0,191$$

$$Rc[2] = 0,259$$

$$Rc[3] = 0,760$$

$$Rc[4] = 0,066$$

Cari k dari nilai $Rc[k]$ yang $\leq P_c$.
 --> $k = 1$ dan 4
 Maka lakukan *crossover two cut point* pada kromosom tersebut dengan menukar nilai pada slot 2 hingga slot 6.

Tabel 2.4 Tabel Crossover antara Parent yang terpilih

Parent 1	Parent 2	Offspring
Kromosom[1] = 22, 37, 1, 16, 495, 523, 56, 530, 483	Kromosom[4] = 29, 7, 16, 19, 95, 93, 46, 60, 33	Offspring[1] = 22, 7, 16, 19, 95, 93, 56, 530, 483
Kromosom[4] = 29, 7, 16, 19, 95, 93, 46, 60, 33	Kromosom[1] = 22, 37, 1, 16, 495, 523, 56, 530, 483	Offspring[4] = 29, 37, 1, 16, 495, 523, 46, 60, 33

b) Populasi setelah crossover

- Kromosom[1] = 22, 7, 16, 19, 95, 93, 56, 530, 483
- Kromosom[2] = 22, 37, 1, 16, 495, 523, 56, 530, 483
- Kromosom[3] = 22, 37, 1, 16, 495, 523, 56, 530, 483
- Kromosom[4] = 29, 37, 1, 16, 495, 523, 46, 60, 33

Nilai kromosom 1 dan kromosom 4 adalah *offspring 1* dan *offspring 4* yang dihasilkan pada tahap *crossover*

7.) Melakukan mutasi dengan nilai Probabilitas Mutasi (Pm) yang telah ditentukan. Misalkan, Pm = 10% = 0,10.

a) Hitung total gen yang ada pada populasi

$$\begin{aligned} \text{Total gen} &= \text{jumlah populasi} \times \text{ukuran kromosom} \\ &= 4 \times 9 \\ &= 36 \text{ gen} \end{aligned}$$

Gen yang di mutasi = Pm x Total gen

$$\begin{aligned} &= 0,10 \times 36 \\ &= 3,6 = 4 \text{ gen (Dibulatkan)} \end{aligned}$$

b) Bangkitkan bilangan random sebanyak gen yang akan dimutasi dengan range 1s/d 36 (Batas range merupakan panjang total gen)

- Rm[1] = 6
- Rm[1] = 21
- Rm[1] = 26
- Rm[1] = 34

Nilai random sebagai penentuan gen yang akan dimutasi. Rm[1] = 6 berarti bahwa gen ke-6 di kromosom 1 akan dimutasi.

c) Bangkitkan bilangan random sebanyak gen yang akan dimutasi dengan range 1s/d 36 (Batas range merupakan panjang total gen)

- Rm[1] = 9
- Rm[1] = 25
- Rm[1] = 28
- Rm[1] = 36

Nilai random sebagai penentuan tujuan gen yang akan dimutasi. Rm[1] = 9 berarti gen yang dimutasi akan dipindahkan ke gen 9 di kromosom 1

d) Populasi hasil mutasi:

- Kromosom[1] = 22, 7, 16, 19, 95, 483, 56, 530, 93
- Kromosom[2] = 22, 37, 1, 16, 495, 523, 56, 530, 483
- Kromosom[3] = 22, 37, 56, 16, 495, 523, 1, 29, 483
- Kromosom[4] = 530, 37, 1, 16, 495, 523, 33, 60, 46

Setelah melakukan mutasi, maka dihasilkan populasi baru. Pada setiap kromosom didalam populasi baru akan dilakukan evaluasi nilai fitness. Siklus ini terus berulang hingga kondisi selesai terpenuhi. Misalkan pada perhitungan diatas, kromosom[1] adalah kromosom yang memiliki nilai fitness terbaik, maka output jadwal perkuliahannya adalah:

A	Senin	Selasa	Rabu	Kamis	Jumat
08.00-08.45					
08.45-09.30			3		
09.30-10.15			3		
10.15-11.00					
11.00-11.45					
11.45-12.30			4		
13.00-13.45	2		4		
13.45-14.30	2				7
14.30-15.15					7
15.15-16.00		1			
16.00-16.45		1			
16.45-17.30		1			

B	Senin	Selasa	Rabu	Kamis	Jumat
08.00-08.45					
08.45-09.30					
09.30-10.15					
10.15-11.00					
11.00-11.45					
11.45-12.30					
13.00-13.45					
13.45-14.30					9
14.30-15.15					9
15.15-16.00					9
16.00-16.45					5
16.45-17.30					5

LAB 2	Senin	Selasa	Rabu	Kamis	Jumat
08.00-08.45					
08.45-09.30					8
09.30-10.15	6				8
10.15-11.00	6				
11.00-11.45	6				
11.45-12.30					
13.00-13.45					
13.45-14.30					
14.30-15.15					
15.15-16.00					
16.00-16.45					
16.45-17.30					

Gambar 2.3 Bentuk Kromosom 1 yang merupakan kromosom terbaik

3. Results and Analysis

3.1 Tampilan Main Menu

Jika inputan username dan password sudah sesuai dengan data di database, maka akan ditampilkan menu utama aplikasi penjadwalan kuliah menggunakan genetic algorithm.



Gambar 3.1 Tampilan Main Menu

Pada menu utama, terdapat tiga pengelompokan, yaitu :

1. Fungsi Utama : Berisi fitur utama seperti proses penjadwalan kuliah, pengecekan penempatan jadwal yang bentrok dan lihat hasil jadwal kuliah.
2. Cetak Report : Berisi fitur untuk mencetak daftar dosen, daftar mata kuliah, jadwal dosen dan jadwal perkuliahan.
3. Olah Data : Berisi fitur untuk penyimpanan dan pengeditan data perkuliahan seperti data jam, data hari, data mata kuliah, data dosen, data slot waktu dan data user.

3.2 Fitur Menu Olah Data

A. Input Data Perkuliahan

Setelah ditekan tombol “Data Perkuliahan” pada Menu Olah Data, maka akan ditampilkan form input data perkuliahan seperti gambar 3.2. Dimana data yang perlu diinputkan adalah data kelas, data mata kuliah dan data dosen.



Gambar 3.2 Tampilan Form Data Perkuliahan

Apabila ditekan tombol "Ambil Data" pada form input data perkuliahan, akan ditampilkan data yang telah ada seperti gambar 3.3 berikut. Ketika menekan tombol "Ok" maka data yang dipilih akan di inputkan ke dalam form.



Gambar 3.3 Tampilan Form Input Data Perkuliahan

B. Input Data Slot Waktu

Setelah ditekan tombol "Data Slot Waktu" pada Menu Olah Data, maka akan ditampilkan form data slot waktu yang memungkinkan untuk digunakan dalam penjadwalan seperti gambar 3.4. Dimana data yang digunakan adalah data hari perkuliahan, data jam perkuliahan dan data ruangan.



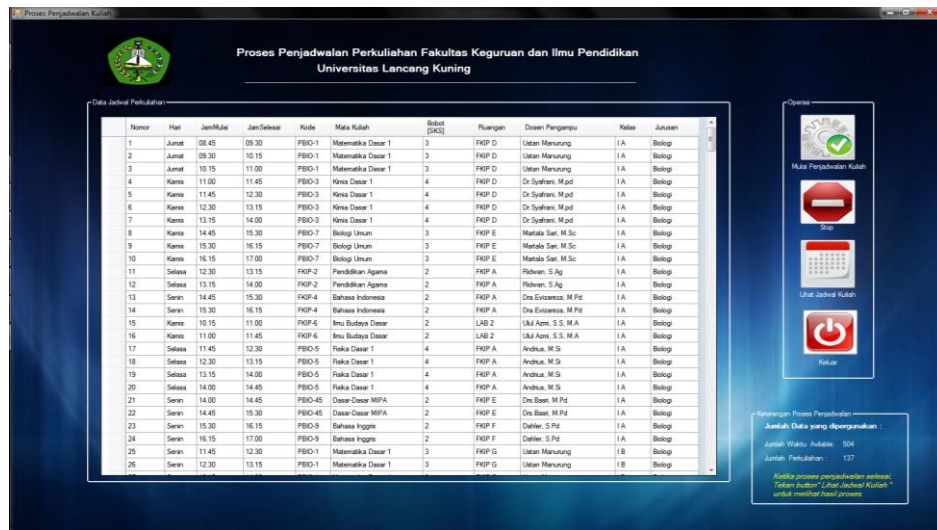
Gambar 3.4 Tampilan Form Data Slot Waktu

Apabila ditekan tombol "Prose Slot Waktu" maka akan dimulai pengisian data slot waktu secara otomatis melalui proses looping. Jumlah record tabel ini tergantung pada data hari, data jam dan data ruangan.

3.3 Fitur Menu Fungsi Utama

A. Proses Penjadwalan Kuliah

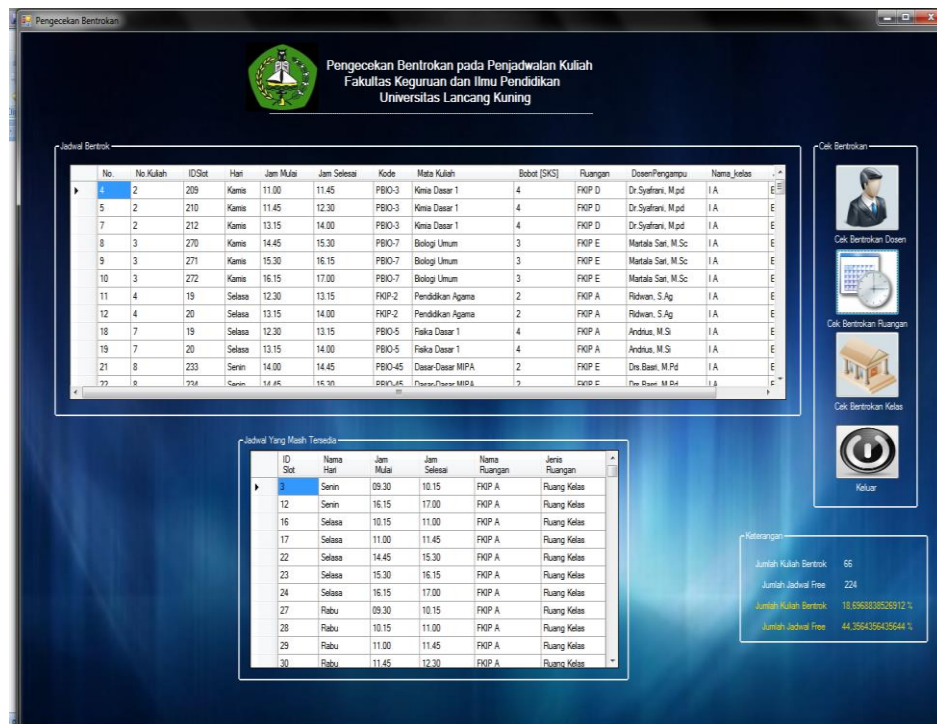
Setelah ditekan tombol "Penjadwalan Kuliah" pada Menu Utama, maka akan ditampilkan form proses penjadwalan kuliah seperti gambar 3.4. Pada form ini di tampilkan jumlah slot waktu perkuliahan dan jumlah perkuliahan yang akan di jadwalkan. Ketika ditekan tombol "Mulai Penjadwalan Kuliah", maka akan dilakukan proses penjadwalan kuliah menggunakan genetic algorithm.



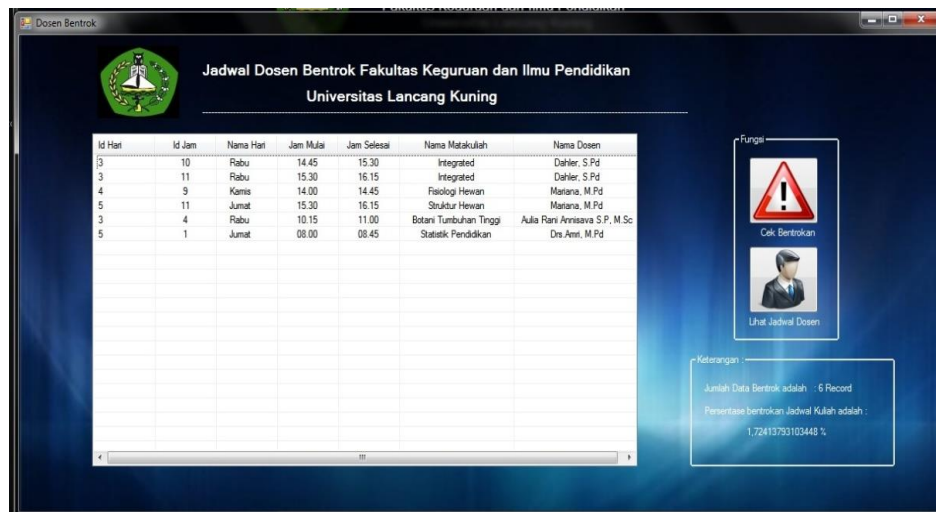
Gambar 3.4 Form proses penjadwalan menggunakan Genetic Algorithm

B. Cek Bentrokan

Setelah ditekan tombol “Cek Bentrokan” pada Menu Fungsi Utama, maka akan ditampilkan form Cek Bentrokan yang digunakan untuk mengecek bentrokan pada sisi dosen, ruangan dan kelas. Form ini menampilkan jadwal yang bentrok dan jadwal yang masih tersedia untuk menjadwalkan kembali jadwal yang bentrok tadi, persentase bentrokan dan persentase jadwal tersedia. Form ini seperti pada gambar 3.5.

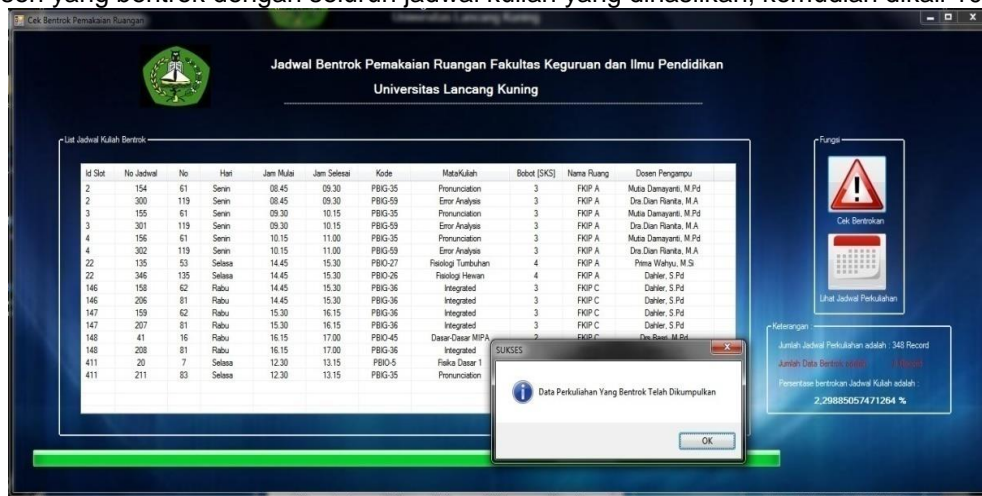


Gambar 3.5 Tampilan form pengecekan bentrokan perkuliahan



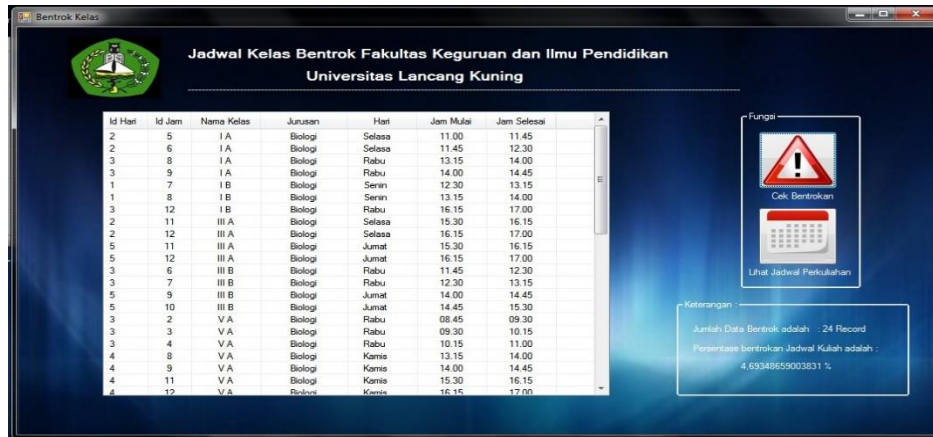
Gambar 3.6 Tampilan form pengecekan bentrokan dosen

Pada form pengecekan bentrokan dosen ini ditampilkan jumlah jadwal dosen yang bentrok. Record ini didapatkan dari pengecekan jadwal dosen pada seluruh hari dan jam. Apabila terdapat jadwal di hari dan jam yang sama, maka jadwal tersebut akan ditampilkan disini. Persentase bentrokan dosen diperoleh dari hasil pembagian antara jumlah jadwal dosen yang bentrok dengan seluruh jadwal kuliah yang dihasilkan, kemudian dikali 100%.



Gambar 3.7 Tampilan form pengecekan bentrokan pemakaian ruangan

Pada form pengecekan bentrokan ruangan ini ditampilkan jumlah jadwal pemakaian ruangan yang bentrok. Record ini didapatkan dari pengecekan jadwal pemakaian ruangan pada seluruh hari dan jam. Apabila terdapat jadwal perkuliahan yang memakai suatu ruangan di hari dan jam yang sama, maka jadwal tersebut akan ditampilkan disini. Persentase bentrokan ruangan diperoleh dari hasil pembagian antara jumlah pemakaian ruangan yang bentrok dengan seluruh jadwal kuliah yang dihasilkan, kemudian dikali 100%.



Gambar 3.8 Tampilan form pengecekan bentrokan kelas

Pada form pengecekan bentrokan jadwal kelas ini ditampilkan jumlah jadwal kelas yang bentrok. Record ini didapatkan dari pengecekan jadwal kelas pada seluruh hari dan jam. Apabila terdapat jadwal kelas yang melakukan perkuliahan di hari dan jam yang sama, maka jadwal tersebut akan ditampilkan disini. Persentase bentrokan jadwal kelas diperoleh dari hasil pembagian antara jumlah jadwal kelas yang bentrok dengan seluruh jadwal kuliah yang dihasilkan, kemudian dikali 100%.

3.4 Fitur Cetak Report

A. Cetak Jadwal Kuliah

Setelah ditekan tombol “Cetak Jadwal Kuliah” pada Menu Cetak Report, maka akan ditampilkan form Cetak Jadwal Kuliah yang memungkinkan untuk digunakan dalam mencetak Jadwal Kuliah seperti gambar 3.9.



Gambar3.9 Tampilan crystal report daftar jadwal kuliah

4. Conclusion

Berdasarkan penelitian yang dilakukan, maka dapat diambil beberapa kesimpulan sebagai berikut:

1. Masalah penjadwalan kuliah dapat direpresentasikan ke dalam bentuk kromosom yang merupakan permodelan dari genetic algorithm dan selanjutnya dapat dibuat menjadi sebuah program setelah disusun algoritma yang merupakan pemenuhan aturan-aturan penjadwalan dan parameter-parameter genetic algorithm yang dipergunakan.
2. Berdasarkan pengujian pada parameter genetika dapat disimpulkan:
 - a) Jumlah generasi dan jumlah populasi yang besar (lebih dari 100) dapat menghasilkan solusi yang lebih baik walaupun memerlukan waktu proses yang relatif lebih lama.

- b) Probabilitas crossover yang besar (lebih dari 50 %) dapat menghasilkan persentase bentrokan yang lebih banyak.

References

- [1] Jankovic, Mladen. (2008). Making Class schedule using genetic algorithm. Diambil 8 November 2010 dari <http://www.codeproject.com/KB/articles/GaClassSchedule.asp>
- [2] Kusumadewi, Sri. (2003). Artificial Intelligence (Teknik dan Aplikasinya). Yogyakarta: Graha Ilmu.
- [3] Kuswadi, Son. (2007). Kendali Cerdas, Teori dan Aplikasi Praktisnya. Yogyakarta: Andi Offset.
- [4] Sukmawan, Budi. (2003). Sekilas Tentang Algoritma Genetika dan Aplikasinya pada Optimasi Jaringan Pipa Air Bersih. Diambil 4 November 2010 dari <http://www.bimacipta.com/ga.htm>.
- [5] Sutanto, Charly. (2009). Penyelesaian Knapsack Problem menggunakan Algoritma Genetika. Pekanbaru: Politeknik Caltex Riau.
- Suyanto. (2008). Evolutionary Computation: Komputasi Berbasis Evolusi dan Genetika. Bandung: Informatika.