

IMPLEMENTASI DUA MODEL CROSSOVER PADA ALGORITMA GENETIKA UNTUK OPTIMASI PENGGUNAAN RUANG PERKULIAHAN

I Wayan Supriana¹, Made Agung Raharja², I Made Satria Bimantara³, Devan Bramantya⁴

^{1,2,3,4}Program Studi Informatika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Udayana
Bukit Jimbaran, Bali, Indonesia

e-mail: wayan.supriana@unud.ac.id¹, made.agung@unud.ac.id², satria_Md@yahoo.com³,
Devanbramantya.3@gmail.com⁴

Received : January, 2021

Accepted : February, 2021

Published : October, 2021

Abstract

The lecture mapping process is often hampered by the number and capacity of rooms, this condition often occurs because of the many obstacles that must be fulfilled. For example, there are courses offered in one semester that cannot be slots in space and time and the lecturer can teach at the same time for different courses. This is experienced by the Informatics Engineering Study Program of the Faculty of Mathematics and Natural Sciences, Udayana University, which offers a fairly large subject in each semester, causing optimization of the lecture space to often experience problems. The Genetic Algorithm (GA) is a model in the optimization of lecture space based on the natural selection mechanism through; coding problem, generate initial population, calculate fitness value, selection, crossover, mutation and optimal population. In this research, the optimization process implements two crossover models in the genetic algorithm, namely the n-point crossover and the cycle crossover. Based on the research that has been carried out, two crossover models provide optimal space usage mapping. From testing the n-point crossover model system gives the best fitness 1 in the 227 generation with a computation time of 5,38 while the cycle crossover model produces the best fitness 1 in the 345 generation with a computation time of 9,27.

Keywords: optimization, genetic algorithms, lecture rooms

Abstrak

Proses pemetaan perkuliahan seringkali terhambat dengan jumlah dan kapasitas ruangan, kondisi ini sering terjadi karena banyaknya kendala yang harus dipenuhi. Misalnya terdapat matakuliah yang ditawarkan di satu semester yang tidak dapat slots ruangan dan waktu serta dosen dapat mengajar pada waktu yang sama untuk matakuliah yang berbeda. Hal ini dialami oleh Program Studi Teknik Informatika Fakultas MIPA Universitas Udayana yang menawarkan matakuliah yang cukup besar di setiap semester, menyebabkan optimasi ruang perkuliahan sering mengalami kendala. Algoritma Genetika (AG) merupakan salah satu model dalam optimasi ruang perkuliahan yang didasari oleh mekanisme seleksi alamiah melalui; pengkodean permasalahan, generate populasi awal, calculate nilai fitness, seleksi, crossover, mutasi dan populasi optimal. Pada penelitian ini proses optimasi mengimplementasikan dua model crossover pada algoritma Genetika yaitu n-point crossover dan cycle crossover. Berdasarkan penelitian yang sudah dilakukan dua model crossover memberikan pemetaan penggunaan ruang secara optimal. Dari pengujian sistem model n-point crossover memberikan fitness terbaik 1 pada generasi ke 227 dengan waktu komputasi 5,38 sementara model cycle crossover menghasilkan fitness terbaik 1 pada generasi 345 dengan waktu komputasi 9,27.

Kata Kunci: optimasi, algoritma genetika, ruang perkuliahan

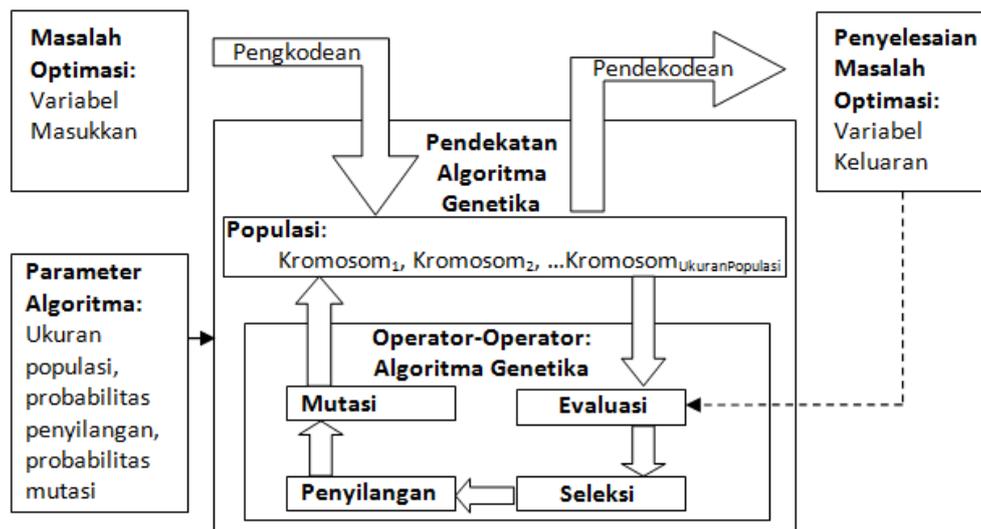
1. PENDAHULUAN

Pelaksanaan perkuliahan program studi Teknik Informatika setiap awal semester selalu didahului dengan melakukan pemetaan matakuliah, dosen, mahasiswa, waktu dan ruangan. Permasalahan klasik selalu muncul dalam proses pemetaannya, seperti sulitnya optimasi ruangan dan waktu karena banyak memiliki variabel kendala yang harus terpenuhi. Kondisi alokasi kapasitas kelas yang kurang tepat pada mata kuliah tertentu. Misalnya pada mata kuliah wajib yang kemungkinan diikuti oleh banyak peserta matakuliah dalam satu semester tertentu dan hanya mempunyai kapasitas ruang kelas kecil, sehingga akan menjadi permasalahan bagi mahasiswa. Hal ini memungkinkan banyak mahasiswa yang tidak dapat terdaftar pada matakuliah tersebut karena kapasitas kelas telah penuh. Sementara mereka harus menunggu semester berikutnya untuk dapat mengambilnya dan akan menyebabkan beban perkuliahan mahasiswa semakin padat. Disamping itu, beberapa kali terdapat mata kuliah wajib atau bahkan mata kuliah pilihan pada semester sama mengalami benturan penggunaan ruangan. Alokasi penggunaan ruangan perkuliahan dilakukan pada saat proses penjadwalan yang masih belum optimal, menyebabkan beberapa matakuliah mengalami benturan pelaksanaan. Proses pengaturan ruangan perkuliahan melalui proses penjadwalan saat ini juga memiliki kelemahan yang lain seperti beban perkuliahan mahasiswa tidak merata dalam tiap harinya. Terdapat satu hari tertentu perkuliahan sangat padat dibandingkan dengan hari lainnya, sehingga mahasiswa akan kesulitan mengatur waktu perkuliahan. Selain kendala tersebut pada sisi dosen yang mengajar lebih dari satu matakuliah dan tidak bisa mengajar pada hari tertentu karena ada penugasan prodi.

Berdasarkan uraian di atas, maka dilakukanlah sebuah penelitian sehingga dapat dibuat optimasi penggunaan ruang perkuliahan yang terbaik dengan permasalahan benturan perkuliahan yang dapat dihindari. Optimasi ruangan dalam kegiatan perkuliahan merupakan kegiatan pengalokasian sejumlah sumber daya dalam jangka waktu tertentu ke dalam berbagai operasi yang mungkin dengan batasan atau syarat tertentu yang harus dipenuhi sehingga fungsi objektif sedekat

mungkin dapat terpenuhi [1]. Optimasi memiliki tujuan memaksimalkan dengan biaya yang minimal melalui pendekatan matematis seperti menggunakan algoritma optimasi [2]. Salah satu algoritma optimasi yang sering digunakan adalah algoritma genetika, merupakan algoritma yang meniru proses seleksi alam serta dapat menyelesaikan permasalahan optimasi yang kompleks dan memiliki model pencarian yang sangat luas [3]. Tidak seperti metode metaheuristik lainnya dengan sebuah solusi optimal pada satu waktu, algoritma genetika mengandung banyak solusi individu pada sebuah populasi. Individu akan dipilih dari populasi kemudian dikawinkan untuk membangun individu baru pada generasi berikutnya sampai menemukan individu paling optimal [4].

Sampai saat ini telah banyak dilakukan penelitian untuk memaksimalkan proses penjadwalan perkuliahan. Pada penelitian sebelumnya dilakukan optimasi penjadwalan mata kuliah pada perguruan tinggi dengan menggunakan algoritma genetika [5]. Melalui penelitian tersebut penjadwalan yang dihasilkan mendekati kondisi ideal yang diinginkan dengan jumlah penalti kendala *hard* sebesar 0 dan penalti kendala *soft* sebesar 8. Penelitian berikutnya mengenai rancangan aplikasi penjadwalan perkuliahan menggunakan metode *brute force* [5], hasil yang diperoleh algoritma *brute force* berhasil memperbaiki kesalahan-kesalahan yang ditimbulkan proses penjadwalan secara manual. Penelitian yang lainnya yaitu penyusunan penjadwalan ujian menggunakan algoritma *rank base ant system* [6], hasil yang diperoleh bahwa algoritma *rank base ant system* dapat memberikan solusi untuk masalah penjadwalan ujian mata kuliah dengan menghasilkan suatu jadwal ujian mata kuliah yang optimal. Sementara penelitian yang akan dilakukan menerapkan algoritma genetika untuk melakukan optimasi penggunaan ruang perkuliahan. Dua model *crossover* akan diimplementasikan untuk melihat perbandingan waktu optimasi. Luaran yang dihasilkan dari penelitian ini adalah optimasi penggunaan ruang perkuliahan berupa daftar matakuliah, waktu dan dosen. Sistem memiliki input berupa matakuliah, jumlah mahasiswa dan semesternya, dosen pengampu matakuliah, jumlah ruangan dan kapasitasnya.



Gambar 1. Kerangka Kerja Algoritma Genetika [Sumber: [4]].

Algoritma genetika (AG) adalah algoritma pencarian yang didasarkan pada mekanisme seleksi alamiah dan genetika [7]. Langkah prosedur algoritma ini diawali dengan menentukan suatu set solusi potensial dan melakukan perubahan dengan beberapa perulangan (iterasi) dengan algoritma genetika untuk menghasilkan solusi terbaik. AG memiliki tujuh buah komponen [8] yaitu, skema pengkodean merupakan teknik penyandian meliputi penyandian gen dari kromosom, nilai *fitness* merupakan evaluasi berdasarkan fungsi tertentu sebagai suatu ukuran performansinya, seleksi merupakan pemilihan dua buah kromosom sebagai orang tua yang akan mengalami *crossover*. *Crossover* merupakan persilangan dua kromosom yang menghasilkan

kromosom anak, mutasi merupakan teknik untuk memperbaiki beberapa gen dalam kromosom, *elitisme* merupakan menjaga agar individu bernilai *fitness* tertinggi tersebut tidak hilang, pergantian populasi merupakan iterasi yang dilakukan untuk mendapatkan populasi optimal [9].

2. METODE PENELITIAN

2.1 Data Penelitian

Sumber data dalam penelitian ini adalah data matakuliah pada kurikulum yang diselenggarakan pada Program studi Teknik Informatika, Fakultas MIPA Universitas Udayana tahun ajaran 2019-2020. Tabel 1 menyajikan data penelitian.

Tabel 1. Data Penelitian

Parameter Perkuliahan	Semester Ganjil 2019-2020	Semester Genap 2019-2020
Mata kuliah	125	115
Dosen	24	24
Angkatan	5	5
Kelas	14	14
Waktu	15	15
Hari	5	5
Ruang	12	12

2.2 Desain Optimasi Penelitian

Model optimasi merupakan penugasan satu set kejadian pada set sumber daya sehingga dapat memenuhi batasan yang sudah ditentukan [10]. Batasan ini terdiri dari *hard constraint* serta *soft constraint*, yang mana *hard constraint* merupakan batasan yang harus

dipenuhi dalam optimasi ruang perkuliahan sehingga memiliki tingkat lebih tinggi dibandingkan *soft constraint* yang sebagai kendala optional [11]. Validitas sebuah solusi apabila semua *hard constraint* tidak ada yang dilanggar. Dalam optimasi ruang perkuliahan pada penelitian ini ditentukan *point-point hard*

constraint dan *soft constraint*, Hc_1: Dalam satu slots waktu yang sama mahasiswa hanya mengikuti satu matakuliah, Hc_2: Ruang memiliki kapasitas yang cukup dalam melaksanakan perkuliahan pada seluruh matakuliah yang dipetakan, Hc_3: Matakuliah yang diselenggarakan pada satu ruangan hanya satu matakuliah dalam slot waktu yang sama, Hc_4: Dosen yang mengajar lebih dari satu matakuliah tidak dipetakan dalam slot waktu yang sama, Hc_5: Matakuliah diselenggarakan dalam waktu yang sudah ditetapkan, Sc_1: Mahasiswa yang mengambil matakuliah tidak ada mendapatkan satu matakuliah setiap hari, Sc_2: Mahasiswa yang mengambil matakuliah tidak ada dapat perkuliahan pada waktu terakhir setiap perkuliahannya

Berdasarkan *hard constraint* dibentuk fungsi kendala yaitu $H_1(n)$, $H_2(n)$, $H_3(n)$, $H_4(n)$, $H_5(n)$. *Soft constraint* dengan fungsi kendala $S_1(n)$ dan $S_2(n)$. Pinalti yang terjadi pada setiap pemetaan matakuliah akan dihitung dengan kondisi persamaan 1 dan persamaan 2 dengan n adalah kuliah ke- n .

$$\begin{aligned} \text{hard}(j) &= H_1(n) + H_2(n) + H_3(n) + H_4(n) \\ &\quad + H_5(n) \\ &= \sum_{i=1}^5 H_i(n) \end{aligned} \quad (1)$$

$$\text{soft}(j) = S_1(n) + S_2(n) = \sum_{i=1}^2 S_i(n) \quad (2)$$

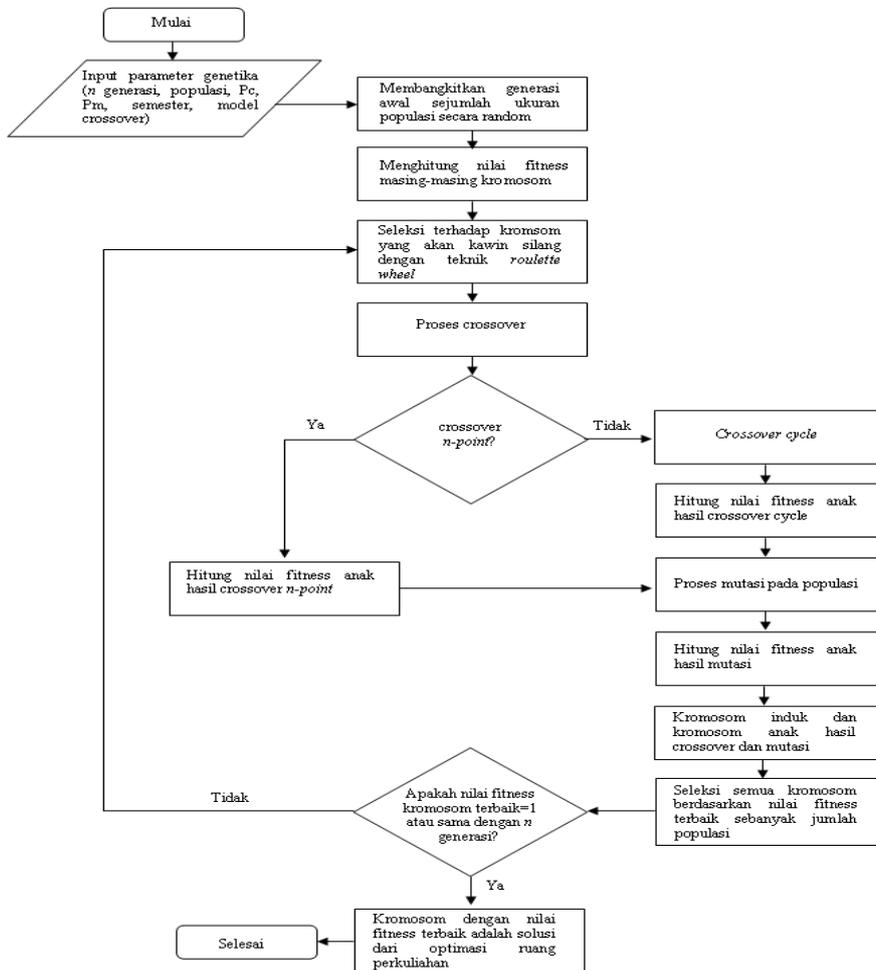
$$\text{Fungsi Tujuan} = \text{hard}(j) + \text{soft}(j) \quad (3)$$

Hard(j): pinalti kendala *hard* untuk jadwal ke- j

Soft(j): pinalti kendala *soft* untuk jadwal ke- j

2.3 Algoritma Genetika Optimasi

Algoritma genetika merupakan teknik pencarian yang meniru seleksi alamiah dan genetika makhluk hidup. Tahapan proses algoritma ini dimulai dengan *generate* set solusi awal dan akan terus diperbaiki selama proses iterasi dengan teknik genetika dalam menentukan solusi terbaik [12]. Set solusi ini digenerate secara random yang berupa kombinasi angka atau karakter. Prosedur algoritma genetika dalam optimasi ruang memiliki alur seperti Gambar 2.



Gambar 2. Flowchart Algoritma Genetika Untuk Optimasi Ruang

Flowchart pada Gambar 2 sebagai proses optimasi ruang memiliki struktur genetika sebagai berikut: (1) representasi kromosom digunakan untuk merepresentasikan masalah riil berupa kode sehingga dapat diproses dalam mesin. Model pengkodean kromosom yang digunakan dalam penelitian ini adalah pengkodean nilai. Susunan kromosom hanya terdiri dari hari, jam dan ruangan, sementara untuk matakuliah, dosen dan kelas/mahasiswa tidak dimasukkan dalam pengkodean kromosom karena sudah dilakukan sebelum pembuatan jadwal melalui pembagian matakuliah dan dosen pengampu oleh program studi. Gambar 3 merupakan kromosom yang dikelompokkan dengan 4 model gen yaitu kelompok pertama adalah matakuliah wajib 3

sks, kelompok kedua matakuliah pilihan 3 sks, kelompok ketiga matakuliah wajib 2 sks dan kelompok keempat matakuliah praktikum 1 sks. Model ini dibentuk untuk meminimalkan terjadinya penalti pada kendala *hard*, matakuliah wajib 3 sks dan 2 sks pelaksanaannya dilakukan pada ruangan yang memiliki kapasitas 35 sampai 40, matakuliah pilihan 3 sks dan matakuliah praktikum 1 sks pelaksanaannya di ruangan yang memiliki kapasitas 10 sampai 25. Sementara jam pelaksanaan untuk matakuliah wajib 3 sks dan pilihan 3 sks dilaksanakan pada slot waktu yang sama, jam pelaksanaan untuk matakuliah wajib 2 sks dan praktikum 1 sks dalam pelaksanaan memiliki slot waktu yang sama.

	Gen Matakuliah 3 sks wajib			Gen Matakuliah 3 sks pilihan			Gen Matakuliah 2 sks wajib		Gen Matakuliah 1 sks															
	Gen 1	Gen 2	Gen 3	Gen 4	Gen 5	Gen 6	Gen 7	Gen 8	Gen 9	Gen 10														
Matakuliah	Matakuliah a		Matakuliah b		Matakuliah c		Matakuliah d		Matakuliah e		Matakuliah f		Matakuliah g		Matakuliah h									
Dosen	Dosen 1		Dosen 2		Dosen 3		Dosen 4		Dosen 5		Dosen 6		Dosen 7		Dosen 8									
Kelas	2019a		2019b		2018a		2018b		2017a		2017b		2017a		2017b									
Kromosom 1	h1	j1	r1	h2	j2	r4	h3	j3	r7	h2	j1	r8	h2	j4	r5	h5	j6	r4	h3	j5	r9	h4	j7	r12
Kromosom 2	h5	j3	r6	h3	j2	r1	h4	j3	r11	h4	j3	r10	h3	j7	r3	h3	j6	r5	h2	j6	r10	h1	j6	r11
Kromosom 3	h3	j2	r5	h4	j3	r2	h5	j3	r9	h2	j1	r11	h2	j5	r2	h5	j4	r4	h3	j6	r9	h3	j4	r12
Kromosom 4	h2	j3	r4	h5	j2	r6	h3	j3	r10	h2	j3	r11	h3	j6	r5	h5	j5	r2	h3	j5	r12	h2	j5	r8
Kromosom 5	h3	j2	r3	h1	j1	r3	h4	j1	r8	h1	j2	r8	h5	j4	r3	h1	j7	r6	h1	j7	r11	h3	j5	r9
Kromosom 6	h4	j1	r4	h2	j2	r5	h1	j1	r12	h3	j3	r9	h4	j5	r2	h2	j5	r3	h2	j5	r8	h5	j4	r12

Gambar 3. Representasi Kromosom

Kromosom 1 merepresentasikan optimasi ruangan, h1 merepresentasikan hari pertama, j1 merepresentasikan slot jam pertama, r1 merepresentasikan ruang pertama. (2) nilai *fitness* digunakan untuk mengevaluasi keunggulan dari sebuah kromosom [13]. Fungsi yang digunakan menghitung nilai *fitness* seperti persamaan 4. Apakah sebuah kromosom baik atau buruk dilihat dari nilai *fitness* kromosom tersebut.

$$Fitness_i = \frac{1}{1 + Fungsi Tujuan} \quad (4)$$

(3) Seleksi dilakukan untuk memilih kromosom dalam satu generasi yang selanjutnya akan mengikuti proses *crossover*. Model seleksi yang digunakan adalah *roulette wheel*, induk dipilih berdasarkan nilai *fitness*, semakin besar nilai *fitness* maka akan semakin besar kemungkinan untuk terpilih. Teknik seleksi ini dilakukan dengan meletakkan semua kromosom pada

roda *roulette*. (4) Proses *crossover* pada optimasi ruang perkuliahan dilakukan dengan mengumpulkan masing-masing *alel* yang sejenis dari masing-masing gen dalam satu kromosom yang terdiri dari hari, jam dan ruang. Terdapat 2 model *crossover* yang digunakan yaitu *n-point crossover* adalah proses *crossover* dilakukan pada *n* titik potong. Pada model ini susunan proses *crossover* pada *alel* waktu dilakukan *crossover 1-point* karena waktu/hari untuk semua kondisi matakuliah adalah sama. *Alel* jam dilakukan *crossover 2-point* karena menggunakan 2 model jam. *Alel* ruang dilakukan *crossover 2-point* karena menggunakan 2 model ruangan. Pada gambar 4 garis putih dan biru menunjukkan titik potong dalam melakukan *crossover*, dimana titik potong dilakukan dengan cara random antara *alel* yang bersesuaian.

Alel Waktu									Alel Jam							Alel Ruang																			
kromosom 1	h1	h2	h3	h2	h2	h5	h3	h4	kromosom 1	j1	j2	j3	j1	j4	j6	j5	j7	kromosom 1	r1	r4	r5	r4	r7	r8	r9	r12	kromosom 3	r5	r2	r1	r1	r9	r7	r8	r10
kromosom 3	h3	h4	h5	h1	h2	h4	h5	h2	kromosom 3	j2	j3	j1	j2	j5	j4	j7	j4	Hasil Persilangan Alel Ruang																	
Hasil Persilangan Alel Waktu									Hasil Persilangan Alel Jam							Hasil Persilangan Alel Ruang Diurutkan Lagi																			
anak 1	h1	h2	h3	h1	h2	h4	h5	h2	anak 1	j1	j2	j1	j2	j4	j4	j7	j4	Anak 1	r1	r4	r7	r8	r1	r1	r9	r10	Anak 2	r5	r2	r9	r7	r5	r4	r8	r12
anak 2	h3	h4	h5	h2	h2	h5	h3	h4	anak 2	j2	j3	j3	j1	j5	j6	j5	j7	Anak 2	r5	r2	r9	r7	r5	r4	r8	r12									

Gambar 4. Model *n-point Crossover*

Kromosom anak 1	h1	j1	r1	h2	j2	r4	h3	j1	r7	h1	j2	r8	h2	j4	r1	h4	j4	r1	h5	j7	r9	h2	j4	r10
Kromosom anak 2	h3	j2	r5	h4	j3	r2	h5	j3	r9	h2	j1	r7	h2	j5	r5	h5	j6	r4	h3	j5	r8	h4	j7	r12

Gambar 5. Kromosom Anak Model *n-point Crossover*

Kromosom anak hasil persilangan pada Gambar 4 disusun dari ketiga jenis *alel* yang disusun berdasarkan kolomnya dari alel waktu, jam dan ruang. Gambar 5 adalah hasil susunan kromosom anak yang dihasilkan. Kedua adalah model *cycle crossover*, melakukan persilangan dengan langkah-langkah sebagai berikut:

pertama temukan *cycle* yang didefinisikan dari relasi posisi *alel* kelompok *alel* tersebut, kedua salin nilai *alel* dalam *cycle* pada kromosom anak dari relasi posisi kromosom induk, ketiga tentukan *alel-alel* diingat yang berasal dari kromosom induk lain dan, keempat isi keturunan dengan *alel-alel* yang diingat tadi.

Alel Waktu									Alel Jam							Alel Ruang																			
kromosom 1	h1	h2	h3	h2	h2	h5	h3	h4	kromosom 1	j1	j2	j3	j1	j4	j6	j5	j7	kromosom 1	r1	r4	r5	r4	r7	r8	r9	r12	kromosom 3	r5	r2	r1	r1	r9	r7	r8	r10
kromosom 3	h3	h4	h5	h1	h2	h4	h5	h2	kromosom 3	j2	j3	j1	j2	j5	j4	j7	j4	kromosom 3	r5	r2	r1	r1	r9	r7	r8	r10									
Pola Alel Waktu									Pola Alel Jam							Pola Alel Ruang																			
h1 --> h3 --> h5 --> h4 --> h2 --> 1									j1 --> j2 --> j3 --> j1 j4 --> j5 --> j7 --> j4							r1 --> r5 --> r1 r7 --> r9 --> r8 --> r7																			
kromosom 1	h1	h3	h2	h5	h4	h2	h3	h4	Kromosom 1	j1	j2	j3	j4	j5	j7	j6	j1	Kromosom 1	r1	r5	r4	r7	r8	r9	r12	r4									
kromosom 3	h3	h5	h1	h4	h2	h5	h4	h2	Kromosom 3	j2	j3	j1	j5	j7	j4	j2	j4	Kromosom 3	r5	r2	r1	r9	r7	r8	r10	r2									
Hasil Persilangan Alel Waktu									Hasil Persilangan Alel Jam							Hasil Persilangan Alel Ruang Diurutkan Lagi																			
Anak 1	h1	h4	h3	h2	h2	h5	h5	h4	Anak 1	j1	j2	j3	j2	j4	j4	j5	j7	Anak 1	r1	r2	r7	r8	r5	r1	r9	r10	Anak 2	r5	r4	r9	r7	r1	r4	r8	r12
Anak 2	h3	h2	h5	h1	h2	h4	h3	h4	Anak 2	j2	j3	j1	j1	j5	j6	j7	j4	Anak 2	r5	r4	r9	r7	r1	r4	r8	r12									

Gambar 6. Model *cycle Crossover*

Kromosom Anak 1	h1	j1	r1	h4	j2	r2	h3	j3	r7	h2	j2	r8	h2	j4	r5	h5	j4	r1	h5	j5	r9	h4	j7	r10
Kromosom Anak 2	h3	j2	r5	h2	j3	r4	h5	j1	r9	h1	j1	r7	h2	j5	r1	h4	j6	r4	h3	j7	r8	h4	j4	r12

Gambar 7. Kromosom Anak Model *cycle Crossover*

Hasil kromosom anak dari persilangan pada Gambar 6 disusun dari ketiga jenis *alel* yang dihasilkan berdasarkan kolomnya dari *alel* waktu, jam dan ruang. Sehingga Gambar 7 adalah hasil susunan kromosom anak yang dihasilkan.

dibandingkan dengan nilai probabilitas mutasi (P_m), apabila nilainya lebih kecil dari P_m maka kromosom tersebut mengalami mutasi dan sebaliknya. Hasil pada tahap mutasi ini akan membentuk anak baru. (6) Update generasi merupakan proses pengelompokan kromosom induk dan kromosom anak dari proses *crossover* serta kromosom anak hasil mutasi. Kumpulan kromosom tersebut dilakukan pengurutan nilai *fitness* dari terbesar sampai terkecil. Dilakukan seleksi sebanyak jumlah kromosom pada populasi yang digunakan

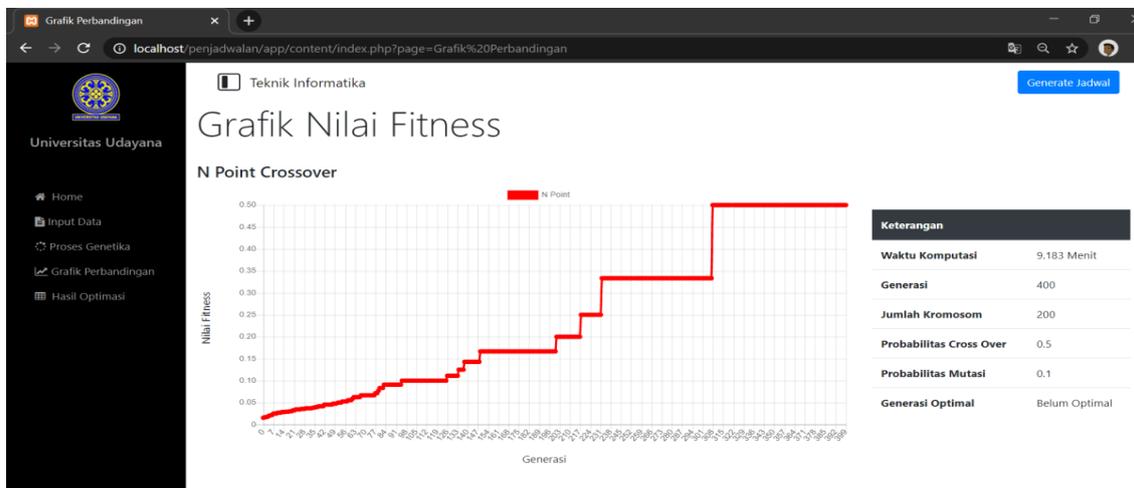
berdasarkan kromosom terbaik. Hasilnya berupa populasi baru yang berisi kromosom terpilih yang kemudian menggantikan populasi sebelumnya. (7) Proses genetika dikatakan optimal apabila nilai *fitness* dari kromosom terbaik adalah 1 pada suatu generasi atau generasi maksimum dari iterasi sesuai dengan yang ditentukan. Nilai *fitness* 1 pada kromosom ini adalah solusi optimasi ruang perkuliahan yang di konversi dalam bentuk tabel, berisi informasi mata kuliah, hari, jam, dosen, kelas dan ruang perkuliahan.

3. HASIL DAN PEMBAHASAN

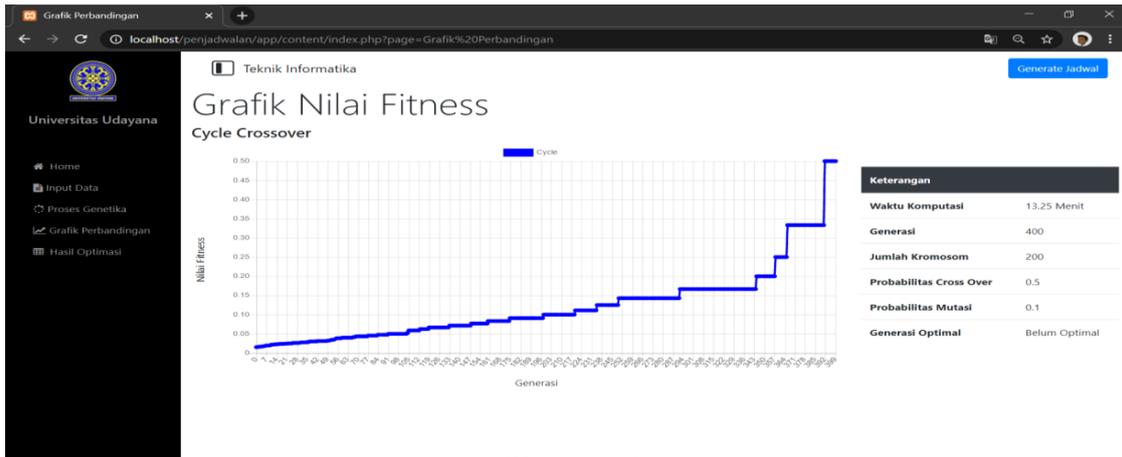
3.1 Hasil Optimasi Ruang Perkuliahan

Hasil optimasi berdasarkan proses genetika yang dilakukan seperti pada Gambar 8, Gambar 9 dan Gambar 10 untuk dua model *crossover* yang digunakan. Input parameter sistem yaitu maksimum generasi, jumlah kromosom, probabilitas *crossover*, probabilitas mutasi, semester dan teknik *crossover*. Optimasi yang dihasilkan berdasarkan iterasi menunjukkan peningkatan nilai *fitness* yang terjadi setiap generasinya.

Gambar 8. Inputan Sistem Optimasi Ruang Perkuliahan



Gambar 9. Grafik Nilai *Fitness* Model *n-point Crossover*



Gambar 10. Grafik Nilai *Fitness* Model *Cycle Crossover*

Hasil yang didapatkan pada Gambar 9 dan Gambar 10 memperlihatkan bahwa generasi ke 400 masih belum optimal yaitu nilai *fitness* terbaik dari kedua model *crossover* baru dapat mencapai 0,5 sehingga optimasi ruang yang dihasilkan masih mengalami benturan antara satu matakuliah dengan matakuliah yang lain. Waktu komputasi untuk 400 generasi pada model *n-point crossover* sebesar 9,183 menit

sementara pada model *cycle crossover* sebesar 13,25 menit. Nilai *fitness* terbaik pada model *n-point crossover* diperoleh pada generasi 308, sementara model *cycle crossover* mencapai *fitness* terbaik pada generasi 392. Hal ini menunjukkan model *n-point crossover* memiliki waktu komputasi lebih singkat dan nilai *fitness* lebih baik dibandingkan dengan *cycle crossover* dengan pengujian nilai parameter yang sama.

No	Hari	Jam	D	E	F	G	H
Jadwal Kuliah Belum Optimal							
1	Senin	08.00-10.30	BD 11 Sistem Digital [2019 C] [BD 11] Komang Ari Mogi, S.Kom., M.Kom	BD 12 Grafika Komputer [2017 B] [BD 12] Gede Arta Wibawa, ST., M.Cs	BC 11 Matematika Diskrit [2019 D] [BC 11] Gede Santi Astawa, ST., M.Cs	BC 22 Desain dan Analisis Algoritma [2018 B] [BC 22] Gusti Agung Gede Arya	BC 21 Interaksi Manusia dan Komputer [2017 A] [BC 21] Dr. Ir. I Ketut Gede Wayan Supriane
2	Senin	10.30-13.00	Logika Informatika [2019 A] [BD 11] Agus Mulliantara, S.Kom., M.Kom				
3	Senin	13.30-16.00	Logika Informatika [2019 B] [BD 11] Agus Mulliantara, S.Kom., M.Kom	Pemrograman Berbasis Web [2017 A] [BD 12] Putu Gede Hendra	Analisis Numerik [2018 C] [BC 11] Dra. Luh Gede Astuti, M.Kom	Teori Bahasa dan Otomata [2017 A] [BC 22] Dr. A.A Istri Ngurah Eka	Program Linear [2018 D] [BC 21] Made Agung Raharja, S.Si., M.Cs
4	Senin	08.00-09.40					
5	Senin	09.40-11.20					
6	Senin	11.20-13.00				Komputer dan Masyarakat [2016 A] [BC 22] Cokorda Rai Adi Pramatha,	Bahasa Inggris [2019 B] [BC 12] Cokorda Rai Adi
7	Senin	13.30-15.10					
8	Selasa	08.00-10.30	Analisis dan Desain Sistem [2017 A] [BD 11] Gusti Agung Gede Arya	Program Linear [2018 C] [BD 12] Luh Arida Ayu Rahning Putri, S.Kom., M.Kom		Analisis Numerik [2018 A] [BC 22] Dr. A A Istri Ngurah Eka Karyawati,	Analisis dan Desain Sistem [2017 B] [BC 21] Cokorda Rai Adi
9	Selasa	10.30-13.00		Program Linear [2018 B] [BD 12] Luh Arida Ayu Rahning Putri, S.Kom., M.Kom		Sistem Operasi [2018 C] [BC 22] Made Agung Raharja, S.Si., M.Cs	
10	Selasa	13.30-16.00		Desain dan Analisis Algoritma [2018 A] [BD 12] Gusti Agung Gede Arya	Basis Data [2018 D] [BC 11] Ida Bagus Gede Dwidasmara, S.Kom., M.Kom	Analisis dan Desain Sistem [2017 C] [BC 22] Cokorda Rai Adi	Basis Data [2018 B] [BC 21] Dr. Ngurah Agus Sanjaya ER, S.Kom., M.Kom
11	Selasa	08.00-09.40					
12	Selasa	09.40-11.20			Bahasa Inggris [2019 D] [BC 11] Gst Ayu Vida Mastrika Giri, S.Kom., M.Kom		
13	Selasa	11.20-13.00	Bahasa Inggris [2019 A] [BD 11] Gst Ayu Vida Mastrika Giri, S.Kom., M.Kom				Metode Penelitian [2017 C] [BC 21] Gusti Anom Cahyadi Putra, ST., M.Kom
14	Selasa	13.30-15.10	Kewarganegaraan [2019 D] [BD 11] Dosen MIPA 1				Komputer dan V [2019 B] [BC 12] Cokorda Rai Adi Pramatha,

Gambar 11. Optimasi Ruang dengan *n-point Crossover*

Luaran sistem yang dihasilkan dalam bentuk excel dari dari model *n-point crossover* pada Gambar 11. Optimasi ruang yang dihasilkan masih belum optimal, ada beberapa benturan pelaksanaan perkuliahan yang masih terjadi seperti matakuliah Pemrograman Berbasis Web dengan matakuliah Teori Bahasa dan Otomata yang yang diambil oleh angkatan 2017 kelas A dilaksanakan pada waktu yang bersamaan. Proses optimasi akan ditentukan berdasarkan parameter input.

3.2 Pembahasan

Optimasi ruang perkuliahan dengan dua model *crossover* yang diimplementasikan pada sistem, dilakukan pengujian dengan mengganti nilai parameter probabilitas *crossover* dan jumlah kromosom dalam satu generasi. Pengujian dilakukan untuk mengetahui kecepatan waktu komputasi dan optimalisasi ruang perkuliahan yang dilakukan dari kedua model *crossover* yaitu *n-point crossover* dan *cycle crossover*. Berikut hasil dari uji coba perbandingan nilai parameter probabilitas

crossover dan jumlah kromosom dengan menetapkan probabilitas mutasi 0.1 dan jumlah

generasi 400 untuk setiap percobaan:

Tabel 2. Hasil Pengujian Model *n-point Crossover*

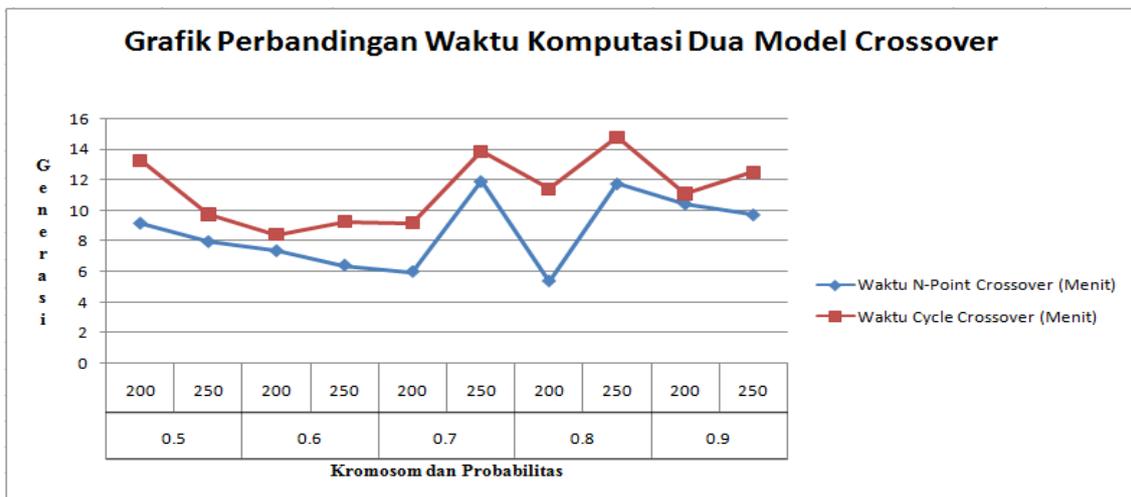
Probabilitas CS	Jumlah kromosom	Generasi Terbaik	Waktu Komputasi (Menit)	Fitness Terbaik	Status Optimasi
0.5	200	400	09,18 Menit	0.5	Belum Optimal
	250	294	07,97 Menit	1	Optimal
0.6	200	322	07,37 Menit	1	Optimal
	250	235	06,42 Menit	1	Optimal
0.7	200	252	06,00 Menit	1	Optimal
	250	400	11,98 Menit	0.5	Belum Optimal
0.8	200	227	05,38 Menit	1	Optimal
	250	400	11,77 Menit	0.5	Belum Optimal
0.9	200	400	10,45 Menit	0.5	Belum Optimal
	250	305	09,73 Menit	1	Optimal

Tabel 3. Hasil Pengujian Model *Cycle Crossover*

Probabilitas CS	Jumlah kromosom	Generasi Terbaik	Waktu Komputasi (Menit)	Fitness Terbaik	Status Optimasi
0.5	200	400	13,25 Menit	0.5	Belum Optimal
	250	390	09,70 Menit	1	Optimal
0.6	200	400	08,40 Menit	0.35	Belum Optimal
	250	345	09,27 Menit	1	Optimal
0.7	200	400	09,13 Menit	0.5	Belum Optimal
	250	400	13,82 Menit	0.5	Belum Optimal
0.8	200	400	11,38 Menit	0.2	Belum Optimal
	250	400	14,77 Menit	0.5	Belum Optimal
0.9	200	361	11,08 Menit	1	Optimal
	250	400	12,48 Menit	0.5	Belum Optimal

Dari pengujian yang telah dilakukan menunjukkan bahwa generasi terbaik dari 400 generasi yang diproses sistem, waktu komputasi yang dibutuhkan dalam proses dan nilai *fitness* terbaik pada setiap proses

percobaan dari kedua model *crossover* yang digunakan. Berdasarkan Tabel 1 dan Tabel 2 hasil optimasi ruang dari dua model *crossover* jika dilihat dari waktu komputasi seperti Gambar 12.



Gambar 12. Perbandingan Waktu Komputasi Dua Model *Crossover*

Pada Gambar 12 dapat diketahui dari dua model *crossover* bahwa model *n-point crossover* memiliki waktu komputasi lebih baik dibandingkan model *cycle crossover* dalam setiap percobaan yang dilakukan. Pada grafik terlihat waktu terbaik dari model *n-point crossover* sebesar 5,38 menit, diperoleh melalui inputan probabilitas *crossover* 0,8 dengan jumlah kromosom sebanyak 200. Sedangkan waktu terbaik dari *cycle crossover* sebesar 9,27 menit, diperoleh melalui inputan probabilitas *crossover* 0,6 dengan jumlah kromosom sebanyak 250.

4. KESIMPULAN

Kesimpulan yang dapat ditarik dari penelitian yang dilakukan adalah (1) dua model *crossover* pada algoritma genetika yang diimplementasikan dapat melakukan proses optimasi ruang perkuliahan dimana model *n-point crossover* lebih baik dari sisi waktu komputasi dibandingkan model *cycle crossover*. (2) Probabilitas *crossover* menentukan nilai *fitness* yang dihasilkan. Semakin tinggi nilai probabilitas *crossover* yang diberikan, maka nilai *fitness* yang dihasilkan akan semakin tinggi. Namun pada kondisi tertentu nilai *fitness* tidak mengalami perubahan atau bahkan akan menurun. (3) Pada proses optimasi ruang perkuliahan nilai probabilitas *crossover* yang sesuai digunakan pada model *n-point crossover* adalah 0,8 dengan waktu komputasi 5,38 menit, sementara pada model *cycle crossover* nilai probabilitas *crossover* yang sesuai digunakan adalah 0,6 dengan waktu komputasi 9,27 menit.

PERNYATAAN PENGHARGAAN

Penulis mengucapkan terimakasih kepada Lembaga Penelitian dan Pengabdian pada Masyarakat (LPPM) Universitas Udayana atas dana hibah penelitian yang diberikan serta teman-teman dosen dan Program Studi Informatika atas kerjasama dan ijin melaksanakan penelitian.

DAFTAR PUSTAKA

- [1] R. A. L. N. Hayati, Ansari, "Sistem Pendukung Keputusan Pemilihan Peserta Olimpiade Mipa Tingkat Sd Menggunakan Metode Saw (Simple Additive Weighting)," *J. Resist.*, vol. 3, no. 2, pp. 82–88, 2018.
- [2] E. Ferawaty, "Optimasi Penjadwalan Mata

Kuliah Pada Perguruan Tinggi Dengan Menggunakan Algoritma Genetika.," 2010.

- [3] E. Suhartono, "Optimasi Penjadwalan Mata Kuliah Dengan Algoritma Genetika (Studi Kasus AMIK JTC Semarang)," *INFOKAM*, vol. II, no. XI, pp. 132–146, 2015.
- [4] A. Qoiriah, "Penjadwalan Ujian Akhir Semester Dengan Algoritma Genetika (Studi Kasus Jurusan Teknik Informatika UNESA)," *J. Manaj. Inf.*, vol. 03, no. 02, pp. 33–38, 2014.
- [5] A. I. K. Dewi, "Rancangan Aplikasi Penjadwalan Perkuliahan Menggunakan Metode Brute Force," *J. Tek. Ibnu Sina*, vol. 3, no. 1, pp. 21–28, 2018.
- [6] H. P. R. F. Astuti, N. Satyahadewi, "Penyusunan Penjadwalan Ujian Menggunakan Algoritma Rank Base Ant System," *Bul. Ilm. Mat. Stat dan Ter.*, vol. 6, no. 02, pp. 151–158, 2017.
- [7] L. Y. Heli, Shanshan, "The Application of Genetic Algorithm Based on Multi-dimension Code Scheme on Course Scheduling In Adult Education," in *Proceedings of the Third International Symposium on Electronic Commerce and Security Workshop (ISECS'10)*, 2010.
- [8] I. P. G. H. Suputra, "Implementasi Algoritma Slope One untuk Rekomendasi Perjalanan Wisata," in *Prosiding Seminar Nasional Teknologi Informasi dan Aplikasinya*, 2018, pp. 33–36.
- [9] Muliadi, "Pemodelan Algoritma Genetikan pada Sistem Penjadwalan Perkuliahan Prodi Ilmu Komputer Universitas Lambungmangkurat," *Kumpul. J. Ilmu Komput.*, vol. 01, no. 01, pp. 67–78, 2014.
- [10] C. R. A. P. I. P. G. H. Suputra, "Rekomendasi Rute Perjalan Wisata Berbasis Web Menggunakan Algoritma Genetika," *J. Ilmu Komput.*, vol. XIII, no. 1, pp. 21–27, 2020.
- [11] S. D. N. L. G. P. Suwirmayanti, I. M. Sudarsana, "Penerapan Algoritma Genetika untuk Penjadwalan Mata Pelajaran," *J. Appl. Intell. Syst.*, vol. 1, no. 3, pp. 220–233, 2016.
- [12] L. H. S. L. W. F. Mahmudi, R. M. Mariana, "Solving Part Type Selection and Loading Problem in Flexible Manufacturing System using Real Coded Genetic Algorithms-Part II: Optimization," *Int. J. Ind. Manuf. Eng.*, vol. 6, no. 9, pp. 1929–1933, 2012.

- [13] W. F. M. N. N. Priandani, "Optimasi Travelling Salesman Problem With Time Windows (TSP-Tw) Pada Penjadwalan Paket Rute Wisata Di Pulau Bali Menggunakan Algoritma Genetika," in *Seminar Nasional Sistem Informasi Indonesia (SESINDO)*, 2015.