

ANALISIS *INSERTION MUTATION* PADA SISTEM PENJADWALAN MATA KULIAH

Gellysa Urva^{*1}, Merina Pratiwi²

Program Studi Teknik Informatika / Sekolah Tinggi Teknologi Dumai
Jl. Utama Karya Bukit Batrem II, Dumai (Riau)
¹*gellysa.urva@gmail.com*, ²*merinapratiwi1920@gmail.com*

Page | 245

Abstrak— Penjadwalan mata kuliah merupakan pekerjaan rutin yang selalu dilakukan setiap perguruan tinggi pada awal semester yang menjadi pekerjaan yang rumit dikarenakan persoalan penjadwalan merupakan masalah kombinatorial yang memiliki batasan-batasan yang harus dipenuhi. Penjadwalan memiliki peranan penting dalam pengelolaan suatu perguruan tinggi dimana masalah penjadwalan termasuk permasalahan kompleks yang menjadi bagian penting dari operasinya. Sistem penjadwalan yang selama ini berjalan di Sekolah Tinggi Teknologi (STT) Dumai dilakukan secara acak yang menyita banyak waktu dan sumber daya. Hal ini mengakibatkan perkuliahan kurang efektif karena harus melakukan penjadwalan ulang sesuai keadaan dan kebutuhan kampus. Oleh karena itu, diperlukan suatu bentuk pengembangan sistem penjadwalan mata kuliah dimana sistem tersebut dapat secara terstruktur membantu menyusun jadwal mata kuliah berdasarkan faktor penentu serta ruang populasi yang ada, dan memungkinkan terjadinya perubahan jadwal yang lebih dinamis, cepat dengan memperhatikan faktor penentu yang sama serta dapat memberikan gambaran kompleksitas permasalahan beban permasalahan terhadap ketersediaan sumber daya di STT Dumai. Pemilihan jenis mutasi dalam algoritma genetika menjadi pilihan yang perlu diperhatikan guna mendapatkan hasil optimal. Hasil penelitian menyatakan sistem penjadwalan mata kuliah dengan *insertion mutation* dapat memenuhi semua batasan dengan hasil yang lebih optimal.

Kata Kunci— Penjadwalan, Algoritma Genetika, *Insertion Mutation*

Abstract— Course scheduling is routine work that is always done at every college at the beginning of the semester which becomes a complicated job because scheduling problems are combinatorial problems that have restrictions that must be met. Scheduling has an important role in the management of a tertiary institution where scheduling problems are including complex issues that are an important part of its operations. The scheduling system that has been running at Sekolah Tinggi Teknologi (STT) Dumai is carried out randomly which takes up a lot of time and resources. This resulted in less effective lectures because they had to reschedule according to the circumstances and needs of the campus. Therefore, it is necessary to develop a course scheduling system where the system can structurally help to arrange course schedules based on determinants and existing population spaces, and allow for more dynamic, faster schedule changes by taking into account the same determinants and can provides an illustration of the complexity of the problem burden on the availability of resources in STT Dumai. The selection of mutations in genetic algorithms is a choice that needs to be considered in order to obtain optimal results. The results of the study stated that the course scheduling system with *insertion mutation* can meet all restrictions.

Keywords— *Schedule, Genetic Algorithm, Insertion Mutation*

I. PENDAHULUAN

Evolusi penjadwalan dan optimasi dari berbagai komponennya merupakan tantangan utama yang perlu menjadi topik penelitian terdepan guna mendapatkan sebuah metode yang tepat untuk solusinya [1]. Penjadwalan mata kuliah merupakan kegiatan rutin dalam sistem akademik di Perguruan Tinggi yang selalu dilakukan dalam menghadapi semester baru serta memiliki peranan penting dalam pengelolaan suatu institusi perguruan tinggi dimana masalah

penjadwalan termasuk permasalahan kompleks yang menjadi bagian penting dari operasinya. Tujuan penjadwalan adalah untuk mengoptimalkan satu atau beberapa tujuan. Inti dari penjadwalan mata kuliah yaitu menjadwalkan sejumlah komponen yang terdiri atas kelas, mata kuliah, dosen, ruang dan waktu dengan sejumlah batasan dan syarat tertentu. Salah satu metode yang dapat dipakai untuk mengatasi permasalahan penjadwalan yaitu dengan pendekatan metaheuristik. Metaheuristik yaitu metode pendekatan yang didasarkan pada metode heuristik

yaitu suatu metode untuk mencari solusi dengan pendekatan komputasi yang memadukan interaksi antara prosedur pencarian lokal dan strategi yang lebih tinggi untuk menciptakan proses yang mampu keluar dari titik-titik lokal optimum dan melakukan pencarian diruang solusi untuk menemukan solusi global dengan cara mencoba secara iteratif untuk memperbaiki kandidat solusi yang diinginkan [2]. Berbagai macam yang tergolong dalam metode metaheuristik salah satu diantaranya yaitu algoritma genetika [3]. Penggunaan algoritma genetika merupakan pilihan yang tepat sebab mampu menyelesaikan masalah penjadwalan kendaraan jalur bus di kota dengan mendapatkan solusi optimal [4]. Algoritma genetika dalam kerangka waktu telah menunjukkan berfungsi lebih baik dan terbukti mampu menyelesaikan permasalahan optimasi [5]. Sistem Pendukung Keputusan (SPK) pembagian beban mengajar dosen dengan metode algoritma genetika menghasilkan distribusi beban mengajar secara otomatis berdasarkan input matriks kompetensi dosen. Sehingga pembagian beban mengajar dosen dapat merata dengan baik [6].

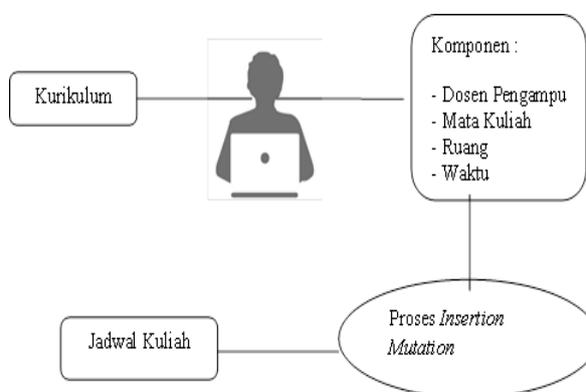
Penentuan parameter-parameter dari proses algoritma genetika mempengaruhi untuk mendapatkan hasil yang optimal. Seperti besarnya populasi, jenis *crossover* serta *mutation rate* berpengaruh terhadap performa algoritma genetika. Pemilihan parameter tersebut yang tidak tepat dapat menyebabkan hasil performa algoritma genetika tidak maksimal [7]. Penggunaan jenis *crossover* dan mutasi yang tepat dalam algoritma genetika menunjukkan hasil efektif dan lebih maksimal dalam mendesain persoalan waktu mengajar di universitas. Proses pembuatan *timetables* mencapai hasil yang efisien serta sangat cepat menemukan nilai *fitness* yang terbaik dalam prosesnya [8]. Pengaruh dari generasi baru menggunakan operator *crossover* menyatakan *Partially Mapped Crossover* (PMX) memberikan jaminan untuk semua posisi kromosom yang disilangkan secara acak menghasilkan keturunan baru yang lengkap dari seluruh gen induk. Dalam kinerjanya PMX merupakan terbaik sebagai *crossover* yang diusulkan untuk memulai persilangan [9]. Salah satu metode metaheuristik yang digunakan adalah algoritma genetika. Konsep dasar algoritma genetika adalah mengelola suatu populasi individu yang mempresentasikan kandidat solusi sebuah penjadwalan Solusi tersebut dievaluasi dengan menggunakan fungsi *fitness*, dan seterusnya dilakukan seleksi, pindah silang (*crossover*), dan *insertion mutation* [10]

Selama ini, pengaturan penjadwalan mata kuliah yang telah berjalan pada Sekolah Tinggi Teknologi (STT) Dumai dilakukan secara acak berdasarkan hari dan sesi oleh ketua program studi di setiap awal semester yang menyita banyak waktu dan sumber daya. Hal ini mengakibatkan perkuliahan di awal semester kurang efektif karena harus melakukan penjadwalan ulang sesuai keadaan dan kebutuhan kampus. Oleh

karena itu, diperlukan suatu bentuk sistem penjadwalan mata kuliah dimana sistem tersebut dapat secara terstruktur membantu menyusun jadwal mata kuliah berdasarkan faktor penentu serta ruang populasi yang ada, dan memungkinkan terjadinya perubahan jadwal yang lebih dinamis, cepat dengan memperhatikan faktor penentu yang sama serta dapat memberikan gambaran kompleksitas permasalahan beban permasalahan terhadap ketersediaan sumber daya yang ada.

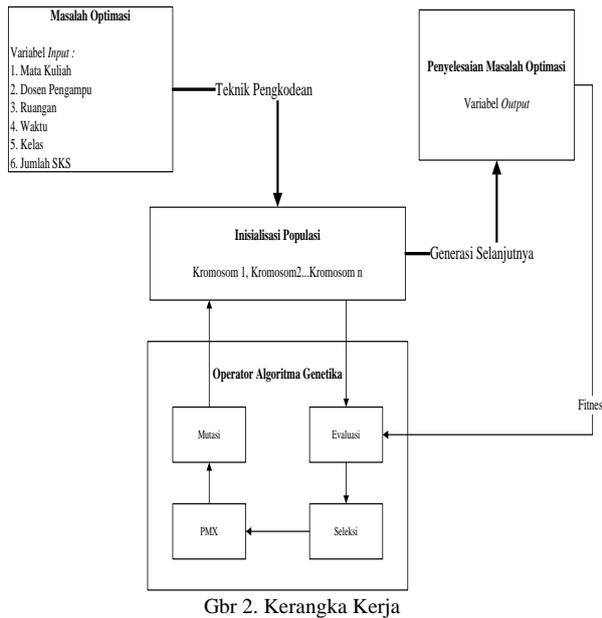
II. METODE PENELITIAN

Rancangan penelitian yang akan dilakukan menggunakan beberapa komponen yang terkait yaitu Dosen pengampu, mata kuliah, ruang dan waktu dengan sejumlah batasan-batasan tertentu, dimana batasan-batasan tersebut ada yang harus dipenuhi atau tidak boleh dilanggar. Untuk selanjutnya diproses melalui *crossover Partially Mapped Crossover* (PMX) dan *Insertion Mutation* guna menghasilkan Penjadwalan mata kuliah yang optimal. Adapun rancangan yang akan dibuat terdapat pada Gambar 1.



Gbr 1. Rancangan Penelitian

Data yang digunakan untuk menyusun jadwal mengajar dosen yaitu data yang terdiri dari mata kuliah, kelas, ruangan, dosen dan jumlah sks. Langkah selanjutnya membentuk kromosom yang direpresentasikan dengan sebuah objek atau *array* yang berisi beberapa data terkait dengan persiapan penyusunan jadwal. Adapun kerangka kerja proses persiapan penyusunan jadwal mata kuliah dapat dilihat pada gambar 2.



Gbr 2. Kerangka Kerja

A. Representasi Kromosom

Representasi kromosom yang dibentuk berdasarkan variabel input yang terdiri dari mata kuliah, dosen pengampu, ruangan, hari, waktu, kelas dan jumlah SKS. Setiap gen terdiri dari enam jenis variabel input tersebut yang dijadikan sebagai sebuah object. Selanjutnya setiap gen tersebut direpresentasikan dalam nomor. Populasi awal dibangkitkan secara acak dari sejumlah individu.

B. Evaluasi Fitness

Pada tahap evaluasi ini melakukan proses evaluasi pada data kromosom terhadap nilai *fitness* yang digunakan. Nilai *fitness* merupakan suatu ukuran baik tidaknya suatu solusi yang dinyatakan sebagai satu individu, atau dengan kata lain nilai *fitness* menyatakan nilai dari fungsi tujuan [11]. Setiap aturan yang dipergunakan dalam penjadwalan dosen pengampu mata kuliah diberi nilai pinalti, dimana semakin wajib dilaksanakan maka nilai pinalti yang diberikan semakin besar. Dari fungsi *fitness* yang digunakan semakin sedikit aturan yang dilanggar, maka semakin besar nilai *fitness*-nya. Pada penelitian ini nilai *fitness* ditentukan dengan formula :

$$F = \frac{1/I}{\sum Pinalty} \quad (1)$$

Dimana penjelasan dari formula tersebut adalah :

F = Fungsi *Fitness*

Pinalty = Pelanggaran yang dilakukan

C. Seleksi

Pada tahap seleksi ini memilih individu- individu yang akan dipilih untuk proses *crossover* dan mutasi. Proses seleksi ini bertujuan memperoleh calon parent yang baik. parent yang baik akan menghasilkan keturunan yang baik [12]. Nilai *fitness* inilah yang nantinya akan digunakan sebagai populasi awal pada

generasi berikutnya. Langkah awal pada proses seleksi ini adalah mencari *fitness*. Setiap individu yang telah dihitung nilai *fitness*-nya, nilai *fitness* yang akan digunakan untuk tahap seleksi berikutnya.

D. Crossover

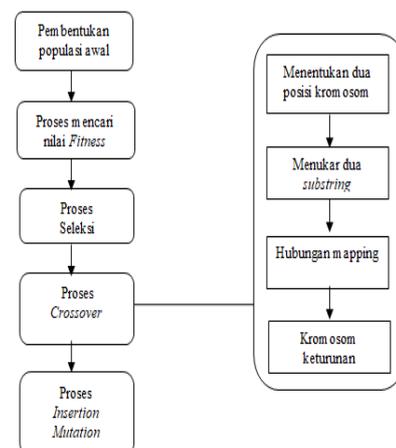
Proses *crossover* dilakukan dengan memecah kromosom menjadi dua tipe. Tipe pertama yaitu gen berupa *course object* dengan SKS berjumlah 2 dan jenis lainnya merupakan gen berupa *course object* dengan 3 SKS. Setelah memisahkan dua jenis gen tersebut, maka proses *crossover* yang akan dilakukan adalah dengan menukarkan posisi gen yang berjenis sama dengan merandom baris-baris gen tersebut. Selanjutnya, hasil dari *crossover* menghasilkan individu baru tanpa melanggar batasan yang telah dibuat.

E. Mutasi

Mutasi otomatis dilakukan pada saat proses pembentukan kromosom induk sebelum dilakukan proses *crossover*. Jenis mutasi yang digunakan pada penelitian ini adalah yaitu *Insertion Mutation*. *Insertion Mutation* ini memilih dua bilangan secara acak dalam satu kromosom, kemudian menukarkan nilai yang terdapat pada kedua bilangan acak tersebut. Banyaknya kromosom yang mengalami mutasi dalam satu populasi ditentukan dengan parameter *mutation rate* (P_m). Proses ini dilakukan dengan cara mengganti satu nilai gen yang terpilih secara *random* dengan suatu nilai baru yang didapat juga secara acak. Prosesnya didapat dari rumus :

Total Gen = Total gen dalam satu kromosom * total populasi

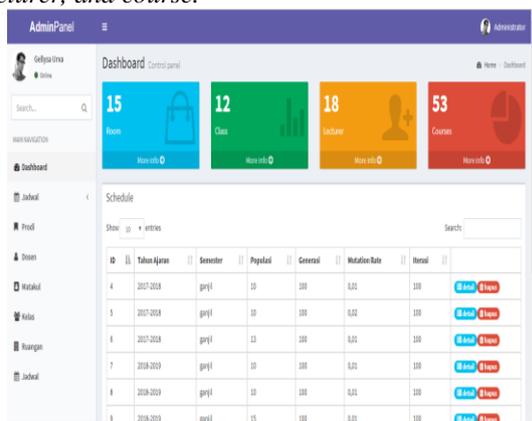
Untuk menentukan posisi gen yang dimutasi dilakukan dengan cara memberikan nilai bilangan bulat (*integer*) secara acak antara 1 sampai total gen. Jika bilangan acak yang dibangkitkan lebih kecil dari nilai *mutation rate* (P_m) maka dipilih posisi tersebut sebagai sub kromosom yang mengalami mutasi.



Gbr 3. Alur Proses *Insertion Mutation*

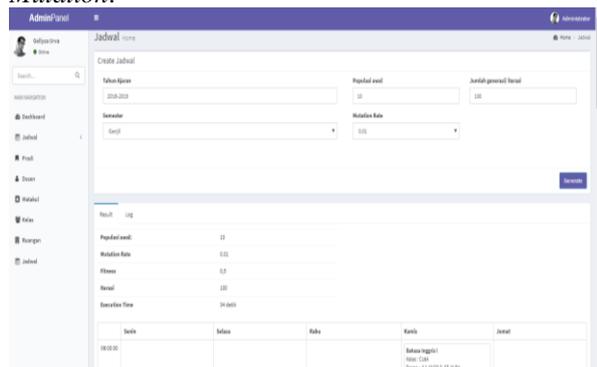
III. HASIL DAN PEMBAHASAN

Implementasi penjadwalan dosen mata kuliah menggunakan algoritma genetika ini berubah-ubah sesuai periode waktu penjadwalan dan matakuliah yang dibuka ataupun digunakan. Sehingga, aplikasi harus dapat dijalankan dengan data *input* matakuliah yang berbeda-beda sesuai dengan kebutuhan. Berikut ini, di tampilkan data penjadwalan matakuliah untuk tahun ajaran semester ganjil tahun 2018-2019 pada program studi teknik informatika di kampus STT Dumai. Gambar 4 berikut ini menunjukkan tampilan beranda dari aplikasi penjadwalan dosen. Pada gambar terlihat beberapa menu terdiri dari *room*, *class*, *lecturer*, and *course*.



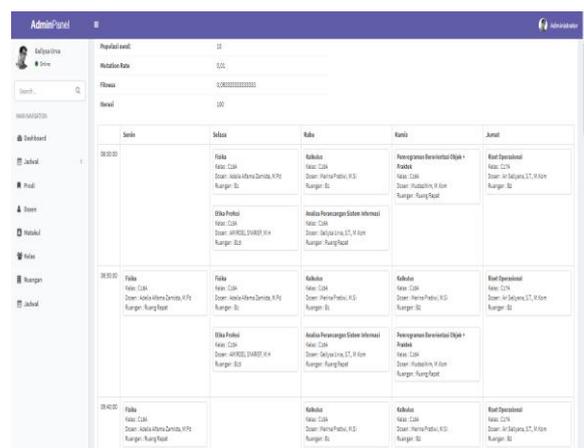
Gbr 4. Tampilan Beranda Penjadwalan

Selanjutnya pada menu jadwal tersedia hari dan waktu perkuliahan yang tersedia di kampus sesuai dengan ketentuannya. Perkuliahan untuk kelas pagi di jadwalkan dari hari senin hingga sabtu mulai pukul 08.00 sampai 13.00. Sedangkan untuk perkuliahan kelas sore dijadwalkan setiap hari senin sampai jum'at dengan waktunya dari jam 16.30 sampai 21.30. Selanjutnya pada menu dosen terdiri dari data dosen-dosen yang akan mengajar matakuliah yang sudah ditentukan oleh program studi. Kemudian terdapat menu mata kuliah yang disesuaikan dengan kebutuhan semester yang akan dilaksanakan serta banyaknya kelas mata kuliah dan ruangan yang disediakan untuk pelaksanaan perkuliahan. . Gambar 5 berikut ini menunjukkan proses penjadwalan dengan *Insertion Mutation*.



Gbr 5. Proses Penjadwalan dengan *Insertion Mutation*

Pada proses persiapan jadwal menggunakan algoritma genetika *crossover* PMX, terlebih dahulu kita menentukan tahun ajaran, semester, jumlah populasi awal, jumlah generasi serta *mutation rate*. Lalu kita pilih *generate* untuk prosesnya. Variasi populasi yang diambil untuk penelitian ini yaitu jumlah populasi 10, 30, 60, 90 dan 100 sedangkan variasi nilai *mutation rate* yang digunakan 0,01 0,03 dan 0,05. Besarnya populasi dan nilai *mutation rate* menghasilkan penjadwalan yang berbeda-beda. Pada penelitian ini yang menjadi variabel pengukuran kinerja mutasi *insertion mutation* pada proses pengolahan penjadwalan dosen pengampu mata kuliah yaitu nilai *fitness*, jumlah iterasi dan waktu proses penjadwalan mata kuliah. Gambar 6 menunjukkan contoh hasil proses penjadwalan mata kuliah.



Gbr 5. Hasil Penjadwalan dengan *Insertion Mutation*

Penjadwalan mata kuliah dengan jumlah populasi 10 dan *mutation rate* 0.01 menghasilkan nilai *fitness* sebesar 0,083 dengan jumlah iterasi 100 serta durasi waktu proses 1 menit 2 detik. Untuk nilai *mutation rate* 0.03 dengan jumlah populasi masih 10 menghasilkan nilai *fitness* sebesar 1 dengan jumlah iterasi 97 dan waktu proses 1 menit 3 detik. Sedangkan untuk nilai *mutation rate* 0.05 menghasilkan nilai *fitness* 1 dengan jumlah iterasi 73 dan waktu komputasi 12 detik. Tabel I merupakan tabel hasil penjadwalan dengan populasi 10 dan variasi *mutation rate* 0,01 0,03 dan 0,05.

TABEL I
PENJADWALAN DENGAN JUMLAH POPULASI 10

<i>Mutation Rate</i>	Jumlah Iterasi	Nilai <i>Fitness</i>	Waktu
0,01	100	0,83	1 menit 2 detik
0,03	100	1	1 menit 3 detik
0,05	73	1	12 detik

Selanjutnya, peneliti menggunakan jumlah populasi 30 untuk *mutation rate* 0,01 0,03 dan 0,05

menghasilkan nilai *fitness* yang sama yaitu 1. Waktu komputasi dari variasi *mutation rate* yang ditentukan 1 menit untuk *mutation rate* 0,03 dan 0,05. Akan tetapi untuk *mutation rate* 0,01 waktu komputasi selama 1 menit 4 detik. Selanjutnya untuk jumlah iterasi yang terjadi tetap berbeda-beda untuk setiap nilai *mutation rate*. Tabel II merupakan hasil penjadwalan dosen pengampu mata kuliah dengan jumlah populasi 30.

TABEL II
PENJADWALAN DENGAN JUMLAH POPULASI 30

Mutation Rate	Jumlah Iterasi	Nilai Fitness	Waktu
0,01	60	1	1 menit 4 detik
0,03	85	1	1 menit
0,05	66	1	1 menit

Uji coba selanjutnya yaitu penjadwalan dengan jumlah populasi sebanyak 60. Pada tabel III menunjukkan hasil proses penjadwalan dosen pengampu mata kuliah dengan jumlah populasi sebanyak 60. Nilai *fitness* yang dihasilkan dari variasi *mutation rate* menghasilkan nilai yang sama untuk ketiga variasi *mutation rate* yaitu 1. Durasi waktu komputasi proses penjadwalan selama 1 menit untuk *mutation rate* 0,01 sedangkan untuk *mutation rate* 0,03 dan 0,05 selama 2 menit. Banyaknya iterasi pada populasi 60 menghasilkan iterasi yang tidak terlalu jauh berbeda selisih rentang nilainya yaitu 58, 59 dan 62.

TABEL III
PENJADWALAN DENGAN JUMLAH POPULASI 60

Mutation Rate	Jumlah Iterasi	Nilai Fitness	Waktu
0,01	58	1	1 menit
0,03	59	1	2 menit
0,05	62	1	2 menit

Hasil penjadwalan dosen pengampu mata kuliah untuk jumlah populasi 90 menghasilkan nilai *fitness* yang sama dari penggunaan 3 variasi *mutation rate*. Untuk waktu komputasi selama 3 menit untuk nilai *mutation rate* bernilai 0,03 dan 0,05 sedangkan untuk nilai *mutation rate* 0,01 waktu komputasi selama 2 menit. Selisih waktu 1 menit selama proses komputasi. Jumlah iterasi yang dihasilkan 56, 62 dan 55. Tabel IV berikut ini menunjukkan hasil penjadwalan dosen pengampu mata kuliah dengan PMX menggunakan jumlah populasi 90.

TABEL IV
PENJADWALAN DENGAN JUMLAH POPULASI 90

Mutation Rate	Jumlah Iterasi	Nilai Fitness	Waktu
0,01	56	1	2 menit

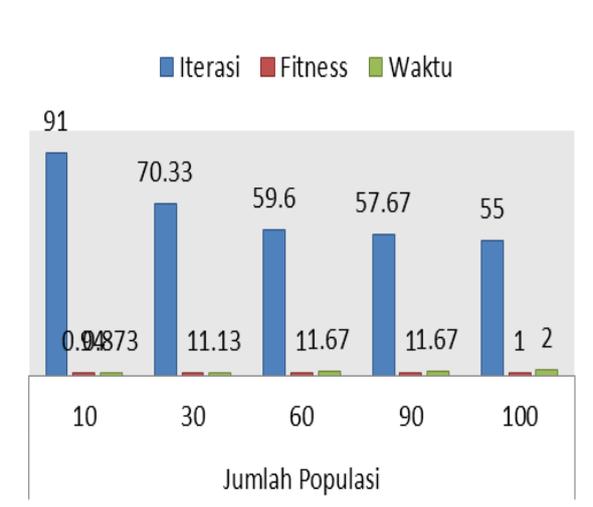
0,03	62	1	3 menit
0,05	55	1	3 menit

Selanjutnya tabel V menunjukkan hasil proses penjadwalan dengan jumlah populasi sebanyak 100. Dapat dilihat untuk tiga jenis *mutation rate* memiliki hasil iterasi dan waktu komputasi yang berbeda-beda. Sedangkan untuk nilai *fitness* masih bernilai 1. Waktu komputasi yang terlama pada saat nilai *mutation rate* 0,03 dengan jumlah iterasi 59, untuk nilai *mutation rate* 0,01 menghasilkan iterasi 50 dengan waktu 2 menit, serta 56 iterasi selama 1 menit untuk nilai *mutation rate* 0,05.

TABEL V
PENJADWALAN DENGAN JUMLAH POPULASI 100

Mutation Rate	Jumlah Iterasi	Nilai Fitness	Waktu
0,01	50	1	2 menit
0,03	59	1	3 menit
0,05	56	1	1 menit

Berdasarkan uji coba data untuk proses penjadwalan dosen pengampu mata kuliah dapat dilihat hasil proses penjadwalan dosen pengampu mata kuliah dengan *Insertion Mutation* menghasilkan penjadwalan yang bervariasi. Banyaknya jumlah populasi juga mempengaruhi jumlah iterasi selama proses penjadwalan dosen pengampu mata kuliah. Semakin besar populasi yang ditentukan semakin kecil iterasi yang dihasilkan. Untuk nilai *fitness* secara keseluruhan memiliki nilai mendekati 1. Hal ini menunjukkan penjadwalan tidak banyak melakukan pelanggaran. Sedangkan untuk durasi waktu komputasi proses penjadwalan dosen pengampu mata kuliah dengan *Insertion Mutation* ini secara maksimal hanya 2 menit. Grafik hasil proses penjadwalan mata kuliah dengan dapat dilihat pada gambar 6.



Gbr 6. Grafik Hasil Penjadwalan

IV. KESIMPULAN

Kesimpulan yang dapat diambil dari penelitian ini yaitu Penerapan algoritma genetika dengan *Insertion Mutation* pada penjadwalan mata kuliah mampu mendistribusikan penjadwalan mata kuliah terhadap dosen secara merata sesuai dengan aturan-aturan yang telah ditetapkan sehingga memudahkan program studi dalam membuat jadwal. Penjadwalan mata kuliah ini mampu menghindari terjadinya tabrakan antara mata kuliah, dosen ruang maupun waktu. Sehingga permasalahan yang seringnya terjadi pada pembuatan jadwal sudah dapat diminimalisir. Proses penjadwalan mata kuliah dengan *Insertion Mutation* dinyatakan lebih efisien dan optimal dibandingkan proses penjadwalan sebelumnya. Hal ini terlihat dari durasi waktu proses penjadwalan mata kuliah hanya berlangsung maksimal selama 2 menit.

REFERENSI

- [1] S. Nguyen, Y. Mei, H. Ma, A. Chen, and M. Zhang, "Evolutionary Scheduling and Combinatorial Optimisation: Applications, Challenges, and Future Directions."
- [2] E. K. Burke *et al.*, "Özcan, Ender and Qu, Rong (2013) Hyper-heuristics: a survey of the state of the art. Journal of the Operational Hyper-heuristics: a survey of the state of the art," pp. 1695–1724, 2013.
- [3] D. S. Widodo, P. B. Santoso, and E. Siswanto, "PENDEKATAN ALGORITMA CROSS ENTROPY - GENETIC ALGORITHM UNTUK MENURUNKAN MAKESPAN PADA PENJADWALAN FLOW SHOP," vol. 2, no. 1, pp. 41–49, 2014.
- [4] X. Zuo, S. Member, C. Chen, W. Tan, and S. Member, "Vehicle Scheduling of an Urban Bus Line via an Improved Multiobjective Genetic Algorithm," vol. 16, no. 2, pp. 1030–1041, 2015.
- [5] A. Ansari and P. S. Bojewar, "Genetic Algorithm to Generate the Automatic Time-Table – An Over View," 2014.
- [6] N. Z. Janah and I. Brayudi, "SISTEM PENDUKUNG KEPUTUSAN PEMBAGIAN BEBAN MENGAJAR DOSEN DENGAN METODE ALGORITMA GENETIKA STUDI KASUS: POLITEKNIK NEGERI BATAM," pp. 6–7, 2016.
- [7] M. Kühn, T. Severin, and H. Salzwedel, "Variable Mutation Rate at Genetic Algorithms: Introduction of Chromosome Fitness in Connection with Multi-Chromosome Representation," *Int. J. Comput. Appl.*, vol. 72, no. 17, pp. 31–38, 2013.
- [8] M. Abbaszadeh and S. Saeedvand, "A Fast Genetic Algorithm for Solving University Scheduling Problem," vol. 3, no. 1, pp. 7–15, 2014.
- [9] N. H. Moin, O. C. Sin, and M. Omar, "Hybrid Genetic Algorithm with Multiparents Crossover for Job Shop Scheduling Problems," vol. 2015, 2015.
- [10] R. Arifudin, "Riza Arifudin," vol. 2, pp. 1–14.
- [11] S. K. Jha, "Exam Timetabling Problem using Genetic Algorithm," *Int. J. Res. Eng. Technol.*, vol. 3, no. 4, pp. 649–654, 2014.
- [12] V. A. Leena, A. B. A. S., and M. S. Rajasree, "Genetic Algorithm Based Bi-Objective Task Scheduling in Hybrid Cloud Platform," vol. 8, no. 1, 2016.