

Mitigasi Premature Convergence Pada Genetic Algorithm Menggunakan Metoda Dynamics Growth Population Dalam Kasus University Course Scheduling

Ewi Ismaredah*¹, Hasdi Radiles²

^{1,2} Universitas Islam Negeri Sultan Syarif Kasim

e-mail: *ismaredah@uin-suska.ac.id, hasdi@uin-suska.ac.id

Abstract – The problem of scheduling lecture activities or what is commonly referred to as University Course Scheduling (UCS), is still a dilemma between the interests of lecturers, students and available facilities. One solution to these problems is to use Genetic Algorithm (GA) to describe the permutation of lecture events with consideration of the desired constraints. This study proposes the use of Dynamics Population for population growth per generation to prevent premature convergence due to limited search space. The research data was obtained based on the scheduling process in the Department of Electrical Engineering at UIN SUSKA - Riau 2019-2020 odd semester and the results of interviews from a number of academics. Several scenarios observed in this study are based on the variation of the initialization population of 50-100 individuals, with a probability of 0.1 to 0.5 and a mutation probability of 0.01 to 0.05. The results showed that initializing population 70 with a crossover probability of 0.5 could solve the problem of premature convergence to get a solution to UCS. In addition, the addition of the mutation probability of more than 0.01 will result in a higher computational burden.

Abstrak – Permasalahan penjadwalan kegiatan perkuliahan atau yang biasa disebut sebagai University Course Scheduling (UCS), hingga saat ini masih menjadi dilema antara kepentingan dosen, mahasiswa dan fasilitas yang tersedia. salah satu solusi terhadap permasalahan permasalahan tersebut ini adalah dengan menggunakan Genetic Algorithm (GA) untuk menguraikan permutasi acara perkuliahan dengan pertimbangan constraint yang diinginkan. Penelitian ini mengusulkan penggunaan Dynamics Population pada pertumbuhan jumlah populasi setiap generasinya untuk mencegah terjadi premature convergence akibat terbatasnya search space. Data penelitian diperoleh berdasarkan proses penjadwalan pada jurusan Teknik Elektro UIN SUSKA –Riau semester Gasal 2019-2020 dan hasil interview dari sejumlah civitas akademika. Beberapa skenario yang diamati dalam penelitian ini adalah berdasarkan variasi inisialisasi populasi 50-100 individu, dengan probabilitas 0,1 hingga 0,5 dan probabilitas mutasi 0,01 hingga 0,05. Hasil penelitian menunjukkan bahwa inisialisasi populasi 70 dengan probabilitas crossover 0,5 dapat mengatasi permasalahan premature convergence untuk mendapatkan solusi terhadap UCS. Selain itu penambahan probabilitas mutasi lebih dari 0,01 akan mengakibatkan beban komputasi yang semakin tinggi.

Kata Kunci – Dynamics Population, Elithsm, Genetic Algorithm, PMX, Reciprocal Exchange Mutation.

I. PENDAHULUAN

Jadwal perkuliahan menjadi hal yang sangat sensitif karena berkaitan dengan pembagian waktu serta efisiensi penggunaan ruang kelas yang tersedia. Penjadwalan kerab dilakukan pihak program studi dengan matakuliah yang memiliki jumlah sks yang beragam. Tidak hanya itu dosen pengampu mata kuliah juga menjadi indikator penting dalam penyusunan jadwal mata kuliah karena dosen dapat mengampu lebih dari satu mata kuliah. Dengan demikian penyusunan jadwal menjadi sorotan terpenting dalam perkuliahan.

Namun fakta dilapangan penyusunan jadwal perkuliahan tidaklah mudah. Sering terjadi bentrok antar mata kuliah yang mengakibatkan mahasiswa menjadi dilema mata kuliah mana yang akan di prioritaskan. Melakukan revisi jadwal juga tidak efisien menimbang perkuliahan sudah berjalan serta timbul keranjuan apabila informasi tidak diterima secara baik.

Pengamatan telah dilakukan dalam beberapa semester di salah fakultas yang ada di UIN SUSKA Riau, selama proses penyusunan jadwal matakuliah sering mengalami peninjauan ulang dari jadwal yang diumumkan sehingga jadwal yang pasti dapat digunakan setelah melakukan penyusunan ulang selama dua hingga empat minggu. Hal ini berdampak pada mahasiswa karena terjadi jadwal yang bentrok, bentrok tersebut berdampak kepada persentase kehadiran mahasiswa yang harus memilih salah satu matakuliah untuk dihadiri.

Berdasarkan masalah tersebut diperlukan sebuah sistem yang dapat menyusun jadwal secara baik. Penelitian luar negeri mengungkapkan beberapa algoritma telah digunakan dalam permasalahan penyusunan jadwal perkuliahan, seperti Tabu Search, Simulated Anneling dan Genetic Algorithm. Namun sebagian besar penelitian fokus pada permasalahan komputasi karena permasalahan penjadwalan ini memiliki sampel permutasi yang sangat besar[1].

Permasalahan penyusunan penjadwalan ini telah dilakukan sebelumnya oleh peneliti internasional, dalam penelitiannya mereka menggunakan Genetic Algorithm, namun penelitian mereka terfokus pada penggunaan Hard Constraint dan soft Constraint dalam merepresentasikan fakta di lapangan[1], [2].

Peneliti dalam negeri juga membahas permasalahan penjadwalan perkuliahan dengan menggunakan teknik seleksi tournament. Namun penelitian ini terfokus kepada komputasi yaitu proses pembuatan atau penyusunan jadwal untuk perkuliahan, yaitu hanya diperlukan waktu sekitar 14,7 menit dibandingkan dengan proses manual yang memerlukan waktu sekitar 2 (dua) hari[3].

Namun Genetic Algorithm standar sendiri memiliki kekurangan, yaitu premature convergence. Dalam penelitian lainnya penggunaan pengembangan metode Genetic Algorithm yang disebut PRoFIGA (Population Resizing on Fitness Improvement Genetic Algorithm) dimana algoritma ini digunakan untuk mengatasi masalah premature convergence. Premature convergence terjadi akibat dari kurang bervariasinya individu didalam populasi[4]. Secara sederhana homogenitas individu menghasilkan ruang pencarian menjadi tidak optimum.

Berdasarkan permasalahan yang telah dipaparkan diatas, tujuan dari penelitian ini yaitu melanjutkan penelitian sebelumnya yang dilakukan di lingkungan UIN SUSKA Riau yaitu University Course Scheduling (UCS)[17], kami mengusulkan penggunaan Dynamics Growth Population Genetic algorithm untuk pencegahan premature convergence. Penggunaan metoda dynamics population ini didasari pada penelitian sebelumnya yang menggunakan metoda PRoFIGA[4] dan APOGA[5] yang menunjukkan dapat menyelesaikan masalah premature convergence.

II. PENELITIAN YANG TERKAIT

Dalam penelitian sebelumnya, permasalahan ruang pencarian yang kurang optimum dapat diperbaiki menggunakan pengembangan dari Genetic Algorithm yang disebut APOGA (Adaptive Population Pool Size Based Genetic Algorithm), Hasil percobaan menunjukkan bahwa APOGA mengungguli GA standar untuk semua ruang solusi[5].

Genetic Algorithm sendiri adalah teknik pencarian stokastik[10][11][12], algoritma ini terinspirasi dari teori evolusi dan seleksi alam Darwin. Menurut teori ini, semakin fit suatu organisme, semakin ia dapat bertahan di alam dan sebaliknya (yaitu survival of the fittest).GA diterapkan dalam banyak masalah pengoptimalan dunia nyata untuk menentukan solusi optimal atau mendekati optimal [13] [14] [15] [16]. Dalam GA, ruang pencarian dibentuk dari individu dan masing-masing individu mewakili solusi yang mungkin untuk masalah tertentu. Individu ini dikodekan sebagai kromosom. Operator genetik utama termasuk Crossover dan mutasi kemudian diterapkan secara probabilistik ke kromosom yang dipilih dengan metode seleksi dengan tujuan menciptakan solusi baru dan berpotensi lebih baik[6].

Penelitian terkait lainnya mengemukakan bahwa penggunaan representasi individu berupa permutasi akan bersifat unik, dikarenakan tidak adanya duplikat disetiap kromosom. Untuk itu metode Crossover yang tepat digunakan adalah metode PMX (Partially mapped Crossover). Hasil dari penelitian ini secara signifikan meningkatkan efisiensi Genetic Algorithm berbasis permutasi dan juga membantu dalam menghasilkan solusi yang berkualitas baik[6].

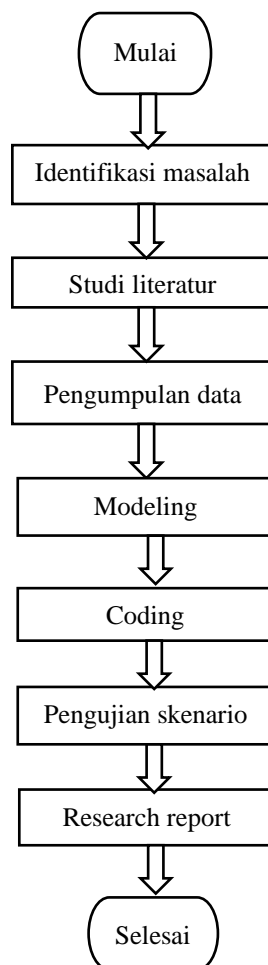
Representasi permutasi juga digunakan oleh peneliti dalam negeri, operator reproduksi yang digunakan untuk mereproduksi offspring yaitu one cut-point diimplementasikan pada Crossover dan reciprocal exchange mutation.

Dari hasil pengujian parameter terbaik, didapat individu terbaik memiliki nilai fitness 1 dengan tidak terdapat nilai pelanggaran [7].

Sebelumnya sudah pernah dilakukan penelitian dengan studi kasus yang sama dilingkungan UIN SUSKA Riau, penelitian tersebut mengemukakan butuh jumlah populasi hingga 1000 untuk mendapatkan variasi solusi sebelum premature convergence terjadi [17].

III. METODE PENELITIAN

2.1 Alur Penelitian



Gambar 1. Aluran penelitian

2.2 Pengumpulan Data

Program Studi Teknik Elektro Universitas Islam Negeri Sultan Syarif Kasim menjadi lokasi survei, tahap pertama dalam survei ini adalah mengumpulkan kurikulum yang digunakan sebagai data utama.

2.3 Data Perkuliahan

Dari hasil pertimbangan, data perkuliahan yang digunakan dalam penelitian ini adalah data perkuliahan Program Studi Teknik Elektro pada semester ganjil TA 2019-2020. Matakuliah praktikum diabaikan dengan tujuan untuk penyederhanaan model dengan pertimbangan matakuliah ini menggunakan laboratorium dan bersifat fleksibel dari segi waktu pelaksanaan, dengan demikian total pelaksanaan perkuliahan menjadi 101 kejadian perkuliahan dalam satu minggu dan dalam pelaksanaannya telah disediakan 7 ruang kelas. Berikut ini merupakan daftar matakuliah dan dosen yang akan dijadikan model dalam pengkodean program.

Tabel 1: Sampel Data Matakuliah

Nama Matakuliah	Kode
Kalkulus 1	PTE1101
Mekanika, Gelombang Dan Optika	PTE1102
Algoritma Dan Pemrograman	PTE1103
Komunikasi Profesional	PTE1105
Pancasila	UIN2001
Aqidah Akhlak	UIN2006
Bahasa Indonesia	UIN2008
Studi Alquran	UIN2004
Rangkaian Listrik	PTE1312
Sinyal Dan Sistem	PTE1314
Sistem Digital	PTE1315
Elektronika	PTE1316
Probabilitas Dan Statistika Terapan	PTE1317

2.4 Model Constraint

Constraint pada penelitian ini telah disederhanakan menjadi dua jenis, yaitu:

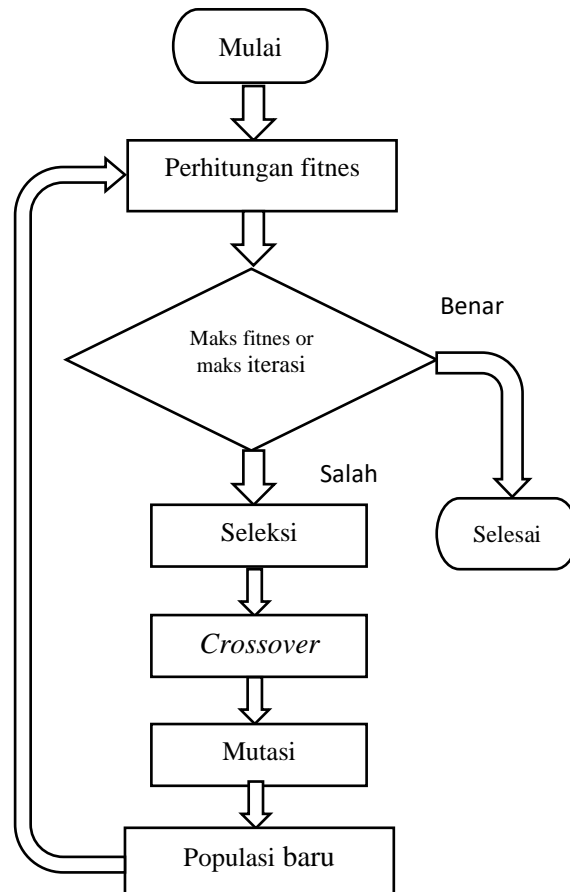
1. *Hard Constraint*
 - a. Seorang dosen tidak dapat menghadiri dua perkuliahan di waktu yang sama.
 - b. matakuliah dengan semester dan kelas tertentu tidak dapat terjadi di waktu yang bersamaan[17].
2. *Soft Constraint*
 - a. Matakuliah semester 1 dan 3 menempati slot waktu yang sama
 - b. Matakuliah semester 5 dan 7 menempati slot waktu yang sama[17].

2.5 Model Kegiatan Perkuliahan

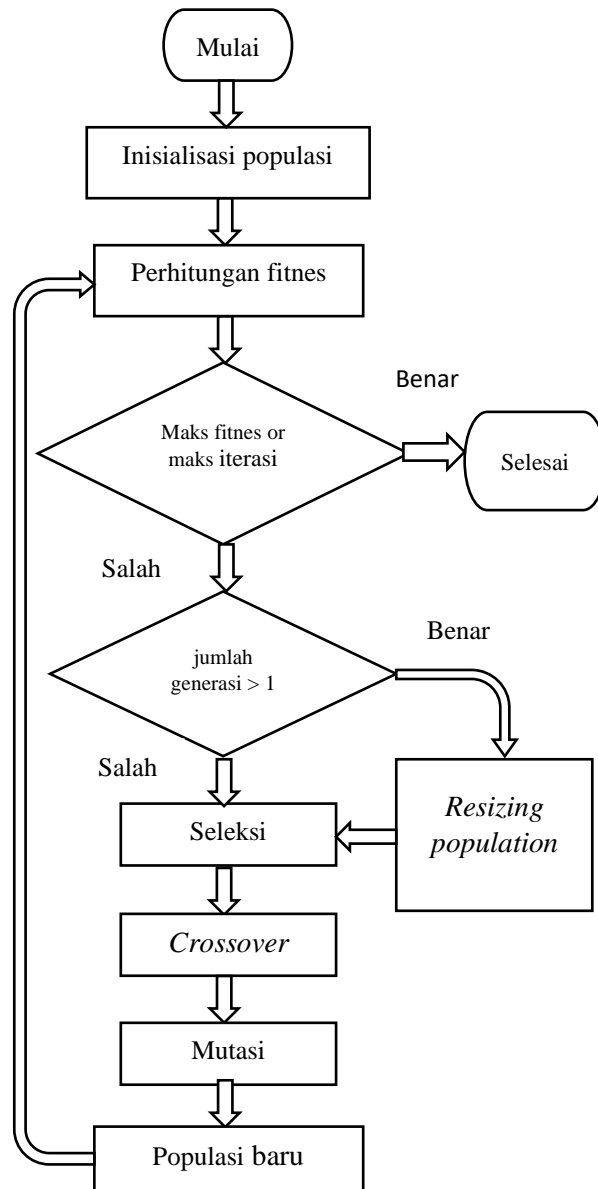
Rancangan penjadwalan ditujukan untuk mempercepat penyelesaian kurikulum dan juga memberikan kesempatan bagi mereka yang ingin memperbaiki hasil matakuliah sebelumnya. Pada semester ganjil, kelas yang disediakan pada semester 1 dan 3, masing-masing secara normal terdiri dari 4 kelas, yaitu kelas A, B, C dan D. Sedangkan semester 5 dan 7 kelas ditentukan berdasarkan konsentrasi yang tersedia pada Program Studi Teknik Elektro Uin Suska Riau.

2.6 Flow chart

Dinamisasi populasi Genetic Algorithm merupakan pengembangan dari Genetic Algorithm standar. Dimana metode ini mengubah ukuran populasi berdasarkan perbandingan kondisi fitness generasi sekarang dengan kondisi fitness generasi sebelumnya. Dengan kata lain populasi setiap generasi memiliki kemungkinan tumbuh atau menyusut. Hal ini demi menjaga tingkat keragaman populasi untuk mencegah terjadinya *premature convergence* pada saat proses pencarian solusi optimal.



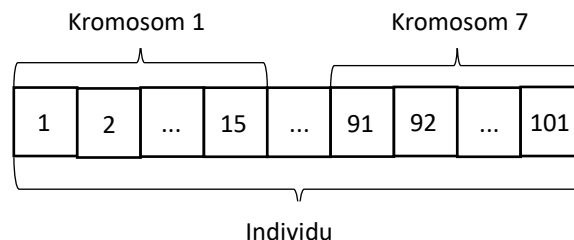
Gambar 2. Flow chart standar GA



Gambar 3. Flow chart DPGA

2.7 Representasi individu

Penelitian ini menggunakan teknik pengkodean permutasi untuk merepresentasi gen-gen dalam individu[6][7]. Teknik ini menggunakan bilangan integer untuk mewakili setiap kejadian perkuliahan dalam satu minggu[8]. Individu pada penelitian ini terdiri dari 101 gen, contoh bentuk individu ditunjukkan pada gambar 3.3 dibawah ini.



Gambar 4. Representasi Individu

Struktur masing-masing individu memiliki 7 kromosom, dimana kromosom merepresentasikan jumlah kelas yang tersedia. Kemudian jumlah gen dalam masing-masing kromosom merepresentasikan jumlah slot waktu dalam satu minggu. artinya terdapat maksimal 15 kejadian perkuliahan dalam satu minggu di masing-masing kelas, untuk penentuan waktu pada masing-masing slot waktu tergantung dari beban sks yang dimiliki setiap matakuliah. Jika

direpresentasikan dalam jadwal perkuliahan akan tampak seperti gambar dibawah ini.

Tabel 2: Representasi Individu dalam Bentuk Jadwal Perkuliahan

Waktu/ Ruang	1	2	3	4	5	6	7
Senin, waktu 1	1	16	31	46	61	76	91
...
kamis, waktu 2	1 1	26	41	56	71	86	101
kamis, waktu 3	1 2	27	42	57	72	87	
jumat, waktu 1	1 3	28	43	58	73	88	
jumat, waktu 2	1 4	29	44	59	74	89	
jumat, waktu 3	1 5	30	45	60	75	90	

2.8 Inisialisasi Populasi

Inisialisasi populasi merupakan proses awal dari Genetic Algorithm. Pada penelitian ini proses inisialisasi populasi dilakukan secara acak. Populasi yang dibangkitkan ini merupakan kumpulan dari beberapa individu dengan jumlah tertentu, populasi yang dibangkitkan memiliki bentuk matriks. Setiap individu memiliki keunikan pada gen mereka masing-masing, tidak ada individu yang memiliki bentuk gen yang sama.

2.9 Perhitungan Fitness

Perhitungan fitness pada masing-masing individu dilakukan berdasarkan dua peraturan, yaitu *hard constraint* dan *soft constraint*. *Hard constraint* merupakan peraturan yang nilai penaltinya paling tinggi sehingga apabila dilanggar maka nilai fitness individu akan turun drastis. Kemudian *soft constraint* tidak seketat *hard constraint* maka dari itu setiap pelanggaran *soft constraint* akan mengurangi nilai fitness 1/100 dari hard constraint, dengan kata lain dari 100 pelanggaran *soft constraint* setara dengan 1 pelanggaran *hard constraint*. Adapun perhitungan fitness untuk setiap individu sebagai berikut.

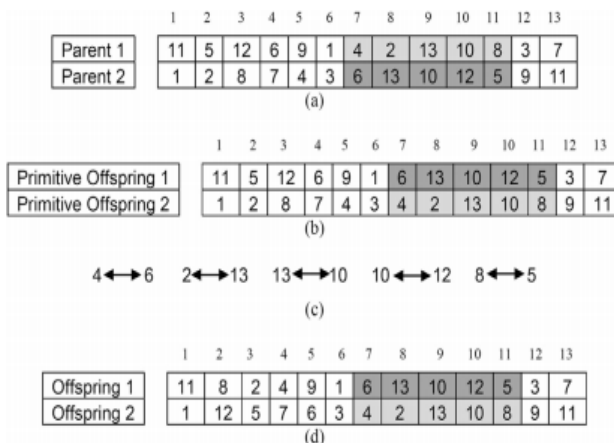
$$F = \frac{1}{1+100 \times \sum HC + \sum SC} \quad (1)$$

2.10 Selection

Penelitian ini menggunakan metode seleksi *elitism*, dimana hanya beberapa individu dengan fitness terbaik yang akan dipilih[8]. Tujuan dari metode seleksi ini adalah memberikan kesempatan yang besar kepada individu terbaik untuk melahirkan *offspring* dengan kualitas fitness yang baik juga dan menghindari kehilangan individu terbaik[9].

2.11 Crossover

Penelitian ini menggunakan metode *Crossover Partially Mapped Crossover* (PMX). Pemilihan metode didasari oleh bentuk representasi kromosom permutasi[6], terdapat parameter yang pentingnya, yaitu *crossover rate* atau probabilitas perkawinan silang, parameter ini didefinisikan untuk mengatur laju tingkat terjadinya perkawinan silang. Adapun gambaran proses metode *Partially Mapped Crossover* dapat dilihat dibawah ini.



Gambar 5. Proses *Crossover* PMX[6]

Pertama pilih secara acak dua titik potong pada dua masing-masing *parent*, ciptakan dua *primitif offspring* dengan menukarkan gen didalam area titik potong, setelah itu definisikan daftar pertukaran gen. Kemudian tentukan jalur pertukaran antar gen selain gen yang berada pada area titik potong. Tukarkan gen yang sudah ditentukan jalurnya sebelumnya. Hasil dari pertukaran tersebut melahirkan *offspring1* dan *offspring2*[6].

2.12 Mutasi

Proses mutasi merupakan salah satu proses reproduksi dalam Genetic Algorithm. Proses ini akan mengubah sebagian kecil struktur gen setiap anak yang lahir dari hasil proses *Crossover*. setiap anak akan mengalami beberapa kali proses mutasi bergantung pada parameter *mutation rate* atau probabilitas mutasi. Proses mutasi sendiri memiliki sejumlah metode, penelitian ini menggunakan metode *reciprocal exchange mutation*, yaitu dengan memilih dua titik pertukaran secara random kemudian menukarkan posisi gen yang terpilih tersebut.

2.13 *Resizing population*

Dinamisasi ukuran populasi ini dilakukan berdasarkan keadaan fitness tertinggi generasi sekarang dengan keadaan fitness tertinggi generasi sebelumnya. Populasi akan meningkat apabila keadaan fitness generasi sekarang meningkat dari generasi sebelumnya, cara menentukan tingkat pertumbuhan populasi dapat menggunakan rumus berikut ini.

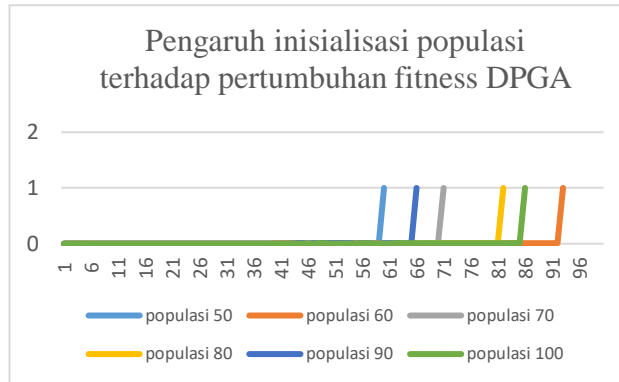
$$G = \alpha \times (I^{max} - I) \times \frac{F^{new} - F^{old}}{F^{max}} \tag{2}$$

Dimana G adalah ukuran pertumbuhan, α adalah konstanta antara 0 dan 1, I^{max} adalah jumlah maksimal iterasi, I adalah iterasi saat ini, F^{new} adalah nilai fitness tertinggi generasi sekarang, F^{old} adalah nilai fitness generasi sebelumnya, F^{max} adalah nilai fitness maksimum. Populasi juga akan mengalami pertumbuhan apabila nilai fitness saat ini tidak mengalami peningkatan sebelumnya selama *n* iterasi, dengan ukuran peningkatan sama dengan kondisi fitness sekarang lebih tinggi dari kondisi fitness sebelumnya[5]. Apabila nilai fitness saat ini menurun atau sama dengan nilai fitness sebelumnya maka ukuran populasi akan mengalami penyusutan, Ukuran penyusutan dianjurkan sebesar 5% dari jumlah populasi terbaru[4], penyusutan pada penelitian ini dilakukan secara random. Dengan demikian populasi pada setiap generasi akan mengalami pertumbuhan atau penyusutan dengan nilai yang bervariasi.

IV. HASIL DAN PEMBAHASAN

Genetic Algorithm telah berhasil diimplementasikan dan telah menjadi sebuah sistem yang dapat menyusun jadwal secara otomatis. Tolak ukur dari keberhasilan algoritma ini yaitu nilai fitness mendekati angka 1, semakin dekat nilai fitness dengan angka 1 maka hasil penjadwalan semakin baik dan tidak terjadi bentrok.

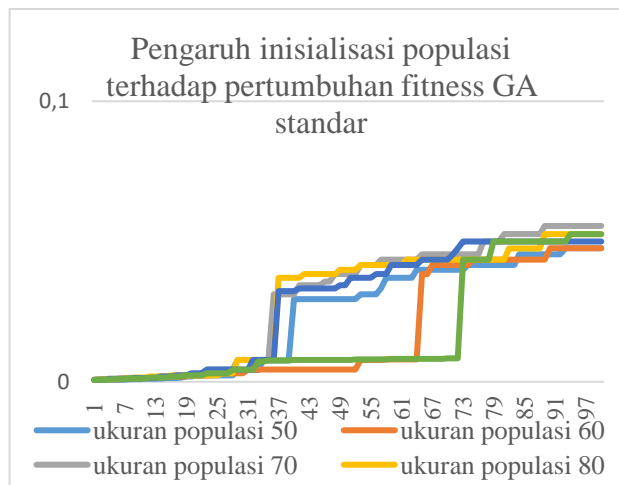
3.1 Pengaruh Jumlah Populasi Terhadap Konvergensi Generasi



Gambar 6. pengaruh populasi terhadap fitness pada DPGA

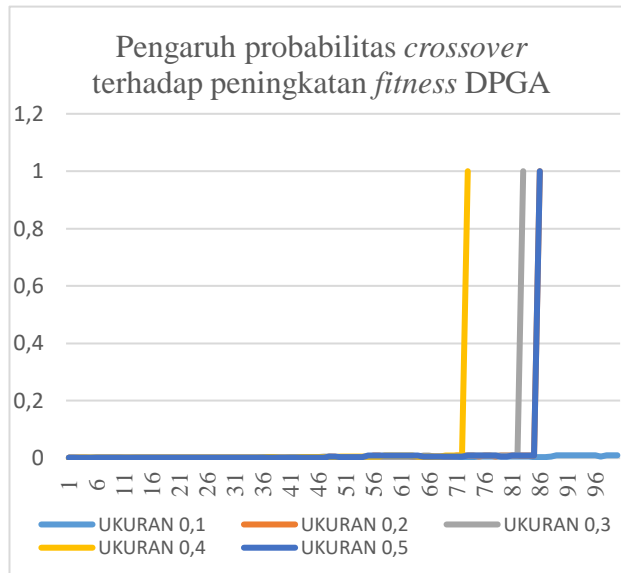
Jika diperhatikan grafik DPGA, pertumbuhan fitness mengalami fluktuatif, namun semakin tinggi generasi peningkatan fitness semakin bagus, dari keenam ukuran populasi semuanya mendapatkan solusi optimal. Hal tersebut menunjukkan ruang pencarian pada ukuran populasi tersebut bervariasi, ketika hasil pencarian bersifat homogen maka DPGA akan mencari referensi individu lain dalam ruang pencariannya untuk dijadikan *parent* dengan melakukan pengaturan ulang jumlah populasi, tetapi semakin tinggi generasi tingkat homogenitas individu semakin tinggi dan menyebabkan konvergen di atas generasi ke-50 sebelum mendapatkan solusi optimal.

Jika dibandingkan dengan GA standar, Pada Genetic Algorithm standar dari semua ukuran populasi, semuanya mengalami *premature convergence* ditandai dengan titik konstan cukup lama setiap generasi sebelum peningkatan fitness, hal ini diakibatkan setiap perpindahan generasi terjadi peningkatan homogenitas susunan individu dalam populasi sebelum mendapatkan fitness optimum. Hasil GA standar dapat dilihat pada gambar berikut.



Gambar 7. pengaruh populasi terhadap fitness pada GA standar

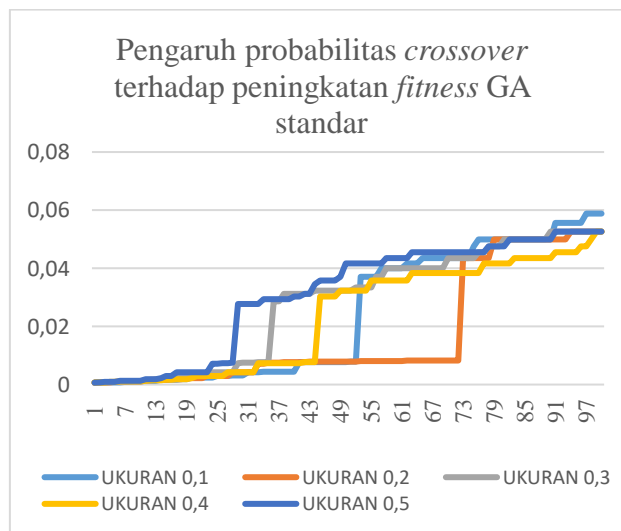
3.2 Pengaruh Probabilitas *Crossover* Terhadap Konvergensi Generasi



Gambar 8. pengaruh probabilitas *crossover* terhadap fitness pada DPGA

Berdasarkan grafik tersebut ukuran 0,1 merupakan ukuran yang tidak ideal untuk melakukan Proses *Crossover*. Hal ini dikarenakan kurangnya individu baru yang lahir bervariasi sehingga tidak dapat mengurangi homogenitas populasi, selain ukuran 0,1 menjadi ukuran yang cukup ideal untuk mengurangi *premature convergence*.

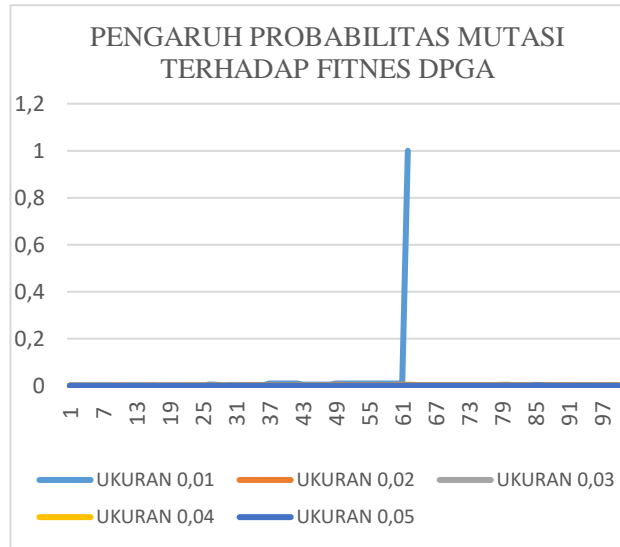
Kemudian skenario tersebut juga dijalankan pada GA standar, hasil dari pengujian dapat dilihat pada gambar berikut.



Gambar 9. pengaruh probabilitas *crossover* terhadap fitness pada GA standar

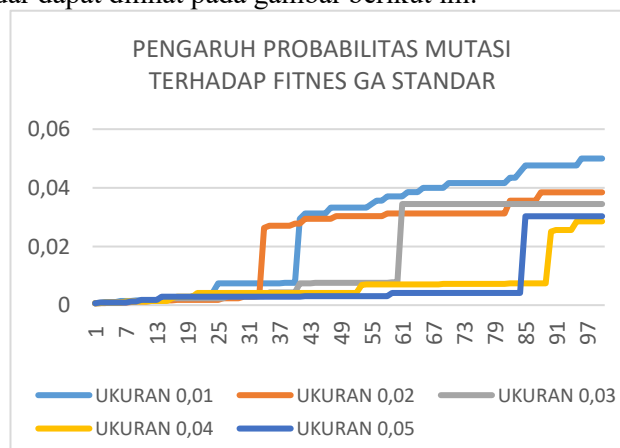
Memberikan variasi nilai pada probabilitas *crossover* tidak mampu membantu algoritma GA standar untuk mendapatkan solusi optimal, pencapaian tertinggi dari algoritma GA standar berada pada nilai 0.05 yaitu dengan ukuran 0.1

3.3 Pengaruh Probabilitas Mutasi Terhadap Konvergensi Generasi



Gambar 10. pengaruh probabilitas mutasi terhadap fitness pada DPGA

Hasil dari pengaruh mutasi terhadap konvergensi diperlihatkan oleh gambar diatas, ukuran 0,01 menjadi ukuran satu-satunya yang dapat menghasilkan solusi optimal, selain ukuran tersebut, *premature convergence* tidak dapat diatasi. Hasil pengujian pada GA standar dapat dilihat pada gambar berikut ini.



Gambar 11. pengaruh probabilitas mutasi terhadap fitness pada GA standar

Pada algoritma GA standar pencapaian terbaik berapa pada angka 0,05 dengan ukuran 0,01 untuk probabilitas mutasi. Dengan demikian variasi ukuran probabilitas mutasi belum mampu mengatasi *premature convergence* pada algoritma GA standar.

V. KESIMPULAN

Sistem penjadwalan telah selesai diimplementasikan dengan metode DPGA. Metoda ini dapat mengoptimalkan ruang pencarian sehingga dapat mencegah terjadinya *premature convergence*. Skenario yang diimplementasikan dalam menguji performa sistem meliputi ukuran populasi, ukuran probabilitas crossover dan probabilitas mutasi. Berdasarkan hasil yang telah dibahas dapat disimpulkan ukuran populasi berjumlah 60 dan ukuran probabilitas crossover 0.2 hingga 0.4 menjadi yang ideal dalam menjaga keragaman ruang pencarian, dan mempercepat mendapatkan solusi optimal. Sedangkan ukuran probabilitas mutasi ideal adalah 0.01, kemudian untuk ukuran yang lebih dari itu akan mengakibatkan beban komputasi bertambah berat.

UCAPAN TERIMA KASIH

Ucapan terima kasih penulis kepada pihak institusi Universitas Islam Negeri Sultan Syarif Kasim yang membantu ataupun memberikan dukungan terkait dengan penelitian yang dilakukan seperti bantuan fasilitas penelitian.

DAFTAR PUSTAKA

- [1] M. Yazdani, B. Naderi, and E. Zeinali, "Algoritmi za probleme planiranja fakultetskih predavanja," *Teh. Vjesn.*, vol. 24, pp. 241–247, 2017, doi: 10.17559/TV-20130918133247.
- [2] E. A. Abdelhalim and G. A. El Khayat, "A Utilization-based Genetic Algorithm for Solving the University Timetabling Problem (UGA)," *Alexandria Eng. J.*, vol. 55, no. 2, pp. 1395–1409, 2016, doi: 10.1016/j.aej.2016.02.017.
- [3] Y. Sari, M. Alkaff, E. S. Wijaya, S. Soraya, and D. P. Kartikasari, "Optimasi Penjadwalan Mata Kuliah Menggunakan Metode Genetic Algorithm dengan Teknik Tournament Selection," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 6, no. 1, p. 85, 2019, doi: 10.25126/jtiik.2019611262.
- [4] T. Listyorini and S. Muzid, "Implementasi Population Resizing on Fitness Improvement Genetic Algorithm (Profiga) Untuk Optimasi Rute Kunjungan Promosi Universitas Muria Kudus Berbasis Android Dan Google Maps Api," *Simetris J. Tek. Mesin, Elektro dan Ilmu Komput.*, vol. 7, no. 1, p. 59, 2016, doi: 10.24176/simet.v7i1.488.
- [5] B. R. Rajakumar and A. George, "APOGA: An Adaptive Population Pool Size based Genetic Algorithm," *AASRI Procedia*, vol. 4, pp. 288–296, 2013, doi: 10.1016/j.aasri.2013.10.043.
- [6] B. Koohestani, "A crossover operator for improving the efficiency of permutation-based genetic algorithms," *Expert Syst. Appl.*, vol. 151, p. 113381, 2020, doi: 10.1016/j.eswa.2020.113381.
- [7] R. Febriyana and W. F. Mahmudy, "Penjadwalan Kapal Penyeberangan Menggunakan Genetic Algorithm," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 3, no. 1, p. 43, 2016, doi: 10.25126/jtiik.201631169.
- [8] A. M. Purnomo, D. Werdiastu, T. Raissa, R. Widodo, and V. N. Wijayaningrum, "Genetic Algorithm untuk Optimasi Komposisi Makanan Bagi Penderita Hipertensi," *J. Teknol. dan Sist. Komput.*, vol. 7, no. 1, pp. 1–6, 2019, doi: 10.14710/jtsiskom.7.1.2019.1-6.
- [9] Syarif, Admi, *Genetic Algorithm Teori dan Aplikasi*, Edisi 2, Yogyakarta: Graha Ilmu, 2014.
- [10] Lei Wang and Dun-bing Tang. An improved adaptive genetic algorithm based on hormone modulation mechanism for job-shop scheduling problem. *Expert Systems with Applications*. 2011;38 7243-7250.
- [11] EzgiDenizÜlker and SadikÜlker. Comparison Study for Clonal Selection Algorithm and Genetic Algorithm. *International Journal of Computer Science & Information Technology*. 2012;4:4.
- [12] Abdolreza. AsadiGhanbari, Ali Broumandnia, HamidrezaNavidi and Ali.Ahmadi. Brain Computer Interface with Genetic Algorithm. *International Journal of Information and Communication Technology Research*. 2012; 2:1.
- [13] R. P. Abeysooriya and T. G. I Fernando. Canonical Genetic Algorithm To Optimize Cut Order Plan Solutions in Apparel Manufacturing. *Journal of Emerging Trends in Computing and Information Sciences*. 2012;3:2.
- [14] Vosough. Amir. Optimization the Rankine Cycle with Genetic Algorithm. 2nd International Conference on Mechanical, Production and Automobile Engineering, Singapore. 2012; 28-29.
- [15] AkhileshVerma and Archana. A Survey on Image Contrast Enhancement Using Genetic Algorithm. *International Journal of Scientific and Research Publications*. 2012; 2:7.
- [16] DawoodTalebiKhanmiri, NasibehNasiri, and TaherAbedinzadeh. Optimal Reactive Power Dispatch Using an Improved Genetic Algorithm. *International Journal of Computer and Electrical Engineering*. 2012;4:4.
- [17] E. Ismaredah, H. Radiles, "Pemodelan *Genetic Algorithm* Pada Kasus *University Course Scheduling* Dalam Meningkatkan Kualitas Pembelajaran Di Lingkungan Uin Suska Riau," UIN SUSKA, Lembaga Penelitian dan Pengabdian Kepada Masyarakat, 2019.