

Pengaruh Probabilitas Crossover Terhadap Kinerja Algoritma Genetika Dalam Optimasi Penjadwalan Matakuliah

Rudi Salman¹⁾, Suprpto²⁾, Irfandi³⁾

¹Jurusan Pendidikan Teknik Elektro

²Jurusan Pendidikan Teknik Mesin

³Jurusan Pendidikan Fisika

^{1,2}Fakultas Teknik

³Fakultas Matematika dan Ilmu Pengetahuan Alam

^{1,2,3}Universitas Negeri Medan

E-mail : rudisalman@unimed.ac.id¹⁾, suprpto@unimed.ac.id²⁾, irfandi@unimed.ac.id³⁾

Abstract

Genetic Algorithm speed is determined by computation time. Computing time in AG for finding the optimum value is strongly influenced by the following parameters: population size, crossover probability (Pc), mutation probability (Pm), and the selected selection method. Pc is one of the essential parameters in AG. A chromosome that will reach the best solution can be obtained from the crossover process of the two parent chromosomes. The Pc value strongly influences the crossover process. Determining the appropriate and correct Pc value indicates how large the parent chromosome will experience crossover. The method used to analyze the effect of Pc on AG performance is changing the Pc value between 0.80-0.95. The simulation used MATLAB R2012b to obtain the best computational time for each Pc value. Meanwhile, the other AG parameters remained the same: Pm=0.05 and population size=100 for each change of Pc value. The test results using MATLAB R2012b show that the fastest computing time is in the range of Pc values between 0.85-0.95 with an average computation time of 0.14564s. This indicates that for the case of optimizing the scheduling of courses in the Unimed Electrical Engineering study program, the Pc value between 0.85-0.95 will provide the fastest computation time.

Keywords- Crossover probability, genetic algorithm, computation time, course scheduling

Intisari

Kecepatan kinerja Algoritma Genetika (AG) ditentukan oleh waktu komputasi. Waktu komputasi dalam AG untuk pencarian nilai optimum sangat dipengaruhi oleh parameter-parameter : ukuran populasi (popsize), Probabilitas Crossover (Pc), Probabilitas Mutasi (Pm) dan metode seleksi yang dipilih. Pc merupakan salah satu parameter penting dalam AG. Suatu kromosom yang akan mencapai solusi terbaik dapat diperoleh dari proses Crossover (pindah silang) dua kromosom induk. Proses Crossover sangat dipengaruhi oleh nilai Pc. Penentuan nilai Pc yang sesuai dan tepat merupakan indikator, seberapa besar kromosom induk yang akan mengalami crossover. Metode yang digunakan untuk Analisis Pengaruh Pc terhadap kinerja AG dengan cara mengganti nilai Pc antara 0,80-0,95. Simulasi dilakukan menggunakan MATLAB R2012b sehingga diperoleh waktu komputasi terbaik untuk setiap nilai Pc yang dipilih. Sementara parameter-parameter AG yang lain tetap, yaitu : Pm=0,05 dan ukuran populasi=100 untuk setiap pergantian nilai Pc. Hasil pengujian menggunakan MATLAB R2012b menunjukkan bahwa waktu komputasi tercepat berada pada rentang nilai Pc antara 0,85-0,95 dengan rata-rata waktu komputasi 0,14564s. Hal ini menunjukkan bahwa untuk kasus optimasi penjadwalan matakuliah di prodi TE Unimed nilai Pc antara 0,85-0,95, akan memberikan waktu komputasi tercepat.

Kata Kunci—Probabilitas crossover, algoritma genetika, waktu komputasi, penjadwalan matakuliah

1. PENDAHULUAN

Algoritma genetika (AG) merupakan algoritma pencarian berdasarkan mekanisme seleksi alam dan genetika. Algoritma ini memiliki tiga operator genetik utama: ukuran populasi, probabilitas pindah silang (Pc) dan probabilitas mutasi (Pm). Pc adalah operator penting untuk menciptakan keturunan baru dari pasangan

kromosom induk yang dipilih [1],[2]. Operator-operator ini mempunyai fungsi untuk menghasilkan solusi yang mendekati optimal [3]. Algoritma genetika juga menciptakan populasi baru melalui proses iterasi yang terus menerus terhadap populasi awal hingga diperoleh populasi yang mendekati solusi optimal [4].

Struktur dasar algoritma genetika, adalah sebagai berikut :

1. Membangkitkan populasi awal

Populasi awal ini dibangkitkan secara random sehingga didapatkan solusi awal. Populasi itu terdiri atas sejumlah kromosom yang merepresentasikan solusi yang diinginkan.

2. Membentuk generasi baru

Operator reproduksi, seleksi, mutasi, dan crossover digunakan untuk menghasilkan generasi baru. Setelah berulang kali dilakukan, jumlah kromosom yang diperlukan untuk membentuk generasi baru akan dihasilkan. Generasi baru akan berfungsi sebagai representasi dari solusi baru. Istilah "anak" mengacu pada generasi baru. Generasi baru ini akan lebih keadaannya dari generasi-generasi sebelumnya

3. Evaluasi solusi

Pada tiap generasi, kromosom akan dievaluasi dengan alat ukur yang disebut *fitness*. Nilai *fitness* setiap kromosom menggambarkan kualitas kromosom dalam populasi tersebut. Proses ini mengevaluasi setiap populasi dengan menghitung nilai *fitness* setiap kromosom dan mengevaluasinya sampai kriteria berhenti dipenuhi.

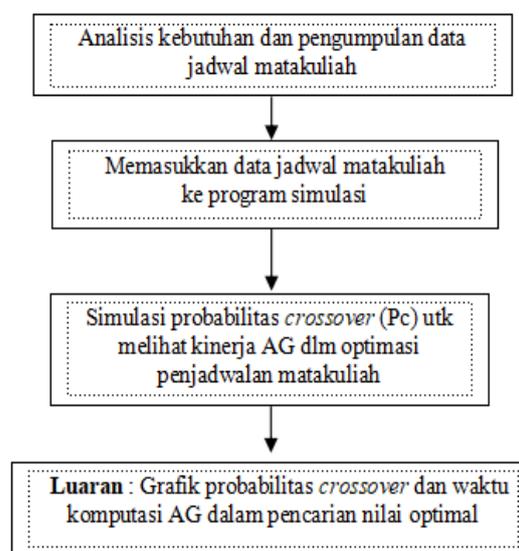
Beberapa operasi menunjang keberhasilannya dalam mencari solusi yang paling optimal, seperti pembangkitan populasi awal, nilai kebugaran setiap kromosom, seleksi kromosom, *Crossover* dan mutasi. Teknik *crossover* adalah salah satu operasi paling dasar dalam evaluasi AG [5]. Proses *crossover* adalah proses pembentukan kromosom anak melalui proses persilangan antara dua kromosom induk [6]. Setiap kromosom akan dievaluasi berdasarkan nilai kebugaran (*fitness*) nya. Nilai kebugaran akan menentukan kualitas dari setiap kromosom anak yang dihasilkan. Proses ini berjalan secara iteratif hingga diperoleh kromosom terbaik. Kromosom dengan nilai kebugaran terbaik mempunyai kemungkinan yang tinggi untuk dapat bereproduksi sehingga menghasilkan kromosom baru pada generasi berikutnya. Dengan demikian, generasi selanjutnya akan lebih baik dari generasi sebelumnya. Ini akan menunjukkan bagaimana kinerja AG. Penjadwalan mata kuliah merupakan pengalokasian sumber daya dalam jangka waktu tertentu dengan memenuhi batasan yang telah ditentukan. Menyusun jadwal kuliah merupakan pekerjaan yang menantang. Ada beberapa kasus terkait penjadwalan mata kuliah; yaitu dosen

tidak dapat mengajar karena jadwal mata kuliah bertabrakan dengan mata kuliah lain, ada mata kuliah yang diadakan dalam ruangan dan waktu yang sama dan seterusnya. Untuk itu perlu adanya optimalisasi penjadwalan mata kuliah. Pada penelitian ini optimasi penjadwalan mata kuliah menggunakan algoritma genetika. Dalam mencari nilai optimal, AG bergantung pada salah satu operator genetika yaitu *Crossover*. *Crossover* berfungsi untuk menghasilkan individu baru yang berbeda dengan kromosom induknya dengan kualitas yang lebih baik [7]. Pengaruh *Pc* terhadap kinerja AG dianalisis dengan memvariasikan nilai *Pc* dan jumlah generasi. Perubahan nilai *Pc* akan terlihat pengaruhnya terhadap waktu komputasi. Waktu komputasi secara dramatis menentukan kinerja AG. Oleh karena itu, pemilihan nilai *Pc* yang sesuai akan menentukan kinerja AG secara signifikan.

2. METODOLOGI

2.1 Tahapan Penelitian

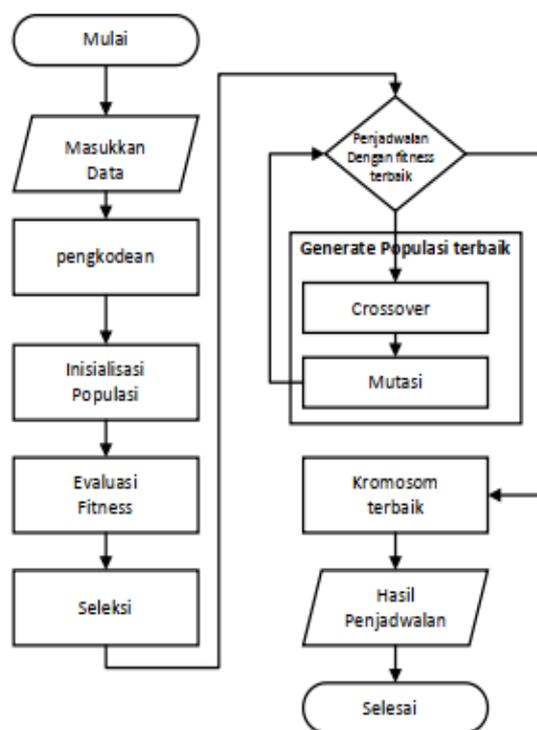
Tahapan awal penelitian adalah melakukan analisis kebutuhan dan pengumpulan jadwal matakuliah, selanjutnya memasukkan atau menginput data jadwal matakuliah ke program simulasi menggunakan Algoritma genetika dalam Matlab R2012b, tahapan berikutnya melakukan simulasi dengan memvariasikan nilai probabilitas *crossover* untuk melihat pengaruhnya terhadap waktu komputasi. Hasil akhir atau luaran simulasi merupakan grafik hubungan antara variasi nilai probabilitas *crossover* dan waktu komputasi. Berikut adalah gambar tahapan penelitian yang dilakukan.



Gambar 1. Bagan tahapan penelitian

2.2 Algoritma Genetika

Algoritma Genetika (AG) adalah algoritma pencarian berbasis genetika dan mekanisme seleksi alam. Untuk pertama kalinya, AG digunakan sebagai algoritma untuk menemukan parameter terbaik. Namun, seiring berjalannya waktu, AG dapat digunakan untuk menangani masalah lain seperti peramalan, pembelajaran, pemrograman otomatis, dan sebagainya. AG banyak digunakan dalam *soft computing* untuk mendapatkan nilai parameter yang optimal. AG memecahkan masalah pada kromosom saat menyelesaikannya. Kromosom ini terdiri dari sejumlah gen yang menunjukkan variabel keputusan yang digunakan dalam pengambilan keputusan yang diinginkan, serta representasi kromosom dan fungsi *fitness*, yang digunakan untuk menentukan seberapa baik suatu kromosom dan sesuai untuk solusi yang dapat diharapkan [8]. Diagram alir simulasi penjadwalan yang menggunakan algoritma genetika ditunjukkan di bawah ini.



Gambar 2. Diagram alir algoritma genetika

1. Pengkodean

Setiap kromosom mengandung beberapa gen yang mengkodekan informasi yang disimpan dalam kromosom; oleh karena itu, pengkodean adalah cara gen dikodekan menjadi kromosom. Karena gen merupakan bagian dari kromosom,

gen harus dikodekan terlebih dahulu dalam bentuk kromosom agar dapat diproses menggunakan AG [9].

2. Menentukan populasi awal dan inialisasi kromosom

Pembentukan sejumlah kromosom secara acak dikenal sebagai penentuan populasi awal. Kromosom adalah pilihan alternatif yang mungkin. Kromosom identik dengan individu. Besar atau kecilnya populasi bergantung pada masalah yang akan diselesaikan. Populasi awal dibangkitkan dengan menginisialisasi solusi yang mungkin pada beberapa kromosom setelah populasi diukur. Pengukuran panjang kromosom didasarkan pada masalah yang akan diselesaikan [10].

3. Evaluasi *fitness*

Fungsi evaluasi dalam algoritma genetika merupakan fungsi yang memberikan penilaian kromosom yaitu nilai *fitness* yang akan digunakan sebagai acuan untuk mencapai nilai optimal dalam suatu algoritma genetika. Nilai *fitness* ini kemudian menjadi nilai bobot kromosom. Saat mengevaluasi kromosom, ada dua hal yang perlu dilakukan, yaitu mengevaluasi fungsi tujuan (*objective function*) dan mengubah fungsi tujuan menjadi fungsi *fitness*. Secara umum fungsi *fitness* ditentukan oleh fungsi tujuan dengan nilai positif, jika ternyata nilai fungsi tujuan negatif maka perlu ditambahkan suatu konstanta x agar nilai *fitness* yang dihasilkan positif [11].

4. Seleksi

Pemilihan kromosom induk untuk dikembangkan berdasarkan nilai *fitness* adalah tujuan seleksi untuk meningkatkan peluang reproduksi dari sebuah populasi [12].

5. *Crossover*

Crossover digunakan sebagai metode persilangan kromosom secara acak dan merupakan penyatuan bagian pertama dari kromosom 1 dengan bagian kedua dari kromosom 2. *Crossover* hanya dapat dilakukan jika bilangan acak yang dihasilkan untuk kromosom kurang dari probabilitas *crossover* yang ditentukan (P_c) biasanya diatur mendekati 1, misalnya 0,80. Metode *crossover* yang paling banyak digunakan adalah *crossover* satu titik potong (*one-point crossover*). Titik persilangan dipilih secara acak, kemudian bagian pertama dari kromosom 1 digabungkan dengan bagian

kedua dari kromosom 2. Untuk menentukan posisi *breakpoint*, dibangkitkan bilangan acak [1 - N], dimana N adalah jumlah gen pada kromosom [13],[14].

6. Mutasi

Mutasi adalah perubahan secara acak nilai satu atau lebih gen pada suatu kromosom. Mutasi merupakan alat algoritma genetika yang digunakan untuk menciptakan suatu yang baik [15]. Selain itu, mutasi digunakan untuk memperbaiki kerusakan genetika yang disebabkan oleh proses *crossover*.

7. Kriteria Pemberhentian

Ada beberapa kriteria terminasi yang dapat digunakan [16] :

- a. Memberikan batasan pada jumlah iterasi yang dapat dilakukan algoritma genetika; jika batas tersebut tercapai, iterasi akan berhenti dan individu dengan nilai fitness tertinggi akan dilaporkan sebagai solusi terbaik.
- b. Memberikan batasan pada waktu yang dapat digunakan algoritma genetika. Kriteria ini digunakan pada sistem waktu nyata, di mana solusi harus ditemukan dalam waktu yang paling lama, misalnya dua belas jam. Oleh karena itu, sudah diantisipasi bahwa proses akan berlangsung hampir dua belas jam.
- c. Menghitung apakah ada atau tidak penggantian anggota populasi secara bertahap hingga jumlah tertentu.

3. HASIL DAN PEMBAHASAN

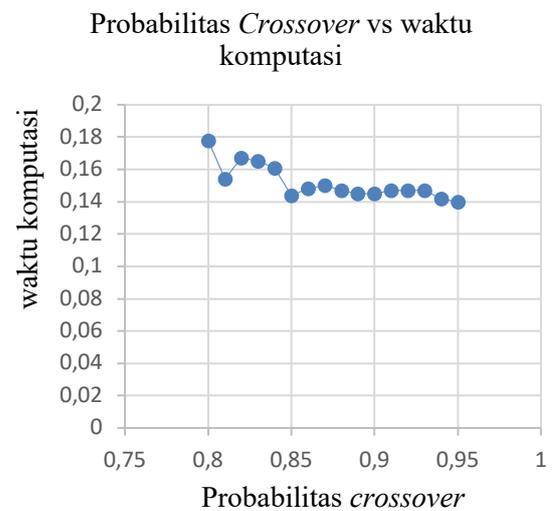
Tabel berikut merupakan tabel hasil simulasi antara variasi nilai probabilitas *crossover* dan waktu komputasi.

Tabel.1 Probabilitas *Crossover* dan Waktu komputasi

No	Probabilitas <i>crossover</i> (Pc)	Waktu komputasi (detik)
1.	0.8	0.178
2.	0.81	0.154
3.	0.82	0.167
4.	0.83	0.165
5.	0.84	0.161
6.	0.85	0.144
7.	0.86	0.148
8.	0.87	0.150

9.	0.88	0.147
10.	0.89	0.145
11.	0.90	0.145
12.	0.91	0.147
13.	0.92	0.147
14.	0.93	0.147
15.	0.94	0.142
16.	0.95	0.140

Berikut ini merupakan grafik hasil simulasi hubungan antara variasi nilai probabilitas *crossover* dan waktu komputasi menggunakan perangkat lunak MATLAB R2012b.



Gambar 3. Grafik nilai probabilitas *crossover* dan waktu komputasi

Dari gambar grafik diatas dapat dilihat bahwa dengan memvariasikan nilai probabilitas *crossover* akan memberikan pengaruh terhadap waktu komputasi. Dimana nilai probabilitas *crossover* ini di variasikan antara 0.80 sampai 0.95, dengan probabilitas mutasi = 0.05 dan ukuran populasi=100, maka diperoleh waktu komputasi rata-rata 0.14564 detik dan waktu komputasi tercepat adalah 0.1440 detik. Waktu komputasi ini menunjukkan bahwa dengan variasi nilai probabilitas *crossover* antara 0.80-0.95, algoritma genetika dapat bekerja secara optimal dalam pencarian solusi khususnya dalam kasus penjadwalan matakuliah.

4. KESIMPULAN DAN SARAN

Berdasarkan hasil penelitian dapat disimpulkan bahwa dengan probabilitas *Crossover* (Pc) antara 0.85-0.95, probabilitas Mutasi (Pm) = 0.05 dan ukuran populasi = 100 diperoleh waktu

komputasi rata-rata 0.14564 detik sedangkan waktu komputasi tercepat 0.1440 detik pada $P_c = 0.85$. Waktu komputasi ini menunjukkan bahwa dengan variasi nilai probabilitas *crossover* antara 0.80-0.95, algoritma genetika dapat bekerja secara optimal untuk mencapai solusi dalam kasus penjadwalan matakuliah di Program Studi Teknik Elektro Unimed.

Dalam penelitian ini simulasi dilakukan dengan memvariasikan nilai Probabilitas *Crossover* (P_c), tetapi ukuran populasi, probabilitas mutasi tetap dijaga konstan nilainya, sedangkan metode seleksi yang digunakan roda roulette (*roulette wheel*). Untuk penelitian yang akan datang diharapkan dapat menggunakan parameter-parameter lain seperti probabilitas mutasi, metode seleksi yang berbeda dan jumlah populasi dapat di variasikan, berikutnya dilihat pengaruhnya terhadap waktu komputasi khusus untuk permasalahan penjadwalan matakuliah.

UCAPAN TERIMA KASIH

Ucapan terima kasih terutama ditujukan kepada Lembaga Penelitian dan Pengabdian Masyarakat (LPPM) Universitas Negeri Medan sebagai pemberi dana riset, melalui SK Ketua LPPM UNIMED Nomor.0067/UN33.8/PPKM/PD/2023. Sehingga riset ini dapat diselesaikan sesuai dengan waktu yang telah ditetapkan.

DAFTAR PUSTAKA

- [1] Ongko,E., Analisis *Performance* atas Metode *Arithmetic Crossover* dalam Algoritma Genetika, Jurnal Teknologi Informasi dan Komunikasi, vol.4,No.2, hal. 76-87,Des 2015.
- [2] Elva.Y., Sistem Penjadwalan Mata Pelajaran Menggunakan Algoritma Genetika, Jur. Teknol. Inf., vol. 3, no. 1, p. 49, 2019.
- [3] Ardiansyah,H.,Junianto,M,B,S.,Penerapan Algoritma Genetika untuk Penjadwalan Mata Pelajaran,J. Media Inf. Budidarma.Vol.6,No.1, pp. 329-336, 2022.
- [4] Arkeman, Y., Seminar, K. B., dan Gunawan, H. Algoritma Genetika, Teori dan Aplikasinya untuk Bisnis dan Industri. Bogor: PT Penerbit IPB Press,2012.
- [5] Raghavendra,B.V., Effect of Crossover Probability on Proformance of Genetic Algorithm in Scheduling of Parallel Machines for BI-Criteria Objectives, Int.Journal of Engineering and Advanced Technology (IJEAT),vol.9, issue-1,2019.
- [6] Oktarina.D, dan Hajjah.A, Perancangan Sistem Penjadwalan Seminar Proposal dan Sidang Skripsi dengan Metode Algoritma Genetika, JOISIE (Journal Inf. Syst. Informatics Eng.), vol. 3, no.1, p.32 2019.
- [7] Ginantra.N.L.W.S.R., dan Anandita.I.B.G., Implementasi Algoritma Genetika Berbasis Web Pada Sistem Penjadwalan Mengajar Di Smk Dwijendra Denpasar, J. Teknol. Inf. dan Komput., vol. 5, no. 1, pp. 130–138, 2019.
- [8] Handayani.T., Fudholi.D.H., dan Rani.S., Kajian Algoritma Optimasi Penjadwalan Mata Kuliah, J. Pengkaj. Dan Penerapan Tek. Inform., vol. 13, no. 2, pp. 212–222, 2020.
- [9] Sugeha.I.H., dan Inkiriwang.R.L., Optimasi Penjadwalan Menggunakan Metode Algoritma Genetika Pada Proyek Rehabilitasi Puskesmas Minanga, J. Sipil Statik, vol. 7, no. 12, 2019.
- [10] Sobirin.S., Implementasi Algoritma Genetika untuk Penjadwalan Kuliah, Jutikomp, vol. 1, no. 2, pp. 188–194, 2018.
- [11] Nugroho.A., Priatna.W., dan Romli.I., Implementasi Algoritma Genetika Untuk Optimasi Penjadwalan Mata Kuliah, J.Teknol. dan Ilmu Komput. Prima, vol. 1, no. 2, pp. 35–41, 2018.
- [12] Jonathan.C.N.,et al., Implementasi Metode Algoritma Genetika Pada Penentuan Menu Makanan Untuk Membentuk Berat Badan Ideal, J.Teknol. Inf. dan Terap., vol. 6, no. 1, pp. 35–40, 2019.
- [13] Sari.Y., Alkaff.M., Wijaya.E.S., Soraya.S., dan Kartikasari.D.P., Optimasi Penjadwalan Mata Kuliah Menggunakan Metode Algoritma Genetika Dengan Teknik Tournament Selection, J. Teknol. Inf. dan Ilmu Komput., vol. 6, no. 1, pp. 85–92,2019, doi: 10.25126/jtiik.201961262
- [14] Juwita.A.R., Pratama.A.R., dan Triono.T., Implementasi Algoritma Particle Swarm Optimization untuk Penjadwalan Perkuliahan di Fakultas Teknik Dan Ilmu Komputer Universitas Buana Perjuangan Karawang, J. Sisfotek Glob., vol.10, no. 1, pp. 18–26, 2020.
- [15] Ongko,E., Analisis pengaruh Mutasi terhadap Performance Algoritma Genetika, Jurnal Teknik Informatika Kaputama (JTIK), vol.1,No.1, Jan 2017.

- [16] Saud,A,T., Nugraha,D,W., dan Dodu,A,Y,E., Sistem Penjadwalan Perkuliahan Menggunakan Algoritma Genetika (Studi Kasus Pada Jurusan Teknologi Informasi Fakultas Teknik Universitas Tadulako), J. Ilm. Mat.dan Terapan., vol.14, no. 2, pp. 242–255, 2017