

Penerapan Metode Algoritma Genetika Untuk Optimasi Penjadwalan Mata Kuliah

A'an Tamim Ma'arif¹, Damar Putra Pamungkas²

^{1,2}Teknik Informatika, Fakultas Teknik, Universitas Nusantara PGRI Kediri

E-mail: ^{*1}contact@aatamim.id, ²damar@unpkediri.ac.id

Abstrak – Pada saat ini penjadwalan mata kuliah pada jurusan Teknik Informatika Fakultas Teknik Universitas Nusantara PGRI Kediri menggunakan cara manual. Pembuatan jadwal yang masih manual seringkali mengalami kesulitan dalam penataan slot jadwal dalam mengatur komponen penyusun jadwal yaitu mata kuliah, dosen, hari, jam, dan ruang yang tersedia. Untuk menyelesaikan masalah penjadwalan mata kuliah diperlukan algoritma yang dapat menyelesaikan masalah pada proses penyusunan jadwal. Algoritma ini melakukan proses optimasi untuk mencari hasil yang terbaik. Yaitu dengan cara kombinasi perkawinan yang didasari secara random. Penerapan Algoritma Genetika dalam proses penjadwalan kuliah ini, dengan cara mengkodekan dosen, kelas, mata kuliah, ruang, jam dan hari. Pengujian ini dilakukan dengan melakukan pengujian terhadap kromosom-kromosom yang ada. Berdasarkan nilai fitness yang diperoleh dari setiap pengujian, maka membuktikan bahwa Algoritma Genetika ini mampu memenuhi setiap kromosom yang ada.

Kata Kunci — Algoritma Genetika, Penjadwalan, Optimasi

1. PENDAHULUAN

Pada sebuah proses perkuliahan selalu diawali beberapa kegiatan yang dilakukan oleh pihak universitas sebelum perkuliahan dimulai, salah satunya adalah membuat penjadwalan mata kuliah. Penjadwalan mata kuliah merupakan bentuk kegiatan perencanaan yang disusun dengan mempertimbangkan beberapa komponen seperti dosen, waktu, kelas, ruang, dan mata kuliah itu sendiri.

Pada jurusan Teknik Informatika Fakultas Teknik Universitas Nusantara PGRI Kediri proses penjadwalan perkuliahan masih dilakukan manual. Pembuatan jadwal mata kuliah yang manual seringkali mengalami kesulitan dalam penataan slot jadwal agar tidak terjadi bentrok dengan mata kuliah lain. Hal ini terjadi karena ada beberapa faktor yang memengaruhi seperti kelas yang banyak, jumlah ruangan yang terbatas, dan jumlah dosen yang terbatas. Namun hal ini tidak akan menjadi masalah jika sebuah jurusan dalam fakultas memiliki kelas yang sedikit.

Untuk mengatasi permasalahan tersebut, perlu dibuat sebuah penjadwalan perkuliahan yang optimal melalui pendekatan Algoritma Genetika untuk mendapatkan suatu nilai solusi optimal terhadap suatu permasalahan yang mempunyai banyak kemungkinan solusi.

Penelitian ini menggunakan beberapa referensi penelitian yang dilakukan oleh peneliti sebelumnya. Penelitian terdahulu pertama yang menggunakan algoritma genetika [1] yang telah menghasilkan jadwal yang optimal, terbukti dengan tidak terdapat konflik jadwal dosen, kelas dan ruangan pada periode

yang sama dan terbentuknya kombinasi jadwal dengan nilai konflik terkecil dari jadwal awal yang dibentuk secara random. Penelitian terdahulu kedua [2] yang telah menghasilkan jadwal mengajar yang optimal dan sesuai dengan apa yang diharapkan. Penelitian terdahulu ketiga [3] yang telah menghasilkan penjadwalan tidak memiliki bentrok antara jadwal 1 dan jadwal lainnya.

Berdasarkan latar belakang di atas, rumusan permasalahannya adalah bagaimana merancang penjadwalan mata kuliah pada jurusan Teknik Informatika Fakultas Teknik Universitas Nusantara PGRI Kediri dengan menggunakan metode Algoritma Genetika dan bagaimana mempermudah dalam penyusunan jadwal agar tidak terjadi bentrok pada mata kuliah.

Batasan masalah dalam penelitian ini adalah studi kasus yang diambil adalah jurusan Teknik Informatika Fakultas Teknik Universitas Nusantara PGRI Kediri.

Tujuan yang ingin dicapai dari penelitian ini adalah untuk merancang dan membangun penjadwalan mata kuliah pada jurusan Teknik Informatika Fakultas Teknik Universitas Nusantara PGRI Kediri dengan metode Algoritma Genetika.

2. METODE PENELITIAN

Metode yang digunakan pada penelitian ini adalah metode penelitian kualitatif dan kuantitatif. Metode kualitatif dalam penelitian ini meliputi tahapan penemuan masalah yang akan diteliti kemudian mengkaji studi literatur yang berkaitan dengan cara untuk menyelesaikan masalah yang ada

dan mengunduh jadwal kuliah di website resmi Teknik Informatika Fakultas Teknik Universitas Nusantara PGRI Kediri. Untuk metode kuantitatif dalam penelitian ini yaitu pada tahapan mengolah data jadwal yang telah didapatkan.

2.1 Analisa Kebutuhan Data

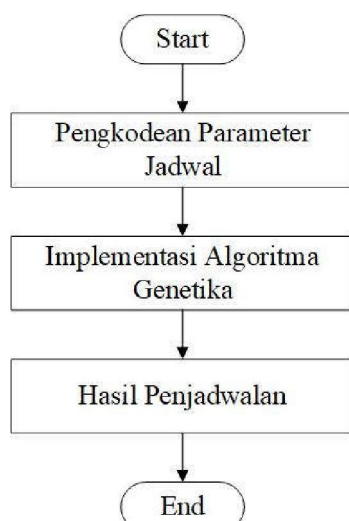
Pengumpulan data dilakukan untuk memperoleh informasi yang dibutuhkan dalam penelitian. Data-input yang dibutuhkan dijelaskan dalam tabel berikut:

Tabel 1. Data Input

Nama Data	Deskripsi
Data Waktu	Merupakan data waktu yang digunakan dalam penjadwalan (hari dan jam).
Data Jam	Merupakan data jam dalam sehari perkuliahan.
Data Hari	Merupakan data hari yang tersedia dalam perkuliahan.
Data Kuliah	Merupakan data kuliah penjadwalan berisi mata kuliah, SKS, kelas, dan dosen.
Data Mata Kuliah	Merupakan data mata kuliah yang ada di penjadwalan.
Data Dosen	Merupakan data dosen yang mengajar di dalam penjadwalan.
Data Kelas	Merupakan data kelas yang tersedia di penjadwalan.
Data Ruang	Merupakan data ruang yang tersedia di penjadwalan.

2.2 Arsitektur Sistem

Desain arsitektur untuk membangun sistem penjadwalan mata kuliah ini menggunakan flowchart dari Algoritma Genetika. Komponen-komponen Algoritma Genetika tersebut dapat dilihat pada Gambar 1.



Gambar 1. Arsitektur Sistem

Komponen-komponen yang terdapat dalam arsitektur sistem Algoritma Genetika untuk penjadwalan pada gambar di atas dijelaskan sebagai berikut:

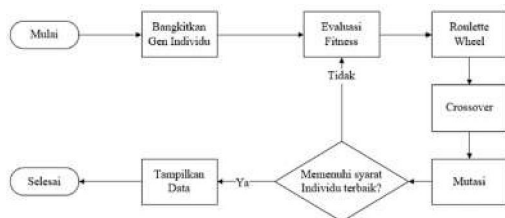
- a. Pengkodean Parameter Jadwal
Semua parameter penyusun jadwal yang digunakan untuk proses Algoritma Genetika harus dikodekan agar susunan kromosom lebih teratur, misal ruang M11 bisa dikodekan R01 dan begitu seterusnya.
- b. Implementasi Algoritma Genetika
Setelah parameter penyusun jadwal dikodekan, proses Algoritma Genetika untuk pembuatan jadwal bisa diterapkan.
- c. Hasil Penjadwalan
Hasil penjadwalan dari Algoritma Genetika bisa didapatkan setelah proses implementasi Algoritma Genetika.

2.3 Algoritma Genetika

Algoritma Genetika adalah algoritma pencarian yang didasarkan pada mekanisme seleksi ilmiah dan genetika alamiah. Konsep yang ada dalam kaidah genetika diterapkan dalam algoritma komputasi yang dapat menyelesaikan masalah dengan constrain tinggi [4].

Algoritma Genetika didasarkan pada proses genetik yang ada dalam makhluk hidup, yaitu perkembangan generasi dalam sebuah populasi yang alami, secara lambat laun mengikuti prinsip seleksi alam. Dengan meniru teori evolusi ini, Algoritma Genetika dapat digunakan untuk mencari solusi permasalahan-permasalahan dalam dunia nyata. Pada awal perkembangannya, metode Algoritma Genetika ini pertama kali diperkenalkan oleh John Holland dari Universitas Michigan pada tahun 1975 dalam bukunya yang berjudul "*Adaption in Natural and Artificial Systems*", dan pada akhirnya dipopulerkan oleh salah seorang muridnya, David Goldberg, yang mampu memecahkan masalah sulit yang melibatkan kontrol transmisi gas-pipa untuk disertasinya yang berjudul "*Computer-aided gas pipeline operation using genetic algorithms and rule learning*" [5].

Proses seleksi alamiah ini melibatkan perubahan gen yang terjadi pada individu melalui proses perkembang biakan. Dalam Algoritma Genetika ini, proses perkembang-biakan ini menjadi proses dasar yang menjadi perhatian utama, dengan dasar berpikir "Bagaimana mendapatkan keturunan yang lebih baik". Algoritma Genetika ini ditemukan oleh John Holland dan dikembangkan oleh muridnya David Goldberg. Secara umum Algoritma Genetika menyelesaikan permasalahan dengan alur yang terdapat pada Gambar 2.



Gambar 2. Penyelesaian Algoritma Genetika Oleh David Goldberg

Gambar di atas menerangkan langkah-langkah penyelesaian suatu masalah menggunakan metode Algoritma Genetika menurut David Goldberg.

2.4 Implementasi Algoritma Genetika

a. Populasi Awal

Dalam menentukan sebuah populasi dirancang dahulu bentuk kromosom yang akan dipakai dalam penjadwalan mata kuliah yaitu sebagai berikut:

1) Kuliah

Terdiri dari mata kuliah, data dosen dan data kelas yang tersedia, disimbolkan dengan K.

2) Ruang

Terdiri dari ruang kelas yang tersedia, disimbolkan dengan R.

3) Waktu

Terdiri dari waktu kuliah yang tersedia, disimbolkan dengan T.

Sehingga susunan object pada kromosom tersebut adalah [K, R, T].

Panjang kromosom ditentukan oleh gabungan gen, satu gen berisi data kuliah, data ruang, dan data waktu. Sebagai contoh untuk pembentukan kromosom dapat dilihat seperti pada tabel 2, tabel 3, dan tabel 4.

Tabel 2. Contoh Data Kuliah

Kode Kuliah	Kelas	Matkul	Dosen
K01	4A	Menpro	Kartika
K02	4A	B. Inggris	Yunik
K03	4A	PITI	Bagus
K04	4A	Technopreneur	Bagus

Tabel 3. Contoh Data Ruang

Kode Ruang	Nama Ruang
R01	M11
R02	M12

Tabel 4. Contoh Data Waktu

Kode Waktu	Hari	Jam
T01	Senin	07:50
T02	Senin	08:45

T03	Senin	09:35
T04	Senin	10:30
T05	Senin	11:20
T06	Selasa	07:50
T07	Selasa	08:45
T08	Selasa	09:35
T09	Selasa	10:30
T10	Selasa	11:20

Diasumsikan dalam satu populasi yang terbentuk berjumlah 4 kromosom dengan jumlah data kuliah yang ada serta masing-masing kromosom memiliki 4 gen. Untuk penyusunan populasi awal diambil dari tabel kuliah, tabel ruang, dan tabel waktu. Dari ketiga tabel tersebut dipilih secara acak untuk penyusunan populasi, dimana K01 adalah kode untuk kuliah ke 1, R01 adalah kode untuk ruang ke 1, T01 adalah kode untuk waktu ke 1. Berikut adalah susunan populasi awal yang ditentukan secara acak.

Kromosom[1] = [K01, R01, T01], [K02, R02, T04],

[K03, R01, T05], [K04, R01, T05]

Kromosom[2] = [K01, R01, T02], [K02, R02, T03],

[K03, R01, T06], [K04, R02, T03]

Kromosom[3] = [K01, R02, T04], [K02, R02, T09],

[K03, R01, T03], [K04, R02, T09]

Kromosom[4] = [K01, R01, T08], [K02, R01, T08],

[K03, R01, T08], [K04, R02, T04]

Dalam penyusunan kode tiap gen dipilih secara acak. Pada contoh terdapat 4 kromosom masing-masing memiliki 4 gen yang artinya terdapat 4 kuliah. Setiap gen adalah kombinasi [kuliah, ruang, waktu].

b. Fungsi Fitness

Dalam tahap seleksi dilakukan pemilihan individu terbaik dengan fungsi *fitness*. Hasil dari fungsi ini untuk mendapatkan solusi optimal yang didapat dari kromosom dengan nilai *fitness* tertinggi.

Kromosom[1] = [K01, R01, T01], [K02, R02, T04],

[K03, R01, T05], [K04, R01, T05]

Kromosom[2] = [K01, R01, T02], [K02, R02, T03],

[K03, R01, T06], [K04, R02, T03]

Kromosom[3] = [K01, R02, T04], [K02, R02, T09],

[K03, R01, T03], [K04, R02, T09]

Kromosom[4] = [K01, R01, T08], [K02, R01, T08],

[K03, R01, T08], [K04, R02, T04]

Dari susunan populasi di atas terdapat pelanggaran pada semua kromosom. Pada kromosom 1 terjadi kesamaan pada 2 gen yang sama yaitu pada gen 3 dan 4 yang memiliki kesamaan pada ruang (R01) dan waktu kuliah (T05). Pada kromosom 2 terjadi kesamaan pada gen 2 dan 4 yang memiliki kesamaan pada ruang (R02) dan waktu kuliah (T03). Pada kromosom 3 terjadi kesamaan pada gen 2 dan 4 yang memiliki kesamaan pada ruang (R02) dan waktu

kuliah (T09). Pada kromosom 4 terjadi kesamaan pada gen 1, 2 dan 3 yang memiliki kesamaan pada ruang (R01) dan waktu kuliah (T08). Dari pelanggaran yang ada akan menghasilkan fitness sebagai berikut :

$$\begin{aligned} \text{Fitness Kromosom 1} &= 1/(1+2) = 0,33 \\ \text{Fitness Kromosom 2} &= 1/(1+2) = 0,33 \\ \text{Fitness Kromosom 3} &= 1/(1+2) = 0,33 \\ \text{Fitness Kromosom 4} &= 1/(1+3) = 0,25 \end{aligned}$$

c. Seleksi (Route Wheel)

Langkah pertama yang akan dilakukan adalah menghitung total *fitness* dari populasi, dapat dilihat seperti pada tabel 5.

Tabel 5. Nilai *Fitness*

Kromosom	Nilai <i>Fitness</i>
1	0,33
2	0,33
3	0,33
4	0,25
Total nilai <i>fitness</i>	1,25

Langkah kedua adalah menghitung probabilitas masing-masing individu dengan membagi nilai *fitness* tiap kromosom dengan total nilai *fitness*, sehingga didapat hasil seperti tabel 6.

Tabel 6. Probabilitas Nilai *Fitness*

Kromosom	Probabilitas
1	$0,33 / 1,25 = 0,26$
2	$0,33 / 1,25 = 0,26$
3	$0,33 / 1,25 = 0,26$
4	$0,25 / 1,25 = 0,2$

Langkah ketiga adalah menempatkan masing-masing kromosom pada *interval* nilai acak 0-1 sejumlah kromosom, dapat dilihat seperti pada tabel 7.

Tabel 7. *Interval* Nilai Probabilitas

Kromosom	<i>Interval</i> Nilai
1	0 – 0,26
2	0,27 – 0,53
3	0,54 – 0,8
4	0,81 – 1

Misal bilangan yang dibangkitkan adalah [0,2; 0,7; 0,4; 0,97]. Dari nilai yang dibangkitkan tersebut dapat dilihat kromosom 0,2 adalah kromosom 1 yang memiliki nilai *interval* 0 – 0,26. Jadi, kromosom 1 tidak mengalami seleksi karena nilai yang dibangkitkan sesuai dengan nilai *interval* kromosom. Selanjutnya kromosom yang memiliki nilai 0,7 adalah kromosom 3 yang memiliki nilai *interval* 0,54 – 0,8 maka kromosom 3 mengalami seleksi ke kromosom 2, dan kromosom 2 secara otomatis juga mengalami seleksi dengan mengisi posisi di kromosom 3. Kromosom 4 tidak mengalami seleksi karena nilai acak yang dibangkitkan telah sesuai

dengan nilai *interval*. Maka susunan populasi baru hasil dari proses seleksi adalah sebagai berikut :

$$\begin{aligned} \text{Kromosom}[1] &= [K01, R01, T01], [K02, R02, T04], \\ & [K03, R01, T05], [K04, R01, T05] \\ \text{Kromosom}[2] &= [K01, R02, T04], [K02, R02, T09], \\ & [K03, R01, T03], [K04, R02, T09] \\ \text{Kromosom}[3] &= [K01, R01, T02], [K02, R02, T03], \\ & [K03, R01, T06], [K04, R02, T03] \\ \text{Kromosom}[4] &= [K01, R01, T08], [K02, R01, T08], \\ & [K03, R01, T08], [K04, R02, T04] \end{aligned}$$

d. Crossover (Kawin Silang)

Misal nilai probabilitas *crossover* diatur 0,5 (mendekati nilai 1). Metode kawin silang yang digunakan adalah *One Point Crossover* (pindah silang satu titik potong). Satu titik potong dipilih secara acak kemudian bagian pertama kromosom induk 1 digabungkan dengan bagian kedua dari kromosom induk 2.

Bilangan acak yang ditentukan untuk menentukan posisi titik potong adalah [1-N] dimana N merupakan banyaknya gen dalam satu kromosom. Dari contoh bilangan acak yang dibangkitkan di atas yang mengalami *crossover* adalah kromosom 1 dan 3 yang bernilai 0,2 dan 0,4 karena kurang dari nilai probabilitas yang sudah diatur. Untuk posisi potong dipilih adalah posisi gen ke-2, maka proses kawin silangnya adalah:

$$\begin{aligned} \text{Kromosom}[1] &= [K01, R01, T01], [K02, R02, T04], \\ & [K03, R01, T05], [K04, R01, T05] \\ \text{Kromosom}[3] &= [K01, R01, T02], [K02, R02, T03], \\ & [K03, R01, T06], [K04, R02, T03] \end{aligned}$$

Hasil kawin silang kedua kromosom di atas adalah sebagai berikut :

$$\begin{aligned} \text{Kromosom}[1] &= [K01, R01, T01], [K02, R02, T03], \\ & [K03, R01, T06], [K04, R02, T03] \\ \text{Kromosom}[3] &= [K01, R01, T02], [K02, R02, T04], \\ & [K03, R01, T05], [K04, R01, T05] \end{aligned}$$

Hitungan *fitness* sesudah pindah silang :

$$\begin{aligned} \text{Fitness Kromosom 1} &= 1/(1+2) = 0,33 \\ \text{Fitness Kromosom 3} &= 1/(1+2) = 0,33 \end{aligned}$$

e. Mutasi

Gen yang dimutasi hanya diganti ruang dan waktu saja, untuk kuliahnya tetap. Jumlah gen yang diganti tergantung Mutation Rate. Misal ada 4 kromosom dengan 4 gen dan misalkan mutation rate diatur 50 persen (bilangan 1 - 100), maka :

$$\begin{aligned} \text{Total Gen} &= 4 \times 4 \text{ (jumlah kromosom} \times \text{jumlah} \\ & \text{gen per kromosom)} = 16 \text{ gen} \\ \text{Total Mutasi} &= 50\% \times 16 = 8 \text{ gen.} \end{aligned}$$

Cara mutasi dengan membangkitkan bilangan acak antara 1 sampe total gen (16) sebanyak 8 kali.

Misal bilangan acak pertama adalah 4 maka akan diambil kromosom pertama gen ke empat. Sehingga ruang dan waktu gen ke empat di kromosom pertama akan diganti dengan mengambil data ruang dan waktu secara acak.

Kromosom sebelum mutasi :

Kromosom[1] = [K01, R01, T01], [K02, R02, T03], [K03, R01, T06], [K04, R02, T03]

Kromosom setelah mutasi :

Kromosom[1] = [K01, R01, T01], [K02, R02, T03], [K03, R01, T06], [K04, R02, T09]

Dari proses mutasi didapatkan susunan kromosom baru sebagai berikut :

Kromosom[1] = [K01, R01, T01], [K02, R02, T03], [K03, R01, T06], [K04, R02, T09]

Kromosom[2] = [K01, R02, T04], [K02, R02, T09], [K03, R01, T03], [K04, R02, T09]

Kromosom[3] = [K01, R01, T02], [K02, R02, T04], [K03, R01, T05], [K04, R01, T05]

Kromosom[4] = [K01, R01, T08], [K02, R01, T08], [K03, R01, T08], [K04, R02, T04]

Nilai *fitness* tiap kromosom dari hasil proses mutasi adalah sebagai berikut :

Fitness Kromosom 1 = $1/(1+0) = 1$

Fitness Kromosom 2 = $1/(1+2) = 0,33$

Fitness Kromosom 3 = $1/(1+2) = 0,33$

Fitness Kromosom 4 = $1/(1+3) = 0,25$

Dari hasil *fitness* di atas dipilih kromosom 1 sebagai kromosom terbaik karena telah mencapai nilai *fitness* = 1.

f. Kondisi Selesai

Kondisi selesai yang dapat menghentikan algoritma genetika jika jumlah generasi atau iterasi maksimum terpenuhi.

3. HASIL DAN PEMBAHASAN

Data mata kuliah yang dipakai dalam uji coba hanya 1 kelas dengan 4 mata kuliah berdasarkan contoh data pada tabel 2, 3, dan 4.

Hasil akhir dari keseluruhan proses dapat dilihat pada tabel 8 yang merupakan hasil dari kromosom 1 karena nilai *fitness* telah mencapai 1.

Tabel 8. Contoh Hasil Proses

Jam	Waktu	Senin	Ruang	Dosen	Selasa	Ruang	Dosen
2	07:50-08:40	Menpro	M11	Kartika	PITI	M12	Bagus
3	08:45-09:35	Menpro	M11	Kartika	PITI	M12	Bagus
4	09:35-10:25	B. Inggis	M12	Yunik	PITI	M12	Bagus
5	10:30-11:20	B. Inggis	M12	Yunik	Technopreneur	M12	Bagus
6	11:20-12:10				Technopreneur	M12	Bagus

Berdasarkan hasil jadwal pada tabel 8, terlihat tidak terdapat bentrok mata kuliah satu dengan yang lain. Ini membuktikan bahwa Algoritma Genetika mampu untuk mengatasi permasalahan yang memiliki banyak kemungkinan solusi seperti pada kasus penjadwalan.

4. SIMPULAN

Proses pembuatan jadwal pada jurusan Teknik Informatika Fakultas Teknik Universitas Nusantara PGRI Kediri dilakukan dengan menggunakan metode Algoritma Genetika yang menghasilkan jadwal tanpa adanya bentrok mata kuliah satu dengan yang lain. Melalui langkah-langkah yang digunakan dalam Algoritma Genetika yaitu merepresentasikan nilai kromosom, menginisialisasi populasi awal, melakukan seleksi, crossover, mutasi, sampai kondisi selesai.

5. SARAN

Penelitian selanjutnya dapat mengimplementasikan metode Algoritma Genetika untuk penjadwalan dalam bentuk aplikasi *cross platform* yang dapat digunakan secara langsung.

DAFTAR PUSTAKA

- [1] Tambunan, Leonard. 2017. Implementasi Algoritma Genetika dalam Pembuatan Jadwal Kuliah. Jaringan Sistem Informasi Robotik. Vol. 1, No. 01.
- [2] Saryanti, I Gusti Ayu Desi. dan Wijanegara, I Kadek. 2017. Penerapan Metode Algoritma Genetika Untuk Penjadwalan Mengajar. Jurnal SIMETRIS. Vol 8 No 1.
- [3] Ivan., Raphae, Stephanus. dan Agung, Halim. 2018. Aplikasi Penjadwalan Mata Pelajaran Di SMAN 31 menggunakan Algoritma Genetika Berbasis Web. Jurnal SIMETRIS. Vol. 9 No. 1.
- [4] Suyanto. 2005. Algoritma Genetika dalam MATLAB. Yogyakarta: Andi Offset.
- [5] Goldberg, D. 1987. Computer-aided gas pipeline operation using genetic algorithm and rule learning. e-ISSN: 1435-5663, e-ISSN 0177-0667.