

## **BAB II**

### **STUDI LITERATUR**

#### **2.1 Kajian Pustaka**

Penelitian terdahulu yang terkait dengan penjadwalan mata kuliah dengan beragam metode akan digunakan sebagai acuan dalam penelitian ini. Penelitian yang dilakukan oleh Andini Puspasari, dkk. [3] Penyelesaian masalah penjadwalan menggunakan algoritma genetika. Data yang digunakan adalah sebanyak 40 slot untuk 2 SKS, dan total 50 slot untuk 3 SKS. Jumlah slot dalam jadwal perkuliahan adalah 120 slot. Kromosom yang memiliki mata kuliah 2 sks berjumlah 23 gen dan kromosom mata kuliah 3 sks berjumlah 39 gen. Lalu dilanjutkan terus menerus sehingga mencapai nilai kromosom yang lebih tinggi dari sebelumnya. Parameter dengan jumlah populasi sebesar 70, mutasi sebesar 0.4, generasi sebesar 50 memiliki nilai dengan solusi optimal yang paling besar. Selanjutnya penelitian yang dilakukan oleh Dwiprima Elvanny Myori [4] kombinasi logika *Fuzzy* dan algoritma genetika untuk masalah penjadwalan mata kuliah. Menggunakan data dengan mata kuliah 3 sks dan 4 sks. Dengan melakukan pembentukan 6 kromosom yang terdiri dari kode dosen, kode mata kuliah, kode waktu dan kode ruangan. Peran sistem inferensi *Fuzzy* pada algoritma genetika sangat membantu dan dapat meminimalisir iterasi yang dilakukan pada saat *crossover* dan mutasi pada Algoritma Genetika.

Pada penelitian yang dilakukan Setyoningsih Wibowo [5], menerapkan Logika *Fuzzy* dalam penjadwalan mata kuliah. Pada tahapan proses desain sistem *Fuzzy* secara umum dari metode Mamdani dan metode Sugeno. Penelitian ini dilakukan dengan dua kali pengujian yaitu pada data semester gasal tahun akademik 2012/2013 dan pada data semester genap tahun akademik 2012/2013, yang menghasilkan nilai akurasi metode Mamdani sebesar 90.12%, dan metode Sugeno menghasilkan nilai akurasi sebesar 70.63% yang dimana memiliki selisih nilai akurasi sebesar 19.60%, maka dengan nilai perbandingan yang dihasilkan bahwa metode mamdani mempunyai nilai akurasi yang lebih tinggi dibandingkan dengan nilai akurasi yang dimiliki dengan menggunakan metode Sugeno. Selanjutnya penelitian yang dilakukan oleh Henry Wulandar Pangestu, dkk. [6] Implementasi Algoritma *Fuzzy* Evolusi pada penjadwalan perkuliahan. Proses *Fuzzy*

yang digunakan pada dua buah masukan dan menghasilkan dua buah keluaran. Dua buah masukan tersebut adalah jumlah populasi 100 dan jumlah generasi 200 dan dua buah keluaran yang diperoleh adalah probabilitas *crossover* dan probabilitas mutasi yang menyelesaikan permasalahan benturan yang dilakukan dengan cara mengubah ke dalam bentuk kromosom.

Penelitian mengenai penjadwalan yang dilakukan oleh Wikatasari, dkk. [9] Metode Simulated Annealing untuk Optimasi Penjadwalan Perkuliahan Perguruan Tinggi. Dengan menggunakan metode Simulated Annealing yang menggunakan data mata kuliah sebanyak 152, data dosen sebanyak 52, data hari sebanyak 5, data waktu periode per hari sebanyak 10 dan data ruang sebanyak 9 menghasilkan rata-rata berbagai alokasi mata kuliah pada slot waktu yang tersedia adalah 77.791 yang memiliki standar deviasi sebesar 1.982803. Selanjutnya penelitian berikut dilakukan oleh Titus Kristanto, dkk. [7] Penjadwalan ruang kuliah menggunakan Vertex Graph Coloring dan Simulated Annealing. Penjadwalan ruang kuliah dengan vertex Graph Coloring, dapat memenuhi aturan yang ditetapkan, terdiri dari hard constraint dan soft constraint. Hasil prosentase keberhasilan hard constraint 98,47%, yang memiliki pelanggaran hard constraint 1,53%, hasil prosentase keberhasilan soft constraint 82,4%, yang memiliki pelanggaran soft constraint 17,6%, sehingga dengan hasil yang diperoleh dapat disimpulkan keberhasilan sistem yaitu 90,435%.

Penelitian yang dilakukan oleh Rusianah, dkk [11] Implementasi Algoritma Genetik-Tabu Search dalam Optimasi Penjadwalan Perkuliahan. Tabu *search* berguna untuk menyimpan solusi yang pernah ditemui selama iterasi. Hasil dari Penelitian adalah performa yang tinggi dengan konvergen *fitness* = 1 pada parameter probabilitas crossover 0.6, waktu proses selama 59 detik di generasi ke-569. Sedangkan untuk penggunaan algoritma genetika menghasilkan performa yang tinggi dengan konvergen *fitness* = 1 pada parameter probabilitas crossover 0.6, waktu proses selama 70 detik pada generasi ke-754. Diketahui waktu proses menggunakan AG-TS selama 59 detik dan AG selama 70 detik, yang artinya waktu proses AGTS untuk mencapai konvergen *fitness* = 1 lebih cepat dari pada algoritma genetika. Selanjutnya penelitian yang dilakukan oleh Yulia Yudihartanti [20] Analisis Komparasi Metode *Fuzzy* Mamdani dan Sugeno dalam Penjadwalan Mata

Kuliah. Penelitian dilakukan dengan menggunakan dua kali proses pengujian yaitu pada jadwal dengan tahun yang berbeda, menghasilkan tingkat akurasi pada metode mamdani sebesar 81,08% sedangkan metode sugeno menghasilkan akurasi sebesar 27,02%. Sehingga komparasi yang dihasilkan menunjukkan tingkat akurasi yang lebih tinggi menggunakan metode mamdani.

Penelitian tentang optimasi algoritma yang dilakukan oleh Fitri Anggasari, dkk [9] Optimasi Kebutuhan Gizi untuk Balita Menggunakan Hybrid Algoritma Genetika dan *Simulated Annealing*. *Simulated Annealing* berfungsi untuk mengambil individu yang lolos seleksi atau memiliki nilai fitness terbaik. Pengujian yang dilakukan menghasilkan rata-rata nilai fitness berdasarkan pengujian ukuran populasi terbesar yakni 0.067. Untuk hasil pengujian banyaknya generasi sebesar 0.07072 sedangkan hasil pengujian kombinasi nilai Cr dan Mr sebesar 0.067132. Untuk hasil pengujian nilai alpha sebesar 0.0705 dan hasil pengujian nilai T0 sebesar 0.07075. Sehingga dari analisa hasil pengujian yang dilakukan menghasilkan rata-rata nilai fitness sebesar 0.10106. Selanjutnya penelitian tentang algoritma genetika yang dilakukan oleh Syaiful Muzid [2] Dinamiasi Parameter Algoritma Genetika menggunakan Population Resizing on *Fitness Improvement Fuzzy Evolutionary Algorithm*. Teknologi hybrid antara algoritma genetika dan logika *Fuzzy* serta teknik PROFIGA atau disebut dengan model PROFIFEA mampu meningkatkan hasil akhir atau solusi yang ditemukan. Dinamisasi parameter dalam PROFIFEA memudahkan dalam penyelesaian permasalahan optimasi khususnya pada kasus Travelling Salesman Problem (TSP) sehingga pengguna tidak kesulitan dalam menentukan nilai dari parameter yang digunakan dalam proses running dan pembentukan populasi baru.

Dilihat dari beberapa penelitian terkait. Pergabungan algoritma genetika banyak dilakukan. Pergabungan dilakukan karena algoritma genetika memiliki kelemahan. Pergabungan menggunakan algoritma tabu *search* untuk menyimpan solusi yang pernah ditemui selama iterasi berjalan sehingga banyak memakan memori [10] atau pergabungan menggunakan algoritma simulated annealing untuk mengambil individu terbaik yang akan digunakan pada iterasi berikutnya yang tidak memberikan pengaruh langsung terhadap nilai *fitness* [9]. Sedangkan pergabungan dengan logika fuzzy yaitu memberi nilai parameter pada proses *crossover* dan

mutasi untuk menghilangkan kesamaran pada perhitungan dalam algoritma genetika [13], hal ini tidak memakan banyak memori serta dapat memberi pengaruh langsung pada proses *crossover* dan mutasi. Penambahan teknik PROFIFEA dapat meningkatkan performa pada algoritma genetika dalam penelitian sebelumnya pada kasus TSP [2]. Maka penelitian ini akan menggunakan pergabungan antara algoritma genetika dan logika *fuzzy* dengan teknik PROFIFEA pada kasus yang berbeda yaitu penjadwalan mata kuliah.

Rangkuman penelitian terdahulu yang terkait dengan penjadwalan mata kuliah dapat dilihat dalam Tabel 2.1.

Tabel 2.1 Penelitian terkait

Peneliti	Tahun	Judul	Hasil
Andini Puspasari, Khusnul Novianingsih, Fitriani Agustina	2019	Penyelesaian Masalah Penjadwalan Perkuliahan Menggunakan Algoritma Genetika (Studi Kasus di Departemen Pendidikan Matematika FPMIPA Universitas Pendidikan Indonesia)	Parameter populasi sebesar 70, mutasi sebesar 0.4, generasi sebesar 50 mempunyai nilai rata-rata solusi optimal paling besar. Dengan total 120 slot.
Dwiprima Elvanny Myori, Hastuti	2018	Kombinasi Logika <i>Fuzzy</i> dan Algoritma Genetika untuk Masalah Penjadwalan Perkuliahan	Hasil 6 kromosom yang terdiri dari kode dosen, kode mata kuliah, kode waktu dan kode ruangan. Peran sistem inferensi <i>Fuzzy</i> pada algoritma genetika sangat membantu proses iterasi yang dilakukan.

Peneliti	Tahun	Judul	Hasil
Fitri Anggarsari, Wayan Firdaus Mahmudy, Candra Dewi	2017	Optimasi Kebutuhan Gizi untuk Balita Menggunakan Hybrid Algoritma Genetika dan Simulated Annealing.	Hasil dari pengujian yang dilakukan menghasilkan rata-rata nilai fitness sebesar 0.10106.
Wiktasari, E. S. Jatmiko	2016	Metode Simulated Annealing untuk Optimasi Penjadwalan Perguruan Tinggi.	Untuk hasil pengujian nilai alpha sebesar 0.0705 dan hasil pengujian nilai $T_0$ sebesar 0.07075. Sehingga dari analisa hasil pengujian yang dilakukan menghasilkan rata-rata nilai fitness sebesar 0.10106.
Henry Wulandari Pangestu, Kartika Yulianti, Rini Marwati	2016	Implementasi Algoritma <i>Fuzzy</i> Evolusi pada penjadwalan perkuliahan.	Jumlah populasi 100 dan jumlah generasi 200 dan dua buah keluaran yang diperoleh adalah probabilitas <i>crossover</i> dan probabilitas mutasi yang menyelesaikan permasalahan benturan.
Rusianah, M. Aziz Muslim, Sholeh Hadi Pramono.	2016	Implementasi Algoritma Genetik-Tabu Search dalam Optimasi Penjadwalan Perkuliahan.	<i>Fitness</i> = 1 artinya bahwa jadwal yang dihasilkan sudah memenuhi kriteria constraint.

Peneliti	Tahun	Judul	Hasil
Titus Kristanto, Tutuk Indriyani, Ni'matul Khoiroh.	2016	Penjadwalan ruang kuliah menggunakan Vertex Graph Coloring dan Simulated Annealing.	Hasil presentase keberhasilan hard constraint 98,47%, pelanggaran hard constraint 1,53%, hasil prosentase keberhasilan soft constraint 82,4%, dapat disimpulkan keberhasilan sistem yaitu 90,435%.
Setyoningsih Wibowo	2015	Penerapan Logika <i>Fuzzy</i> Dalam Penjadwalan Waktu Kuliah.	Nilai akurasi metode Mamdani sebesar 90.12%, dan metode Sugeno mempunyai nilai akurasi sebesar 70.63% dengan selisih nilai akurasi sebesar 19.60%.
Syaiful Muzid	2014	Dinamiasi Parameter Algoritma Genetika menggunakan Population Resizing on <i>Fitness</i> Improvement <i>Fuzzy</i> Evolutionary Algorithm.	Teknik PRoFIGA atau disebut dengan model PRoFIFEA mampu meningkatkan hasil akhir atau solusi yang ditemukan.

Peneliti	Tahun	Judul	Hasil
Yulia Yudihartanti	2011	Analisis Komparasi Metode Fuzzy Mamdani dan Sugeno dalam Penjadwalan Mata Kuliah.	Hasil menunjukkan tingkat akurasi dengan menggunakan metode mamdani sebesar 81,08% sedangkan metode sugeno 27.02%

## 2.2 Penjadwalan Mata Kuliah

Jadwal menurut KBBI adalah pembagian waktu berdasarkan rencana pengaturan urutan kerja atau rencana kegiatan dengan pembagian waktu pelaksanaan yang terperinci. Sedangkan penjadwalan adalah proses, cara, perbuatan menjadwalkan atau memasukkan dalam jadwal. Masalah penjadwalan mata kuliah pada perguruan tinggi, atau dikenal dengan istilah *University Course Timetabling Problem (UCTP)* [1]. Dalam proses penjadwalan terdapat suatu keadaan dimana dua kejadian tidak boleh menempati *slot* pada waktu yang sama [7]. Masalah jadwal yang sering ditemukan adalah dosen atau mahasiswa tidak dapat terlibat lebih dari satu pertemuan pada waktu dan ruang yang bersamaan. Banyak parameter yang harus diperhatikan dalam pembuatan jadwal yang meliputi dosen, ruang, waktu, dan mata kuliah yang diambil kembali. Jika parameter tersebut masih ada yang tidak sesuai dengan parameter lainnya maka jadwal tersebut tidak dapat dipakai. Jadwal yang layak dipakai adalah jadwal yang telah memenuhi segala aturan.

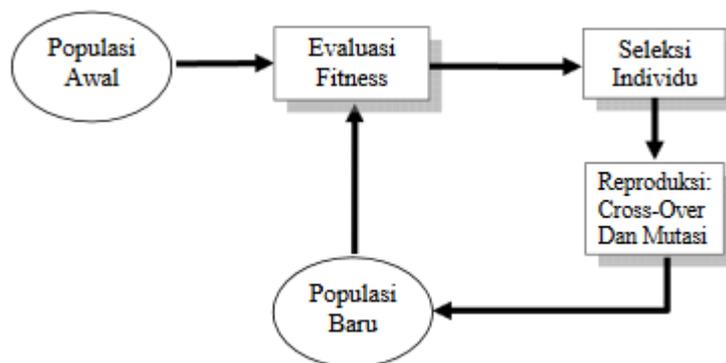
## 2.3 Algoritma Fuzzy Evolusi

Algoritma *Fuzzy* evolusi merupakan suatu teknik komputasi gabungan antara algoritma genetika dan logika *fuzzy* dimana parameter-parameter yang dipakai dalam algoritma genetika dihasilkan dari proses logika *fuzzy* [2]. Pada algoritma *fuzzy* evolusi, tahapan proses dalam menyelesaikan masalah hampir sama halnya seperti tahapan yang ada pada algoritma genetika. Namun untuk penentuan parameter-parameter genetika seperti nilai probabilitas *crossover* dan nilai probabilitas mutasi dihasilkan melalui logika *Fuzzy* karena pada proses seleksi

algoritma genetika mungkin dapat terjadi lokal optimum, maka untuk mengatasi masalah tersebut dilakukan penggabungan dengan logika *fuzzy*.

## 2.4 Algoritma Genetika

Algoritma genetika adalah model *soft computing* yang dikenalkan oleh John Holland dari Universitas Michigan pada tahun 1975, dimana algoritma genetika merupakan teknik pencarian heuristik berdasar mekanisme evolusi biologis yang meniru dari teori Darwin dan operasi genetika pada kromosom dan sering digunakan untuk menyelesaikan permasalahan optimasi [2]. Terdapat siklus pada pencarian algoritma genetika. Siklus akan terus berjalan mencari *fitness* terbaik dengan aturan – aturan yang telah ditentukan. Siklus dari algoritma genetika pertama kali dikenalkan oleh David Goldberg [15]. Ketika nilai *fitness* terbaik telah ditemukan, maka proses akan berhenti. Proses algoritma genetika dimulai dari pembangkitan populasi awal sampai pembentukan populasi baru. Untuk lebih jelasnya berikut adalah siklus algoritma genetika yang dapat dilihat pada Gambar 2.1 berikut.



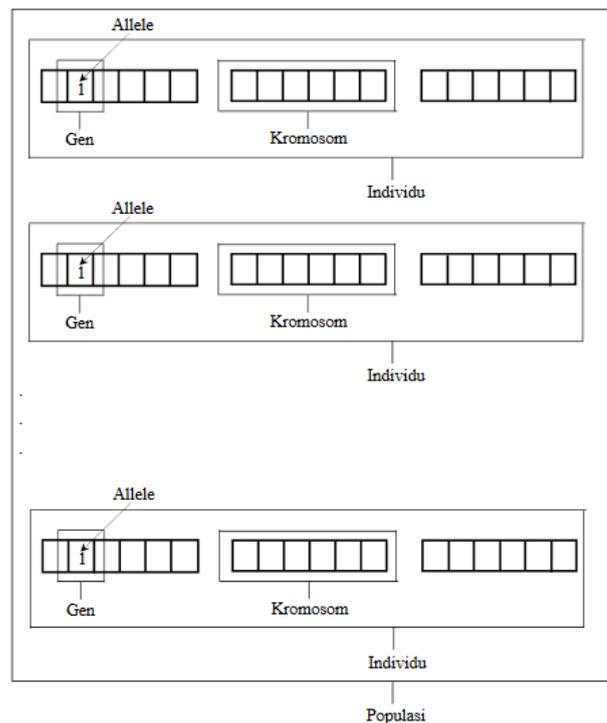
Gambar 2.1 Siklus algoritma genetika [16]

Pada Gambar 2.1 dijelaskan bahwa siklus dari algoritma genetika berawal dari proses pembangkitan populasi awal, evaluasi nilai *fitness*, seleksi individu, proses *crossover*, proses mutasi, dan pembangkitan populasi baru. Proses tersebut akan terus berjalan sampai menemukan nilai *fitness* yang terbaik. Pencarian *fitness* terbaik dilakukan pada evaluasi *fitness* yang berisi aturan – aturan yang telah ditentukan sesuai dengan masalah yang akan diselesaikan. Berikut adalah enam tahapan utama pada algoritma genetika, yaitu:

1. Representasi kromosom yaitu mempresentasikan masalah riil ke dalam bentuk terminologi biologi genetika.
2. Inisialisasi individu secara acak.
3. Fungsi evaluasi sebagai dasar proses seleksi.
4. Seleksi untuk memilih individu yang hendak diikuti dalam proses reproduksi.
5. *Crossover* menghasilkan keturunan dari dua buah kromosom induk yang terpilih.
6. Mutasi melakukan perubahan sebuah gen atau lebih dari sebuah individu.

### 2.4.1 Pengertian Individu

Individu merupakan suatu nilai yang menyatakan salah satu solusi dari kromosom[16]. Berikut adalah gambaran dari individu yang dapat dilihat pada gambar 2.2.



Gambar 2.2 Definisi Individu [16].

Berikut adalah penjelasan definisi pada individu :

1. Genotype (Gen) adalah sebuah nilai yang menyatakan satuan dasar yang membentuk suatu arti tertentu dalam satu kesatuan gen yang dinamakan kromosom.

2. Allele adalah nilai dari gen.
3. Kromosom adalah gabungan gen-gen yang membentuk nilai tertentu.
4. Individu adalah suatu nilai atau keadaan yang menyatakan salah satu solusi dari permasalahan yang diangkat.
5. Populasi adalah sekumpulan individu yang akan diproses bersamaan dalam satu siklus proses evolusi.
6. Generasi adalah satu siklus yang menyatakan proses evolusi atau satu iterasi dalam algoritma genetika.

#### **2.4.2 Representasi kromosom**

Representasi kromosom adalah bagaimana mengkodekan gen dari kromosom atau mengubah bilangan riil kedalam bentuk terminologi biologi genetika [16]. Satu gen mewakili satu variable. Gen dapat direpresentasikan dalam bentuk bit, bilangan real dan daftar aturan. Proses pengkodean nantinya akan berfungsi sebagai pembandingan antar kromosom pada sistem algoritma genetika.

Kromosom yang direpresentasikan dengan menggunakan beberapa bentuk adalah sebagai berikut;

1. String bit : 10111, 11011, dan seterusnya.
2. Array bilangan real : 65.65, 77.34, dan seterusnya.
3. Daftar aturan : R1, R2, dan seterusnya.

#### **2.4.3 Inisialisasi Individu**

Inisialisasi individu adalah proses membangkitkan sejumlah individu secara acak atau melalui prosedur tertentu [16]. Ukuran untuk populasi tergantung pada masalah yang akan diselesaikan dan jenis operator genetika yang akan diimplementasikan, kemudian dilakukan pembangkitan populasi awal. Syarat-syarat yang harus dipenuhi untuk menunjukkan suatu solusi harus benar-benar diperhatikan dalam pembangkitan setiap individu.

Random Generator adalah salah satu teknik yang digunakan dalam pembangkitan populasi awal [16]. Teknik ini melibatkan pembangkitan bilangan random untuk nilai setiap gen sesuai dengan representasi kromosom yang digunakan.

Untuk lebih jelasnya dapat dilihat pada Persamaan 2.1 berikut untuk penggunaan pembangkitan secara acak.

$$IPOP = \text{round} \{ \text{random}(N_{ipop}, N_{bits}) \} \quad (2.1)$$

*IPOP* adalah gen yang akan berisi pembulatan dari bilangan acak kemudian dibangkitkan sebanyak  $N_{ipop}$  (jumlah populasi) X  $N$  (Jumlah Gen dalam tiap kromosom). Pemangkitan akan diacak berdasarkan gen – gen yang telah ditentukan pada representasi kromosom.

#### 2.4.4 Fungsi evaluasi

Fungsi evaluasi adalah aturan untuk suatu permasalahan yang ditimbulkan antara mata kuliah, mata kuliah yang diambil kembali oleh mahasiswa, atau mata kuliah yang diambil disemester depan, dosen, ruang, dan waktu. fungsi objektif yang dapat digunakan agar memperoleh solusi adalah fungsi\_objektif = banyaknya aturan yang dilanggar antara satu kromosom dengan kromosom lainnya [16]. Nilai ini akan digunakan untuk menentukan nilai *fitness* pada setiap kromosom.

#### 2.4.5 Seleksi

Seleksi adalah proses memilih kromosom yang akan digunakan untuk proses *crossover* dan mutasi [16]. Kromosom yang dipilih adalah kromosom yang memiliki nilai *fitness* terbaik. Nilai *fitness* adalah nilai yang menyatakan baik atau tidaknya suatu individu. Nilai *fitness* yang dijadikan acuan dalam mencapai nilai optimal dalam algoritma genetika. Algoritma genetika bertujuan mencari individu dengan nilai *fitness* yang paling tinggi. Kemudian individu yang dipilih akan menuju pada proses selanjutnya.

Berikut adalah Persamaan 2.2 untuk menghitung nilai *fitness*.

$$Fitness = \frac{A}{F(X)+1} \quad (2.2)$$

Keterangan :

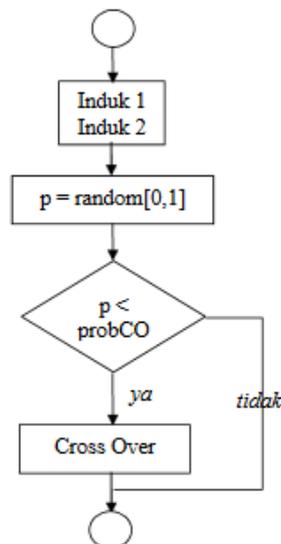
$A$  = Konstanta yang ditentukan.

$F(X)$  = Fungsi Objektif (kromosom<sub>*i*</sub>).

Dari persamaan dijelaskan bahwa baik atau buruknya kromosom akan dilihat pada banyaknya aturan yang dilanggar pada fungsi evaluasi. Proses seleksi dilakukan untuk mencari induk yang terbaik. Salah satu metode seleksi adalah mesin *roulette*. Metode ini merupakan metode yang sering dikenal dengan nama *stochastic sampling with replacement* yaitu menentukan individu yang akan dihitung nilai *fitness* secara acak [16]. Setelah tahap seleksi selesai, selanjutnya adalah tahap *crossover* dan mutasi. Pada algoritma genetika, tidak ada aturan baku untuk menentukan masing – masing parameter dalam tahapan tersebut, maka akan digabungkan dengan logika *fuzzy*. Dimana parameter-parameter pada tahapan *crossover* dan mutasi dalam algoritma genetika didapatkan dari sistem *Fuzzy* untuk menentukan tingkat probabilitas *crossover* dan probabilitas mutasi.

#### 2.4.6 Crossover

*Crossover* adalah operator dari algoritma genetika yang melibatkan dua induk yang ditukar untuk membentuk kromosom yang baru. Individu dipilih secara acak[16]. Prinsip *crossover* adalah melakukan operasi pertukaran pada gen-gen yang bersesuaian dari dua induk yang berbeda untuk menghasilkan individu yang baru. Berikut adalah diagram alir proses *crossover* yang dapat dilihat pada Gambar 2.3.



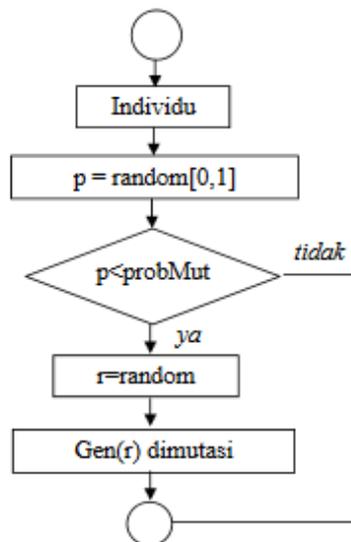
Gambar 2.3 Diagram alir proses *crossover* [16].

Dilihat pada Gambar 2.3 menjelaskan bahwa proses *crossover* dilakukan pada dua induk yang berbeda. Lalu dilakukan pembangkitan bilangan acak yang akan

dibandingkan dengan nilai probabilitas *crossover* dan melakukan proses pertukaran. Kromosom yang terpilih akan ditukar dan menjadi kromosom baru. Kromosom baru nantinya akan dipakai pada tahapan berikutnya.

#### 2.4.7 Mutasi

Mutasi berperan untuk menggantikan gen yang hilang dari populasi akibat proses seleksi yang memunculkan kembali gen yang tidak ada pada inisialisasi populasi [16]. Hasil Kromosom pada langkah sebelumnya dimutasi dengan menambahkan nilai random dengan probabilitas yang akan ditentukan menggunakan logika *Fuzzy*. Kromosom yang terpilih akan melakukan proses mutasi dengan mengganti gen secara acak. Untuk lebih jelasnya berikut adalah diagram alir proses mutasi yang dapat dilihat pada Gambar 2.4.



Gambar 2.4 Diagram alir proses mutasi [16].

Gambar 2.4 menjelaskan bahwa proses mutasi adalah proses mengganti gen yang dilakukan sesuai dengan probabilitas mutasi yang telah ditentukan. posisi gen yang terpilih akan dilakukan mutasi. Proses mutasi pada gen akan dilakukan secara acak. Setelah proses mutasi selesai maka akan dilakukan pembentukan populasi baru.

#### 2.5 Penentuan Parameter menggunakan Logika *Fuzzy*

Pada algoritma tidak ada aturan baku mengenai jumlah populasi dan jumlah generasi yang menyebabkan perhitungan pada algoritma genetika dilakukan dalam

beberapa kali percobaan hingga mendapatkan hasil yang diharapkan. Untuk mengatasi permasalahan tersebut, dibutuhkan suatu sistem yang dapat menghilangkan kesamaran dalam penentuan parameter untuk nilai probabilitas *crossover* dan mutasi. Sistem yang dapat digunakan adalah logika *fuzzy*. Logika *fuzzy* adalah algoritma yang digunakan untuk menghilangkan keambiguan. Logika *fuzzy* menyediakan suatu cara untuk merubah pernyataan linguistik menjadi suatu numerik [20]. Metode *fuzzy* yang digunakan adalah sistem inferensi *fuzzy* mamdani karena sistem ini dapat menghasilkan nilai untuk probabilitas yang lebih mendekati pasti dibandingkan logika *fuzzy* tsukamoto dan sugeno [19].

Berikut adalah 4 tahapan yang ada pada sistem inferensi *fuzzy* mamdani [20], yaitu :

1. Pembentukan himpunan *fuzzy* (*fuzzyfication*).

Pada metode mamdani baik variabel input maupun variabel output dibagi menjadi satu atau lebih himpunan fuzzy.

2. Fungsi implikasi.

Pada Metode Mamdani, fungsi implikasi yang digunakan adalah min.

3. Komposisi aturan.

Metode yang digunakan dalam melakukan inferensi sistem fuzzy, yaitu metode Max (Maximum). Secara umum dapat dituliskan pada persamaan 2.3 berikut :

$$\mu_{sf}[X_i] = \max (\mu_{sf} [X_i], \mu_{kf} [X_i]) \quad (2.3)$$

Keterangan :

- $\mu_{sf} [X_i]$  = nilai keanggotaan solusi fuzzy sampai aturan ke i
- $\mu_{kf} [X_i]$  = nilai keanggotaan konsekuan fuzzy aturan ke i

4. Penegasan (*defuzzy*)

*Defuzzyfikasi* pada komposisi aturan mamdani dengan menggunakan metode *centroid*. Dimana pada metode ini, solusi *crisp* diperoleh dengan cara mengambil titik pusat daerah fuzzy. Metode penegasan *center of average* atau *centroid* merupakan metode yang digunakan dalam sistem *fuzzy* dan kontrol *fuzzy* karena secara komputasi, metode ini lebih mudah dan masuk akal [20]. Secara umum dapat dituliskan pada persamaan 2.4 berikut :

$$X = \frac{\int \mu(x).x dx}{\int \mu(x) dx} \quad (2.4)$$

Keterangan :

- $X$  = nilai hasil *defuzzyfikasi* atau titik pusat daerah *fuzzy*
- $\mu(x)$  = nilai keanggotaan
- $\int \mu(x).x dx$  = momen untuk semua daerah hasil komposisi aturan

Dengan  $X$  menyatakan nilai hasil *defuzzifikasi* /titik pusat daerah *fuzzy*.  $\mu(x)$  menyatakan nilai keanggotaan, dan  $\int \mu(x).x dx$  menyatakan momen untuk semua daerah hasil komposisi aturan. Nilai dari titik pusat merupakan nilai hasil dari proses *defuzzifikasi*, nilai ini merupakan hasil dari keputusan akhir yang nantinya akan digunakan pada tahap algoritma genetika yaitu pada tahap *crossover* dan tahap mutasi. Nilai tersebut akan digunakan untuk nilai probabilitas *crossover* dan probabilitas mutasi yang menjadi aturan baku pada proses selanjutnya.

## 2.5 Population Resizing on Fitness Improvement Fuzzy Evolutionary Algorithm

PROFIFEA adalah pengembangan dari Algoritma *Fuzzy* Evolusi (*Fuzzy* Evoluitonary Algorithm) yang digabungkan dengan model Population Resizing on *Fitness* Improvement Genetic Algorithm (PROFIGA)[2]. Pada tahap terakhir proses pembentukan populasi baru dapat memungkinkan terjadinya kesamaan nilai *fitness* pada populasi yang lama. Jika ukuran populasi terlalu kecil akan memungkinkan terjadinya konvergensi dini, dan jika terlalu besar akan mengakibatkan lamanya waktu yang dibutuhkan algoritma genetika dalam menghasilkan solusi terbaik[2]. Untuk meningkatkan nilai *fitness* pada algoritma genetika dalam pembentukan populasi baru dapat menggunakan PROFIFEA sebagai aturan dalam menentukan populasi baru yang akan dipilih. PROFIFEA menggunakan aturan-aturan kondisi untuk penentuan ukuran populasi baru yang digabungkan dengan PROFIGA. Berikut adalah tiga aturan PROFIGA yang dapat dilihat pada persamaan sebagai berikut:

1. Nilai *fitness* terbaik meningkat

Jika nilai *fitness* terbaik pada generasi saat ini meningkat maka ukuran populasi

yang baru bertambah dengan Persamaan 2.5 berikut:

$$newPopSize = newPopSize + \left( oldPopSize * \left( IncFac * (maxG - curG) * \left( \frac{newMax - oldMaxf}{initMaxf} \right) \right) \right) \quad (2.5)$$

Keterangan :

- a. newPopSize adalah ukuran populasi baru
- b. oldPopSize adalah ukuran populasi lama
- c. IncFac adalah persentase faktor penambahan
- d. maxG adalah jumlah maksimum generasi
- e. curG adalah generasi saat ini
- f. newMaxF adalah nilai *fitness* maksimum pada generasi saat ini
- g. oldMaxF adalah nilai *fitness* maksimum pada generasi sebelumnya
- h. initMaxF adalah nilai *fitness* maksimum yang diharapkan

## 2. Nilai *fitness* terbaik menurun

Jika nilai *fitness* terbaik pada generasi saat ini menurun maka ukuran populasi yang baru menyusut dengan Persamaan 2.6 berikut:

$$newPopSize = oldPopSize + |1 - DecFac| \quad (2.6)$$

Keterangan :

- a. newPopSize adalah ukuran populasi baru
- b. oldPopSize adalah ukuran populasi lama
- c. DecFac adalah persentase faktor penyusutan

## 3. Nilai *fitness* terbaik tetap sama

- a. Nilai *fitness* terbaik tetap sama selama beberapa generasi dan masih dibawah batas maksimal generasi tidak ada perubahan maka ukuran populasi yang baru menyusut seperti pada persamaan 2. Contohnya batas maksimal generasi tidak ada perubahan adalah 5 generasi, dan nilai *fitness* terbaik sama selama 2 sampai 4 generasi maka ukuran populasi yang baru menyusut.
- b. Jika nilai *fitness* terbaik pada generasi saat ini sama dan sama dengan batas maksimal generasi tidak ada perubahan maka ukuran populasi yang baru bertambah dengan Persamaan 2.7 sebagai berikut:

$$newPopSize = oldPopSize + |oldPopSize * IncFac| \quad (2.7)$$

Keterangan :

- a. newPopSize adalah ukuran populasi baru
- b. oldPopSize adalah ukuran populasi lama
- c. IncFac adalah persentase faktor penambahan

Dalam pemanfaatan algoritma genetika tingkat keberagaman individu kromosom dalam populasi harus dipertahankan dengan baik, sehingga tidak terjadi homogenitas individu. Karena hal ini dapat menyebabkan terjadinya konvergensi dini. Menurut Muzid dan Wardoyo [2], Cara atau metode yang digunakan dalam PRoFIFEA untuk menghasilkan solusi yang lebih optimum dan mencegah homogenitas dari individu atau kromosom.