

BAB III

METODE PENELITIAN

Bab ini akan membahas metode yang akan digunakan dalam implementasi Algoritma Genetika untuk menyelesaikan masalah penjadwalan dokter dan perawat di IGD. Pembahasan ini mencakup deskripsi masalah, tahapan penelitian, dan contoh kasus serta penyelesaiannya.

3.1 Deskripsi Masalah

Penelitian ini meneliti mengenai masalah penjadwalan dokter dan perawat Instalasi Gawat Darurat (IGD) di suatu rumah sakit umum daerah yang berada di kota Bandung. Saat ini, sistem penjadwalan di IGD masih dilakukan secara manual, sehingga memakan waktu yang lama dan menyulitkan penanggung jawab dalam membagi waktu antara menjaga *shift* dan membuat jadwal. Tujuan penelitian ini adalah untuk membangun sistem penjadwalan yang optimal, yaitu suatu sistem penjadwalan yang mampu memenuhi *hard constraints* dan mengupayakan *soft constraints*. *Hard constraints* merupakan peraturan yang tidak dapat dilanggar seperti batas minimal jam kerja dokter dan perawat, banyaknya dokter dan perawat yang bekerja pada setiap *shift* sesuai dengan banyaknya yang tersedia, dan larangan bagi dokter dan perawat untuk bertugas lebih dari satu *shift* dalam satu hari. Sementara itu, *soft constraints* merupakan kendala tambahan berupa batasan peraturan yang dapat dilanggar seminimal mungkin seperti batas maksimal kebutuhan dokter dan perawat bekerja di setiap *shift* kerja, serta ketentuan cuti dan hari libur dokter dan perawat selama satu periode. Algoritma Genetika akan diimplementasikan dalam masalah penjadwalan tersebut. Jadwal yang dihasilkan diharapkan dapat mempercepat proses penyusunan jadwal dan menghasilkan jadwal yang optimal

3.2 Tahapan Penelitian

Penelitian ini terdiri dari beberapa tahapan sebagai berikut:

1. **Studi Pustaka**
Studi pustaka dilaksanakan dengan cara menggali teori-teori terkait implementasi metode Algoritma Genetika dalam konteks bidang kesehatan melalui referensi buku atau artikel jurnal.
2. **Pengumpulan Data**
Data yang diperlukan dalam penelitian ini adalah data yang berhubungan dengan penjadwalan *shift* kerja dokter dan perawat di IGD.
3. **Pembangunan Model Optimasi**
Pada tahapan ini akan dibangun model optimasi masalah penjadwalan dokter dan perawat di IGD berdasarkan asumsi-asumsi.
4. **Penyelesaian Masalah**
Pada tahapan ini akan dibangun model masalah optimasi penjadwalan dokter dan perawat di IGD menggunakan Algoritma Genetika.
5. **Validasi**
Model dan metode penyelesaian akan divalidasi dengan memeriksa perbandingan antara perhitungan manual dan komputasi pada contoh kasus. Apabila hasilnya belum valid, maka tahapan akan kembali ke proses pemodelan. Apabila hasilnya valid, maka proses selanjutnya akan dilanjutkan ke tahapan implementasi.
6. **Implementasi**
Model optimasi dan teknik penyelesaian yang telah valid selanjutnya akan diimplementasikan untuk menyelesaikan masalah penjadwalan dokter dan perawat di IGD salah satu Rumah Sakit Umum Daerah di Kota Bandung menggunakan Algoritma Genetika.
7. **Penarikan Kesimpulan**
Dalam menyimpulkan, peneliti akan mengkompilasi hasil dari penelitian yang telah dilaksanakan. Hasil dari penelitian ini mencakup sejauh mana ketepatan dengan rumusan masalah yang diajukan serta implementasi dari optimasi penjadwalan untuk dokter dan perawat di IGD. Semua temuan ini didasarkan pada data yang telah dikumpulkan dan dianalisis selama proses penelitian.

3.3 Data Penelitian

Data yang diperlukan dalam penelitian ini adalah data yang berhubungan dengan penjadwalan *shift* kerja dokter dan perawat di IGD, data-data tersebut terdiri dari data dokter dan perawat yang bertugas di IGD, data *shift* kerja yaitu batasan jam kerja tiap *shift*, dan data aturan kerja yang harus diterapkan dalam membuat jadwal *shift* kerja di IGD.

3.4 Model Optimasi

Model optimasi masalah penjadwalan dokter dan perawat di IGD dibangun berdasarkan asumsi-asumsi berikut:

1. Banyaknya dokter dan perawat yang tersedia mencukupi untuk melibatkan setiap *shift* kerja di IGD.
2. Tidak ada dokter dan perawat yang baru bergabung dan berhenti bekerja, sehingga banyak dokter dan perawat tetap selama periode penjadwalan.
3. Tidak ada dokter dan perawat yang izin atau sakit selama periode penjadwalan.
4. Penjadwalan dilakukan untuk satu bulan kerja (30 hari).

Sebelum membangun model optimasi, terlebih dahulu didefinisikan himpunan dan parameter model. Himpunan dan parameter yang akan digunakan dalam pemodelan ini adalah sebagai berikut:

Tabel 3.1 Parameter dan Indeks pada Model

Parameter dan Indeks	Keterangan
D	himpunan semua dokter yang bertugas di IGD
d	indeks dokter, $d = 1, 2, 3, \dots, u, \forall d \in D$
N	himpunan perawat yang bertugas di IGD
n	indeks perawat, $n = 1, 2, 3, \dots, v, \forall n \in N$
H	Himpunan hari dalam periode penjadwalan, dengan indeks $h(i)$, dengan $i = 1, 2, 3, \dots, w$
S	Himpunan <i>shift</i>

s	indeks <i>shift</i> , $S = \{p, r, m\}$ di mana: $p =$ pagi, $r =$ sore, $m =$ malam
A	himpunan hari libur dokter
B	himpunan hari libur perawat
p_d	banyaknya dokter yang bertugas pada <i>shift</i> pagi
r_d	banyaknya dokter yang bertugas pada <i>shift</i> sore
m_d	banyaknya dokter yang bertugas pada <i>shift</i> malam
p_n	banyaknya perawat yang bertugas pada <i>shift</i> pagi
r_n	banyaknya perawat yang bertugas pada <i>shift</i> sore
m_n	banyaknya perawat yang bertugas pada <i>shift</i> malam
c_{\min_d}	minimal <i>shift</i> yang harus dipenuhi oleh setiap dokter dalam satu bulan
c_{\max_d}	maksimal <i>shift</i> yang bisa dipenuhi oleh setiap dokter dalam satu bulan
c_{\min_n}	minimal <i>shift</i> yang harus dipenuhi oleh setiap perawat dalam satu bulan
c_{\max_n}	maksimal <i>shift</i> yang bisa dipenuhi oleh setiap perawat dalam satu bulan

Variabel keputusan model didefinisikan untuk menentukan penugasan dokter dan perawat pada *shift* untuk setiap harinya. Berikut adalah variabel keputusan model:

$$X_{d,h,s} = \begin{cases} 1, & \text{jika dokter } d \in D \text{ bertugas pada hari } h \in H \text{ dan } \textit{shift} s \in S, \\ 0, & \text{yang lainnya.} \end{cases}$$

$$X_{d,h,A} = \begin{cases} 1, & \text{jika dokter } d \in D \text{ mendapatkan libur pada hari } h \in H, \\ 0, & \text{yang lainnya.} \end{cases}$$

$$Y_{n,h,s} = \begin{cases} 1, & \text{jika perawat } n \in N \text{ bertugas pada hari } h \in H \text{ dan } \textit{shift} s \in S, \\ 0, & \text{yang lainnya.} \end{cases}$$

$$Y_{n,h,B} = \begin{cases} 1, & \text{jika perawat } n \in N \text{ mendapatkan libur pada hari } h \in H, \\ 0, & \text{yang lainnya.} \end{cases}$$

Selanjutnya menentukan *hard constraints* atau kendala-kendala utama yang tidak boleh dilanggar dari fungsi tujuan pada model optimasi. Adapun *hard constraints* adalah sebagai berikut:

1. Kebutuhan akan dokter dan perawat yang bertugas di setiap *shift* terpenuhi
 - a. Kebutuhan dokter di *shift* pagi

$$\sum_d^u X_{d,h,p} = p_d, \forall d \in D, h \in H$$

- b. Kebutuhan dokter di *shift* sore

$$\sum_d^u X_{d,h,r} = r_d, \forall d \in D, h \in H$$

- c. Kebutuhan dokter di *shift* malam

$$\sum_d^u X_{d,h,m} = m_d, \forall d \in D, h \in H$$

- d. Kebutuhan perawat di *shift* pagi

$$\sum_n^v Y_{n,h,p} = p_n, \forall n \in N, h \in H$$

- e. Kebutuhan perawat di *shift* sore

$$\sum_n^v Y_{n,h,r} = r_n, \forall n \in N, h \in H$$

- f. Kebutuhan perawat di *shift* malam

$$\sum_n^v Y_{n,h,m} = m_n, \forall n \in N, h \in H$$

2. Setiap dokter dan perawat hanya mendapat satu *shift* setiap hari (pagi, sore, malam) atau libur dalam satu periode

- a. Dokter

$$\sum_{s \in S} X_{d,h,s} + X_{d,h,A} = 1, \forall d \in D, h \in H$$

- b. Perawat

$$\sum_{s \in S} Y_{n,h,s} + Y_{n,h,B} = 1, \forall n \in N, h \in H$$

3. Dokter dan perawat yang bertugas pada *shift* malam, tidak diperbolehkan mendapat *shift* pagi di hari berikutnya

- a. Dokter

$$X_{d,h(i),m} + X_{d,h(i+1),p} \leq 1, \forall d \in D, m, p \in S$$

b. Perawat

$$Y_{n,h(i),m} + Y_{n,h(i+1),p} \leq 1, \forall n \in N, m, p \in S$$

4. Dokter dan perawat tidak bertugas pada *shift* malam lebih dari 2 hari berturut-turut

a. Dokter

$$X_{d,h(i),m} + X_{d,h(i+1),m} + X_{d,h(i+2),m} \leq 2, \forall d \in D, m \in S$$

b. Perawat

$$Y_{n,h(i),m} + Y_{n,h(i+1),m} + Y_{n,h(i+2),m} \leq 2, \forall n \in N, m \in S$$

5. Total *shift* dokter dan perawat tidak kurang dari ketentuan minimal atau melebihi ketentuan maksimal dalam satu periode penjadwalan:

a. Dokter

$$c_{\min_d} \leq \sum_{h \in H} \sum_{s \in S} X_{d,h,s} \leq c_{\max_d}, \forall d \in D$$

b. Perawat

$$c_{\min_n} \leq \sum_{h \in H} \sum_{s \in S} Y_{n,h,s} \leq c_{\max_n}, \forall n \in N$$

Setelah menentukan *hard constraints*, akan ditentukan *soft constraints* atau kendala-kendala tambahan yang masih dapat dilanggar tetapi diharapkan dipenuhi semaksimal mungkin. Kendala tambahan ini adalah sebagai berikut:

1. Setiap dokter dan perawat tidak ditugaskan pada waktu *shift* pagi dan sore lebih dari 3 hari berturut-turut

a. Dokter

$$\min T_1 = \sum_{d=1}^u \sum_{h=1}^{w-3} (X_{d,h(i),p} + X_{d,h(i+1),p} + X_{d,h(i+2),p} + X_{d,h(i+3),p})$$

$$\min T_2 = \sum_{d=1}^u \sum_{h=1}^{w-3} (X_{d,h(i),r} + X_{d,h(i+1),r} + X_{d,h(i+2),r} + X_{d,h(i+3),r})$$

b. Perawat

$$\min G_1 = \sum_{n=1}^v \sum_{h=1}^{w-3} (Y_{n,h(i),p} + Y_{n,h(i+1),p} + Y_{n,h(i+2),p} + Y_{n,h(i+3),p})$$

$$\min G_2 = \sum_{n=1}^v \sum_{h=1}^{w-3} (Y_{n,h(i),r} + Y_{n,h(i+1),r} + Y_{n,h(i+2),r} + Y_{n,h(i+3),r})$$

2. Setiap dokter dan perawat diusahakan untuk mendapatkan alokasi antara *shift*-nya seimbang atau meminimalkan ketidakseimbangan antara *shift*.

a. Dokter

$$\min T_3 = \sum_{d \in D} \left(\left| \frac{\sum_{h \in H} \sum_{s \in S} X_{d,h,s}}{w - \sum_{h \in H} X_{d,h,A}} - \frac{1}{3} \right| \right)$$

b. Perawat

$$\min G_3 = \sum_{n \in N} \left(\left| \frac{\sum_{h \in H} \sum_{s \in S} Y_{n,h,s}}{w - \sum_{h \in H} Y_{n,h,B}} - \frac{1}{3} \right| \right)$$

Fungsi tujuan model optimasi masalah penjadwalan dokter dan perawat di IGD diekspresikan sebagai berikut:

1. Dokter

Meminimumkan Pelanggaran:

$$z_1 = T_1 + T_2 + T_3 \quad (3.1)$$

Terhadap:

$$\sum_d^u X_{d,h,p} = p_d, \forall d \in D, h \in H \quad (3.2)$$

$$\sum_d^u X_{d,h,r} = r_d, \forall d \in D, h \in H \quad (3.3)$$

$$\sum_d^u X_{d,h,m} = m_d, \forall d \in D, h \in H \quad (3.4)$$

$$\sum_{s \in S} X_{d,h,s} + X_{d,h,A} = 1, \forall d \in D, h \in H \quad (3.5)$$

$$X_{d,h(i),m} + X_{d,h(i+1),p} \leq 1, \forall d \in D, m, p \in S \quad (3.6)$$

$$X_{d,h(i),m} + X_{d,h(i+1),m} + X_{d,h(i+2),m} \leq 2, \forall d \in D, m \in S \quad (3.7)$$

$$c_{\min_d} \leq \sum_{h \in H} \sum_{s \in S} X_{d,h,s} \leq c_{\max_d}, \forall d \in D \quad (3.8)$$

$$X_{d,h,s}, X_{d,h,A} \in \{0,1\}, \forall d \in D, h \in H \quad (3.9)$$

2. Perawat

Meminimumkan Pelanggaran:

$$z_2 = G_1 + G_2 + G_3 \quad (3.10)$$

Terhadap:

$$\sum_n^v Y_{n,h,p} = p_n, \forall n \in N, h \in H \quad (3.11)$$

$$\sum_n^v Y_{n,h,r} = r_n, \forall n \in N, h \in H \quad (3.12)$$

$$\sum_n^v Y_{n,h,m} = m_n, \forall n \in N, h \in H \quad (3.13)$$

$$\sum_{s \in S} Y_{n,h,s} + Y_{n,h,B} = 1, \forall n \in N, h \in H \quad (3.14)$$

$$Y_{n,h(i),m} + Y_{n,h(i+1),p} \leq 1, \forall n \in N, m, p \in S \quad (3.15)$$

$$Y_{n,h(i),m} + Y_{n,h(i+1),m} + Y_{n,h(i+2),m} \leq 2, \forall n \in N, m \in S \quad (3.16)$$

$$c_{\min_n} \leq \sum_{h \in H} \sum_{s \in S} Y_{n,h,s} \leq c_{\max_n}, \forall n \in N \quad (3.17)$$

$$Y_{n,h,s}, Y_{n,h,B} \in \{0,1\}, \forall n \in N, h \in H \quad (3.18)$$

Model optimasi yang digunakan dalam penelitian ini termasuk ke dalam kategori model *binary integer programming*.

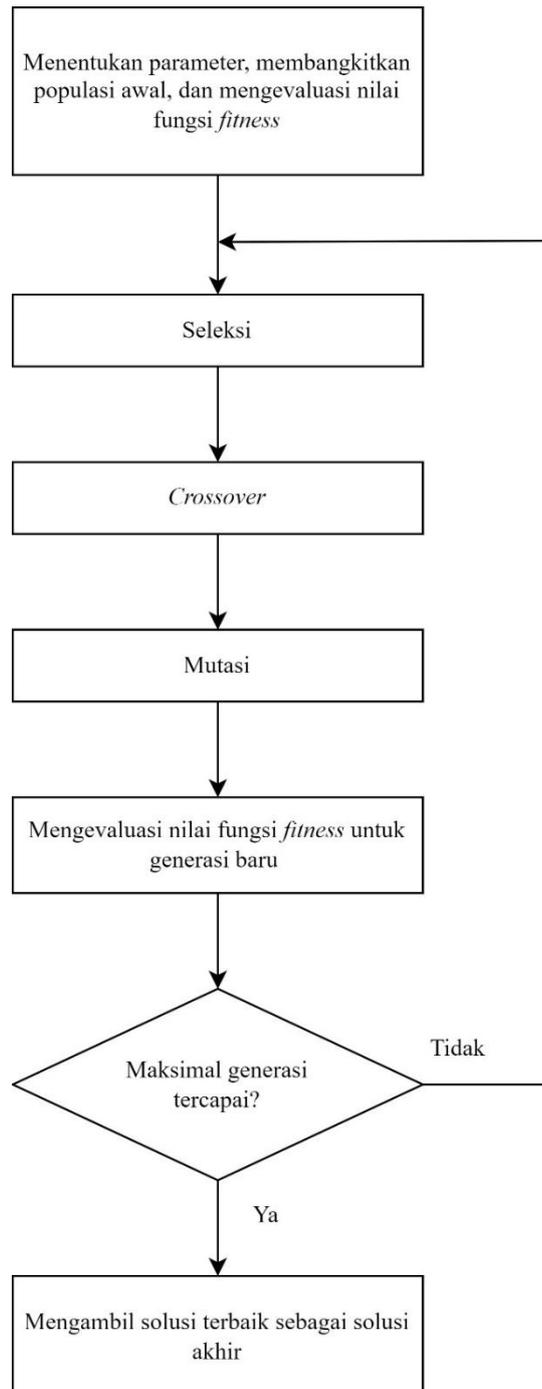
3.5 Penyelesaian Model dengan Algoritma Genetika pada Masalah

Penjadwalan Dokter dan Perawat di IGD

Algoritma Genetika digunakan untuk menyelesaikan model optimasi masalah penjadwalan dokter dan perawat di IGD. Dalam Algoritma Genetika, solusi dari permasalahan direpresentasikan sebagai kromosom. Setiap kromosom ini tersusun dari rangkaian gen. Proses Algoritma Genetika dimulai dengan pembangkitan acak sejumlah kromosom yang membentuk suatu populasi awal. Kromosom-kromosom ini mewakili individu-individu dalam suatu populasi dan dianalogikan sebagai himpunan-himpunan solusi yang mungkin dari suatu permasalahan optimasi. Setelah itu, himpunan-himpunan solusi dievaluasi menggunakan nilai *fitness*. Setiap himpunan solusi memiliki nilai *fitness* terhadap

tujuan yang ingin dicapai. Semakin tinggi nilai *fitness*-nya, semakin besar peluang himpunan solusi tersebut untuk mendekati solusi optimal. Himpunan solusi yang memiliki nilai *fitness* rendah akan dihapus dan diganti dengan himpunan solusi baru. Himpunan solusi baru tersebut dibentuk berdasarkan informasi genetik yang bermanfaat dari himpunan solusi sebelumnya yang memiliki nilai *fitness* tinggi.

Individu dengan pertahanan yang tinggi dapat bereproduksi dengan individu lain dalam populasinya. Individu baru yang dihasilkan disebut keturunan, yang membawa beberapa sifat dari induknya. Individu dalam populasi yang tidak dapat bereproduksi akan dieliminasi. Dengan cara ini, populasi akan berkembang menjadi generasi-generasi baru dengan karakteristik yang semakin baik. Semakin banyak individu yang dikawinkan, semakin besar kemungkinan untuk mendapatkan individu terbaik. Seleksi alam genetika diharapkan menghasilkan kromosom baru dengan nilai *fitness* lebih rendah dari generasi sebelumnya. Setelah beberapa generasi, akan didapatkan kromosom terbaik yang merupakan solusi optimal. Berikut adalah tahapan-tahapan dari Algoritma Genetika:



Gambar 3.1 Diagram Alir Algoritma Genetika

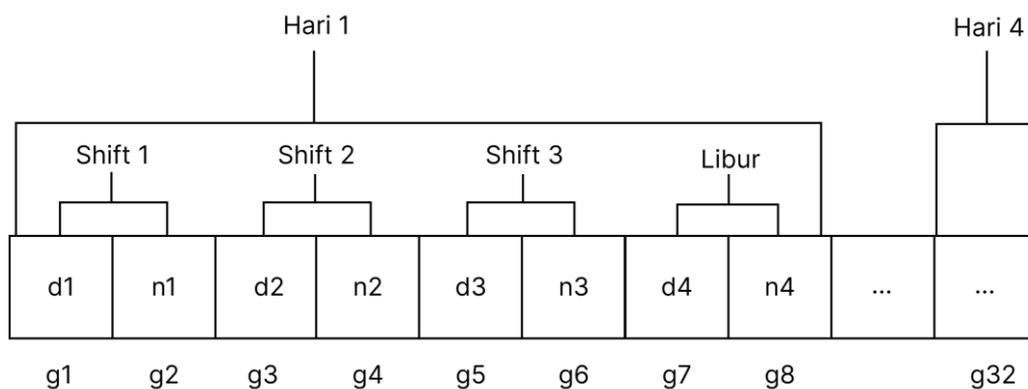
1. Representasi Kromosom

Dalam Algoritma Genetika, kromosom mewakili solusi permasalahan. Satu kromosom terdiri atas n gen, n menyatakan hasil perkalian antara banyaknya dokter dan perawat dengan banyaknya hari penjadwalan. Dalam penelitian ini, kromosom direpresentasikan dalam bentuk elemen bilangan bulat beserta simbolnya. Satu gen

merepresentasikan individu (dokter dan perawat) yang bertugas pada *shift* tertentu di hari tertentu.

Misalkan periode penjadwalan adalah 7 hari dengan banyaknya 4 dokter dan 4 perawat. Dalam satu hari terdapat tiga *shift*: pagi, sore, dan malam. Dalam satu *shift*, terdapat satu dokter dan satu perawat yang bertugas, sementara mereka yang tidak bertugas dianggap sedang libur. Satu kromosom dalam konteks ini mewakili jadwal *shift* untuk dokter dan perawat serta hari libur mereka.

Gen untuk dokter dimisalkan dengan d_1, d_2, \dots, d_j , dengan j merupakan banyak dokter, sedangkan gen untuk perawat dimisalkan dengan n_1, n_2, \dots, n_q , dengan q merupakan banyak perawat. Dengan adanya 4 dokter dan 4 perawat dalam periode 4 hari kerja, maka setiap kromosom terdiri dari $4 \times (4 + 4) = 32$ gen. Representasi kromosom dalam contoh kasus ini dapat dilihat pada Gambar 3.2.



Gambar 3.2 Representasi Kromosom Contoh Kasus

Pada Gambar 3.2, setiap gen dalam kromosom berisi nilai yang merepresentasikan individu (dokter atau perawat) yang bertugas pada *shift* tertentu di hari tertentu. Misalnya, gen pertama d_1 menunjukkan bahwa Dokter 1 bertugas pada *shift* 1 di hari pertama dalam periode penjadwalan tersebut. Gen berikutnya n_1 , menunjukkan bahwa Perawat 1 juga bertugas pada *shift* yang sama di hari yang sama, begitu pula dengan gen-gen selanjutnya. Setelah jadwal *shift* untuk dokter dan perawat direpresentasikan dalam satu hari tertentu, selanjutnya direpresentasikan juga hari libur mereka. Jika seorang dokter atau perawat tidak bertugas pada *shift* tertentu di hari tertentu, maka ia akan mendapatkan hari libur pada hari tersebut.

2. Menentukan Nilai Parameter – Parameter Algoritma Genetika

Sebelum menjalankan Algoritma Genetika, perlu ditetapkan terlebih dahulu nilai parameter – parameter yang akan digunakan. Parameter – parameter tersebut meliputi:

a. Ukuran populasi (*pop size*)

Parameter ini mengatur banyaknya kromosom yang akan diproses dalam setiap iterasi

b. Banyaknya generasi (iterasi)

Parameter ini menentukan banyaknya siklus atau iterasi yang akan dilakukan oleh Algoritma Genetika. Setiap siklus terdiri dari tahap seleksi, *crossover*, dan mutasi.

c. Ukuran turnamen

Parameter ini digunakan dalam metode seleksi turnamen. Ukuran turnamen menentukan banyaknya individu yang akan dipilih secara acak untuk berpartisipasi dalam setiap turnamen seleksi. Kromosom terbaik dari turnamen ini akan dipilih untuk reproduksi.

d. Probabilitas *crossover* (P_c)

P_c dinyatakan sebagai bilangan riil antara 0 dan 1. Parameter ini menentukan kemungkinan terjadinya *crossover* pada suatu kromosom.

e. Probabilitas mutasi (P_m)

Seperti P_c , P_m juga dinyatakan dengan bilangan riil antara 0 dan 1. Parameter ini mengatur kemungkinan terjadinya mutasi pada suatu kromosom.

Misalkan dalam contoh kasus ini, nilai-nilai parameter Algoritma Genetika ditetapkan sebagai berikut: ukuran populasi 4, banyaknya generasi 1, ukuran turnamen 3, P_c 0,7, dan P_m 0,6.

3. Membangkitkan Populasi Awal

Proses membangkitkan populasi awal dilakukan secara acak. Populasi tersebut berisi beberapa kromosom yang telah didefinisikan sebelumnya. Banyaknya populasi awal yang dibangkitkan ditentukan oleh parameter ukuran

populasi (*pop size*). Masing-masing kromosom memiliki panjang 32 gen. Pembangkitan kromosom untuk populasi awal dapat dilihat pada Gambar 3.3.

Kromosom 1	d4	n3	d3	...	n3	d2	n4	...	n3	d3	n1
	g1	g2	g3	...	g16	g17	g18	...	g30	g31	g32
Kromosom 2	d2	n4	d3	...	n3	d2	n2	...	n4	d1	n1
	g1	g2	g3	...	g16	g17	g18	...	g30	g31	g32
Kromosom 3	d1	n2	d4	...	n4	d2	n2	...	n2	d4	n4
	g1	g2	g3	...	g16	g17	g18	...	g30	g31	g32
Kromosom 4	d1	n3	d4	...	n2	d2	n2	...	n4	d4	n2
	g1	g2	g3	...	g16	g17	g18	...	g30	g31	g32

Gambar 3.3 Populasi Awal Contoh Kasus

4. Menghitung Nilai *Fitness*

Kualitas suatu individu dalam proses optimasi ditentukan oleh nilai *fitness* yang dihasilkannya. Nilai *fitness* yang dihasilkan dari suatu individu menunjukkan seberapa optimal solusi yang diperoleh. Pada masalah optimasi penjadwalan dokter dan perawat di IGD, perhitungan *fitness* dilakukan berdasarkan batasan-batasan (*constraints*) yang telah ditetapkan sebelumnya. Semakin sedikit batasan yang dilanggar, semakin baik nilai *fitness* yang diperoleh. Fungsi *fitness* untuk masalah optimasi penjadwalan dokter dan perawat di IGD dirumuskan sebagai berikut:

$$fitness = \frac{1}{1 + ((10 \times \sum HC_i) + (0,5 \times \sum SC_i))}$$

di mana:

$\sum HC_i$: jumlah pelanggaran *hard constraint*, $i = 1, 2, \dots, 5$

$\sum SC_i$: jumlah pelanggaran *soft constraint*, $i = 1, 2, 3$

Setiap pelanggaran *hard constraint* diberikan bobot pelanggaran 10, sedangkan setiap pelanggaran *soft constraint* diberikan bobot pelanggaran 0,5. Berdasarkan fungsi *fitness* tersebut, semakin tinggi nilai *fitness* yang diperoleh, semakin baik kualitas solusi yang dihasilkan. Sebaliknya, semakin rendah nilai *fitness* yang diperoleh, semakin buruk kualitas solusinya. Nilai penalti memiliki hubungan terbalik dengan nilai *fitness*; semakin rendah nilai penalti yang

diperoleh, semakin baik nilai *fitness* yang didapatkan Penjelasan pelanggaran setiap *hard constraint* dan *soft constraint* disajikan pada Tabel 3.2.

Tabel 3.2 Keterangan Setiap Pelanggaran

Pelanggaran	Jenis Pelanggaran	Keterangan
HC_1	<i>Hard Constraint</i>	Kebutuhan akan dokter dan perawat yang bertugas di setiap <i>shift</i> terpenuhi
HC_2	<i>Hard Constraint</i>	Setiap dokter dan perawat hanya mendapat satu <i>shift</i> setiap hari (pagi, sore, malam) atau libur dalam satu periode
HC_3	<i>Hard Constraint</i>	Dokter dan perawat yang bertugas pada <i>shift</i> malam, tidak diperbolehkan mendapat <i>shift</i> pagi di hari berikutnya
HC_4	<i>Hard Constraint</i>	Dokter dan perawat tidak bertugas pada <i>shift</i> malam lebih dari 2 hari berturut-turut
HC_5	<i>Hard Constraint</i>	Total <i>shift</i> dokter dan perawat tidak kurang dari ketentuan total minimal atau melebihi ketentuan total maksimal dalam satu periode penjadwalan
SC_1	<i>Soft Constraint</i>	Setiap dokter dan perawat tidak ditugaskan pada waktu <i>shift</i> pagi lebih dari 3 hari berturut-turut
SC_2	<i>Soft Constraint</i>	Setiap dokter dan perawat tidak ditugaskan pada waktu <i>shift</i> sore lebih dari 3 hari berturut-turut
SC_3	<i>Soft Constraint</i>	Setiap dokter dan perawat mendapatkan alokasi antara <i>shift</i> -nya seimbang

Dalam contoh kasus ini, total *shift* jaga minimal untuk dokter dan perawat ditetapkan sebanyak 2 *shift*, sedangkan total *shift* jaga maksimal ditetapkan sebanyak 3 *shift*. Berdasarkan pemaparan nilai *fitness* di atas akan dilakukan perhitungan nilai *fitness*-nya dalam contoh kasus ini. Berikut adalah perhitungan

nilai *fitness* pada masing-masing kromosom dalam populasi awal yang ditunjukkan pada Tabel 3.3.

Tabel 3.3 Nilai *Fitness* Kromosom Populasi Awal

Kromosom	Pelanggaran								<i>Fitness</i>
	HC_1	HC_2	HC_3	HC_4	HC_5	SC_1	SC_2	SC_3	
Kromosom 1	0	0	1	1	4	0	0	6,667	0,015
Kromosom 2	0	0	2	0	4	0	0	6,500	0,016
Kromosom 3	0	0	1	0	4	0	0	4,833	0,019
Kromosom 4	0	0	1	1	2	0	0	5,167	0,022

Setelah nilai *fitness* dari setiap kromosom dalam populasi awal dihitung, langkah selanjutnya adalah melakukan proses seleksi.

5. Seleksi Populasi

Seleksi merupakan proses pemilihan individu yang akan menjadi *parent* dalam proses reproduksi. Metode yang digunakan dalam kasus ini adalah seleksi turnamen. Dalam metode ini, sejumlah k kromosom (ukuran turnamen) dipilih secara acak dari populasi untuk berkompetisi dalam turnamen. Kromosom dengan nilai *fitness* tertinggi akan terpilih sebagai *parent*. Proses ini diulang hingga jumlah induk yang dibutuhkan terpenuhi. Berikut langkah – langkah seleksi turnamen:

- a. Tentukan ukuran turnamen (k), ukuran turnamen umumnya lebih kecil dari ukuran populasi.
- b. Pilih k kromosom secara acak dari populasi.
- c. Bandingkan nilai *fitness* dari k kromosom terpilih.
- d. Pilih kromosom dengan nilai *fitness* tertinggi sebagai pemenang turnamen.
- e. Kembalikan semua kromosom, termasuk pemenang, ke dalam populasi.
- f. Ulangi langkah a - e untuk mendapatkan *parent* kedua.
- g. Lanjutkan proses hingga jumlah *parent* yang dibutuhkan terpenuhi.

Dalam contoh kasus ini, dibutuhkan dua induk untuk proses reproduksi dengan ukuran turnamen $k = 2$. Berikut prosesnya:

- a. Turnamen pertama:

Misalnya pada turnamen pertama, Kromosom 2 dan 4 terpilih

Kromosom 2	0,016
-------------------	-------

Kromosom 4	0,022
-------------------	-------

Kromosom 4 terpilih menjadi *parent* pertama karena mempunyai nilai *fitness* lebih besar dari Kromosom 2.

b. Turnamen Kedua:

Misalnya pada turnamen pertama, Kromosom 1 dan 2 terpilih.

Kromosom 1	0,015
-------------------	-------

Kromosom 2	0,016
-------------------	-------

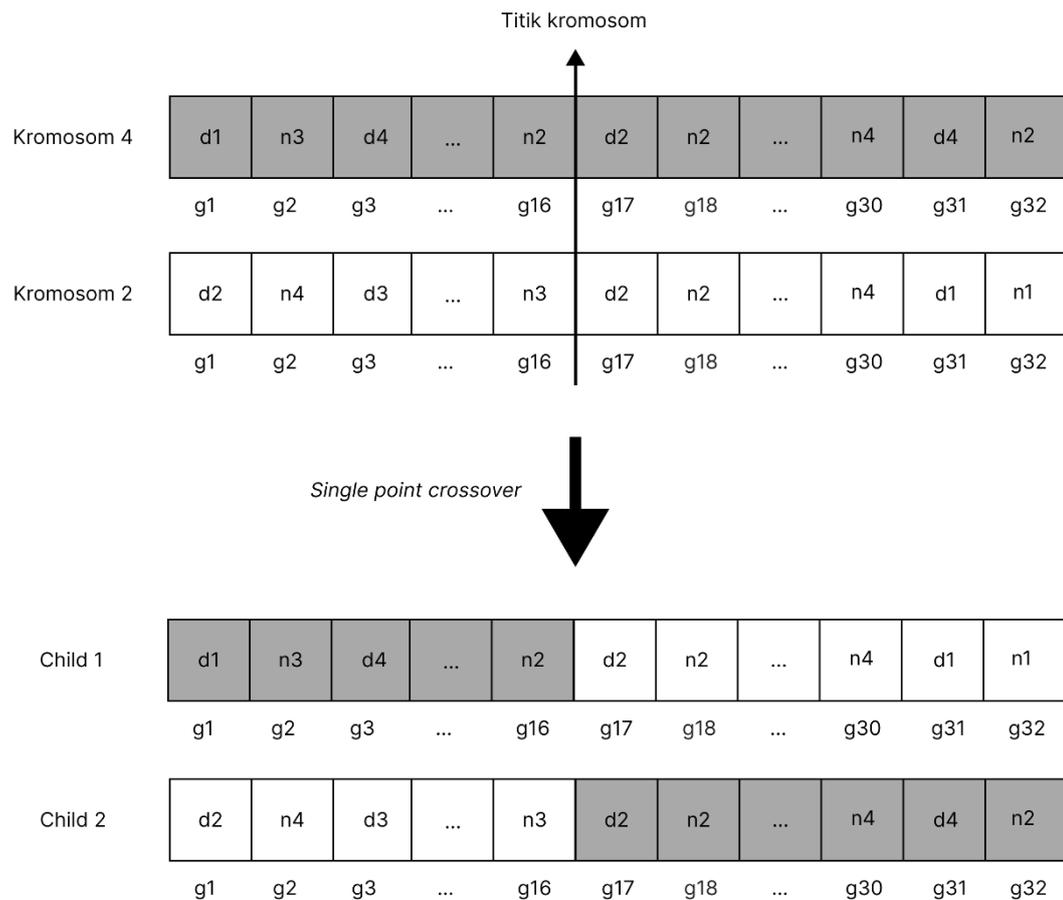
Kromosom 2 terpilih menjadi *parent* kedua karena mempunyai nilai *fitness* lebih besar dari Kromosom 1.

Dari proses tersebut, terpilih Kromosom 4 dan Kromosom 2 menjadi *parent*. Kromosom yang terpilih menjadi *parent* akan melanjutkan ke proses *crossover*.

6. *Crossover*

Crossover adalah proses pertukaran gen antara dua kromosom untuk menciptakan kromosom baru melalui satu titik tertentu. Pemilihan titik tersebut terjadi dengan menghasilkan angka acak di antara 1 hingga panjang kromosom (n), yang menandai lokasi pemotongan kromosom. Dalam metode *single point crossover*, terdapat parameter probabilitas *crossover* (P_c) yang dapat bernilai antara 0 hingga 1. Bilangan acak dipilih dari rentang 0 hingga 1, dan jika bilangan tersebut kurang dari nilai P_c , maka proses *crossover* dilakukan; sebaliknya, jika lebih besar dari nilai P_c , maka tidak ada *crossover* yang terjadi.

Dalam contoh kasus ini, dilakukan *crossover* antara Kromosom 4 dan Kromosom 2 dengan nilai P_c sebesar 0,7. Misalkan, bilangan acak yang dipilih untuk pasangan tersebut adalah 0,25, yang mana kurang dari nilai P_c , sehingga proses *crossover* dilaksanakan. Titik *crossover* dipilih pada posisi gen ke-16. Hasil dari *crossover* ini menghasilkan dua kromosom *child*, yaitu *Child 1* dan *Child 2*. Proses *crossover* dapat dilihat pada Gambar 3.4.



Gambar 3.4 Proses *Single Point Crossover*

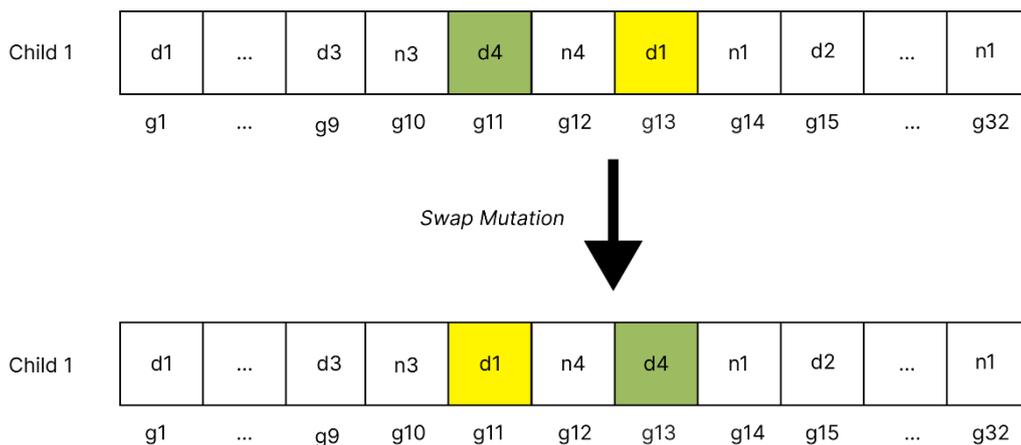
Child 1 terbentuk dari kombinasi bagian depan Kromosom 4 dan bagian belakang Kromosom 2, sementara *Child 2* terbentuk dari kombinasi bagian depan Kromosom 2 dan bagian belakang Kromosom 4. Kromosom *child* yang dihasilkan dari proses *crossover* ini akan menjadi bagian dari populasi untuk generasi berikutnya dalam Algoritma Genetika.

7. Mutasi

Setelah proses *crossover*, dilakukan proses mutasi. Mutasi bertujuan untuk memperkenalkan variasi dalam populasi dengan mengubah satu atau lebih gen dalam sebuah individu. Proses ini penting untuk memastikan keanekaragaman individu dalam populasi dan mencegah konvergensi prematur. Mutasi dilakukan berdasarkan parameter probabilitas mutasi (P_m).

Proses mutasi dalam Algoritma Genetika melibatkan beberapa langkah. Pertama, kromosom dipilih dari populasi untuk mengalami mutasi. Kemudian, bangkitkan nilai acak yang berada dalam rentang antara 0 hingga 1. Jika nilai acak tersebut lebih rendah dari probabilitas mutasi (P_m), maka mutasi akan dilakukan pada kromosom yang terpilih. Namun, jika nilai acak lebih besar dari P_m , maka kromosom tersebut tidak akan mengalami mutasi. Dalam penelitian ini, menggunakan metode *swap mutation*, di mana posisi dua gen dalam kromosom yang sama akan dipertukarkan.

Pada contoh kasus ini mutasi akan dilakukan pada kromosom-kromosom dengan probabilitas mutasi sebesar 0,6. Misalkan untuk *Child 1*, diperoleh bilangan acak sebesar 0,4. Karena nilai tersebut lebih kecil dari P_m , maka mutasi dilakukan pada *Child 1*. Misalkan gen ke-11 dipilih untuk ditukar dengan gen ke-13. Proses mutasi ini dapat dilihat pada Gambar 3.5.



Gambar 3.5 Proses *Swap Mutation*

Kromosom yang telah dimutasi akan menjadi bagian dari solusi akhir, memberikan variasi yang dapat membantu algoritma genetika dalam mencari solusi optimal. Proses reproduksi, menghasilkan kromosom – kromosom baru. Kromosom – kromosom baru tersebut, kemudian akan dihitung kembali nilai *fitness*-nya. Hasil perhitungan nilai *fitness* untuk kromosom – kromosom baru disajikan pada Tabel 3.4.

Tabel 3.4 Nilai *Fitness* Kromosom – Kromosom Baru

Kromosom	Nilai <i>Fitness</i>
Kromosom 4	0,022
Kromosom 2	0,016
<i>Child 1</i>	0,023
<i>Child 2</i>	0,012

Setelah menghitung kembali nilai *fitness* kromosom-kromosom baru, didapatkan bahwa *Child 1* merupakan solusi terbaik karena memiliki nilai *fitness* tertinggi. Berdasarkan solusi terbaik yang diperoleh menggunakan Algoritma Genetika, penjadwalan dokter dan perawat di IGD untuk periode penjadwalan 7 hari dengan jumlah 4 dokter dan 4 perawat dapat dilihat pada Tabel 3.5.

Tabel 3.5 Penjadwalan Dokter dan Perawat di IGD Berdasarkan Kromosom Hasil Algoritma Genetika

	Pagi		Sore		Malam		Libur	
Hari 1	d1	n3	d4	n2	d3	n1	d2	n4
Hari 2	d3	n3	d1	n4	d4	n1	d2	n2
Hari 3	d2	n2	d3	n1	d1	n3	d4	n4
Hari 4	d4	n3	d2	n2	d3	n4	d1	n1