

**ESTIMASI PARAMETER MODEL HIV  
MENGUNAKAN ALGORITMA GENETIKA**

**SKRIPSI**

Oleh :  
**KARTIKOWATI**  
**0510940029-94**



**PROGRAM STUDI MATEMATIKA  
JURUSAN MATEMATIKA  
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS BRAWIJAYA  
MALANG  
2009**

UNIVERSITAS BRAWIJAYA



**ESTIMASI PARAMETER MODEL HIV  
MENGUNAKAN ALGORITMA GENETIKA**

**SKRIPSI**

Sebagai salah satu syarat untuk memperoleh gelar  
Sarjana Sains dalam bidang Matematika

Oleh :  
**KARTIKOWATI**  
**0510940029-94**



**PROGRAM STUDI MATEMATIKA  
JURUSAN MATEMATIKA  
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS BRAWIJAYA  
MALANG**

**2009**

iii

UNIVERSITAS BRAWIJAYA



**LEMBAR PENGESAHAN SKRIPSI**

**ESTIMASI PARAMETER MODEL HIV  
MENGUNAKAN ALGORITMA GENETIKA**

Oleh:  
**KARTIKOWATI**  
**0510940029-94**

Setelah dipertahankan di depan Majelis Penguji  
pada tanggal 9 Juli 2009  
dan dinyatakan memenuhi syarat untuk memperoleh gelar  
Sarjana Sains dalam bidang Matematika

**Pembimbing I**

**Pembimbing II**

**Dra. Trisilowati, MSc**  
**NIP. 131 837 955**

**Drs. Moch. Aruman Imron, MSi**  
**NIP. 131 879 051**

**Mengetahui,**  
**Ketua Jurusan Matematika**  
**Fakultas MIPA Universitas Brawijaya**

**Dr. Agus Suryanto, MSc**  
**NIP. 132 126 049**

UNIVERSITAS BRAWIJAYA



## LEMBAR PERNYATAAN

Saya yang bertanda tangan di bawah ini :

Nama : Kartikowati  
NIM : 0510940029-94  
Jurusan : Matematika  
Penulis Skripsi berjudul : ESTIMASI PARAMETER MODEL  
HIV MENGGUNAKAN  
ALGORITMA GENETIKA

Dengan ini menyatakan bahwa :

1. Isi Skripsi yang saya buat adalah benar-benar karya sendiri dan tidak menjiplak karya orang lain, selain nama-nama yang termaktub di isi dan tertulis di daftar pustaka dalam Skripsi ini.
2. Apabila di kemudian hari ternyata Skripsi yang saya tulis terbukti hasil jiplakan, maka saya bersedia menanggung segala resiko yang akan saya terima.

Demikian pernyataan ini dibuat dengan segala kesadaran.

Malang, 9 Juli 2009  
Yang menyatakan,

( Kartikowati )  
NIM. 0510940029-94

UNIVERSITAS BRAWIJAYA



# ESTIMASI PARAMETER MODEL HIV MENGUNAKAN ALGORITMA GENETIKA

## ABSTRAK

Skripsi ini membahas konstruksi model matematika sistem imun HIV dan estimasi parameter yang terdapat dalam model. Model ini menjelaskan interaksi antara populasi sel T CD4<sup>+</sup> yang tidak terinfeksi ( $X_1$ ), sel T CD4<sup>+</sup> yang terinfeksi ( $X_2$ ), dan virus yang hidup bebas dalam darah ( $X_3$ ). Interaksi antara populasi sel melibatkan parameter-parameter, yaitu laju produksi sel T CD4<sup>+</sup> yang tidak terinfeksi ( $S$ ), laju kematian sel T CD4<sup>+</sup> yang tidak terinfeksi ( $d$ ), laju infeksi pada sel T CD4<sup>+</sup> oleh virus bebas ( $\beta$ ), laju kematian sel T CD4<sup>+</sup> yang terinfeksi ( $\mu_1$ ), laju produksi virus bebas ( $k$ ), dan laju kematian virus bebas karena antibodi ( $\mu_2$ ). Pada umumnya parameter-parameter tersebut tidak diketahui nilainya dan dapat diestimasi dengan memanfaatkan data percobaan. Dalam Skripsi ini, estimasi parameter dilakukan dengan menggunakan algoritma genetika dengan cara meminimumkan fungsi objektif  $h$  yang

didefinisikan sebagai jumlah kuadrat *error*,  $h = \sum_{i=1}^3 \sum_{j=1}^n [y_{i,j} - y_{i,j}^*]^2$ ,

dengan  $y_{i,j}$  adalah komponen data percobaan dan  $y_{i,j}^*$  adalah komponen penyelesaian persamaan diferensial. Data akan dibangkitkan dari model dengan mengambil nilai parameter awal  $S=1$ ,  $d=0.8$ ,  $\beta=1$ ,  $\mu_1=0.8$ ,  $k=1$ ,  $\mu_2=0.01078$ ,  $X_1=1$ ,  $X_2=0.2$ , dan  $X_3=0.8$ , untuk  $t=0$  sampai dengan  $t=1$  dan  $\Delta t=0.1$ .

**Kata kunci** : estimasi parameter, sistem imun, HIV, algoritma genetika

UNIVERSITAS BRAWIJAYA



# PARAMETER ESTIMATION OF AN HIV MODEL BY GENETIC ALGORITHM

## ABSTRACT

This final project discusses construction of a mathematical model of human immune system for HIV infection and parameter estimation of this model. This model explains interaction between population of non-infected  $CD4^+$  T cells ( $X_1$ ), infected  $CD4^+$  T cells ( $X_2$ ), and virus which live freely in the blood ( $X_3$ ). Interaction among population of cells involves some parameters, i.e. rate of production of non-infected  $CD4^+$  T cells ( $S$ ), death rate of the non-infected  $CD4^+$  T cells ( $d$ ), infection rate of non-infected  $CD4^+$  T cells by HIV virus ( $\beta$ ), death rate of infected  $CD4^+$  T cells ( $\mu_1$ ), rate of production of free virus ( $k$ ), and death rate of free virus ( $\mu_2$ ). Generally these parameters are not known and could be predicted by using experimental data. In this final project, parameter estimation is conducted by using genetic algorithm that minimizing the objective function  $h$  which is defined as sum of squares error,

$$h = \sum_{i=1}^3 \sum_{j=1}^n [y_{i,j} - y_{i,j}^*]^2 \text{ where } y_{i,j} \text{ is observed value and } y_{i,j}^* \text{ is}$$

predicted value. The data will be generated from this model by taking initial conditions  $S=1$ ,  $d=0.8$ ,  $\beta=1$ ,  $\mu_1=0.8$ ,  $k=1$ ,  $\mu_2=0.01078$ ,  $X_1=1$ ,  $X_2=0.2$ , and  $X_3=0.8$ , for  $t=0$  until  $t=1$  and  $\Delta t=0.1$ .

**Key words** : parameter estimation, immune system, HIV, genetic algorithm

UNIVERSITAS BRAWIJAYA



## KATA PENGANTAR

Segala puji dan syukur penulis panjatkan ke hadirat Allah SWT atas segala rahmat serta hidayahNya, sehingga penulis dapat menyelesaikan Skripsi yang berjudul “Estimasi Parameter Model HIV Menggunakan Algoritma Genetika”. Shalawat dan salam selalu tercurahkan kepada Nabi Muhammad SAW sebagai suri tauladan bagi penulis.

Dalam penyusunan Skripsi ini banyak pihak yang telah membantu dengan memberikan dukungan baik moral maupun spiritual secara langsung maupun tidak langsung. Oleh karena itu, penulis ingin menyampaikan ucapan terima kasih kepada:

1. Dra. Trisilowati, MSc selaku pembimbing I atas segala bimbingan, motivasi, pengarahan dan nasihat yang telah diberikan selama penulisan Skripsi ini.
2. Drs. Moch. Aruman Imron, Msi selaku pembimbing II atas segala bimbingan, pengarahan dan nasihat yang telah diberikan selama penulisan Skripsi ini.
3. Kedua orang tuaku tercinta, serta adikku Irum atas segala kasih sayang, dukungan dan do'a yang tiada henti.
4. Syaiful Anam, S.Si, MT, Drs. Hery Subagio, M.Kes, dan Dr. Wuryansari Muharini K., MSi selaku dosen penguji atas koreksi serta motivasinya sehingga Skripsi ini dapat terselesaikan.
5. Drs. Imam Nurhadi Purwanto, MT selaku dosen pembimbing akademik atas segala bimbingan dan motivasi yang telah diberikan.
6. Dr. Agus Suryanto, MSc selaku Ketua Jurusan Matematika.
7. Seluruh staf pengajar dan karyawan TU Jurusan Matematika atas segala bantuannya.
8. Semua sahabatku, Nia, Far, Prisca, Liza, Zuro, Indra, Matris, Ira, Dina, Astika yang tidak pernah bosan menghibur dan memberi dukungan, serta teman-teman seperjuangan Matematika 2005 juga keluarga besar kost Jl. Bunga Anyelir 5 atas semangat yang diberikan dan kebersamaannya selama ini.
9. Serta semua pihak yang tidak dapat penulis sebutkan satu persatu.

Penulis menyadari bahwa masih banyak kekurangan dalam penulisan Skripsi ini, oleh karena itu penulis dengan senang hati

menerima masukan, kritik dan saran yang membangun demi perbaikan selanjutnya.

Akhir kata, penulis berharap semoga tulisan ini dapat memberikan manfaat dan sumbangan bagi dunia Sains di masa yang akan datang.

Malang, 9 Juli 2009

Penulis



## DAFTAR ISI

	Halaman
<b>HALAMAN JUDUL</b> .....	i
<b>LEMBAR PENGESAHAN</b> .....	iii
<b>LEMBAR PERNYATAAN</b> .....	v
<b>ABSTRAK</b> .....	vii
<b>ABSTRACT</b> .....	ix
<b>KATA PENGANTAR</b> .....	xi
<b>DAFTAR ISI</b> .....	xiii
<b>DAFTAR TABEL</b> .....	xv
<b>DAFTAR GAMBAR</b> .....	xvii
<b>DAFTAR LAMPIRAN</b> .....	xix
 <b>BAB I PENDAHULUAN</b>	
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	2
1.3 Batasan Masalah .....	3
1.4 Tujuan Penelitian .....	3
 <b>BAB II TINJAUAN PUSTAKA</b>	
2.1 Persamaan Diferensial .....	5
2.2 Sistem Otonomus .....	6
2.3 Optimasi .....	6
2.4 Metode Runge-Kutta .....	8
2.5 Metode Kuadrat Terkecil .....	10
2.6 Algoritma Genetika .....	10
2.6.1 Struktur Umum Algoritma Genetika .....	10
2.6.2 Komponen-komponen Algoritma Genetika .....	11
2.6.3 Parameter Algoritma Genetika .....	17
2.6.4 Kekonvergenan Algoritma Genetika .....	18
2.7 HIV ( <i>Human Immunodeficiency Virus</i> ) .....	19
2.7.1 Infeksi HIV dan AIDS .....	19
2.7.2 Cara Kerja Infeksi HIV .....	19
2.7.3 Cara Penularan HIV .....	20
 <b>BAB III METODOLOGI PENELITIAN</b>	
3.1 Konstruksi Model Sistem Imun HIV .....	23

3.2 Mengestimasi Parameter dengan Algoritma Genetika ..... 23

**BAB IV HASIL DAN PEMBAHASAN**

4.1 Struktur Model ..... 27  
4.2 Konstruksi Model ..... 28  
4.3 Estimasi Parameter Model Sistem Imun HIV ..... 30  
    4.3.1 Solusi Numerik pada Kondisi Awal ..... 30  
    4.3.2 Contoh Implementasi Algoritma Genetika ..... 34  
    4.3.3 Simulasi Estimasi Parameter dengan Menggunakan  
        Data yang Dibangkitkan ..... 44

**BAB V PENUTUP**

5.1 Kesimpulan ..... 55  
5.2 Saran ..... 55

**DAFTAR PUSTAKA** ..... 57

**LAMPIRAN** ..... 59



## DAFTAR TABEL

	Halaman
Tabel 2.1 Skema Pengkodean <i>Real Number Encoding</i> .....	12
Tabel 2.2 Skema Pengkodean <i>Discrete Decimal Encoding</i> .....	12
Tabel 2.3 Skema Pengkodean <i>Binary Encoding</i> .....	12
Tabel 2.4 Contoh Proses <i>Crossover</i> untuk <i>Binary Encoding</i> ....	15
Tabel 2.5 Contoh Proses Mutasi untuk <i>Binary Encoding</i> .....	16
Tabel 4.1 Jumlah Individu dalam Satu Populasi .....	34
Tabel 4.2 Nilai Parameter dalam Satu Populasi.....	35
Tabel 4.3 Data Solusi Numerik dalam Satu Populasi .....	36
Tabel 4.4 Data yang Dibangkitkan dengan Nilai Awal .....	39
Tabel 4.5 Nilai <i>Fitness</i> Setiap Individu dalam Satu Populasi ....	40
Tabel 4.6 Nilai <i>Fitness</i> Setelah Diurutkan .....	41
Tabel 4.7 <i>Linear Fitness Ranking</i> .....	41
Tabel 4.8 Nilai Kumulatif pada <i>Roulette Wheel Selection</i> .....	42
Tabel 4.9 Hasil Estimasi untuk 1 Kali Percobaan pada Berbagai Ukuran Populasi dan Probabilitas <i>Crossover</i> .....	49
Tabel 4.10 Hasil Estimasi untuk 1 Kali Percobaan pada Berbagai Ukuran Populasi dan Probabilitas Mutasi .....	50
Tabel 4.11 Hasil Estimasi untuk Setiap 10 Kali Percobaan .....	53

UNIVERSITAS BRAWIJAYA



## DAFTAR GAMBAR

	Halaman
Gambar 2.1	(a). Fungsi yang memuat titik minimum lokal sekaligus minimum global. (b). Fungsi yang memuat titik minimum lokal dan titik minimum global ..... 8
Gambar 2.2	Analisis Kurva Data Pengamatan ..... 10
Gambar 3.1	Diagram Alir Estimasi Parameter Menggunakan Algoritma Genetika ..... 25
Gambar 4.1	Diagram Interaksi Virus HIV ..... 27
Gambar 4.2	Solusi Numerik ( $X_1$ ), ( $X_2$ ), dan ( $X_3$ ) Terhadap Waktu $t$ ..... 33
Gambar 4.3	Proses Pindah Silang ..... 43
Gambar 4.4	Proses Mutasi ..... 43
Gambar 4.5	Data yang Dibangkitkan dengan Nilai Parameter Awal ..... 45
Gambar 4.6	Perbandingan antara Data dan Solusi Numerik pada Generasi ke-1 ..... 46
Gambar 4.7	Estimasi Parameter Menggunakan Algoritma Genetika ..... 47
Gambar 4.8	Perbandingan antara Data dan Solusi Numerik pada Generasi ke-50 ..... 48
Gambar 4.9	Estimasi Parameter Menggunakan Algoritma Genetika dengan $Pop\_size$ 100, $p_c$ 0.95, $p_m$ 0.01 .... 51
Gambar 4.10	Perbandingan antara Data dan Solusi Numerik untuk 1 Kali Percobaan Setiap Simulasi ..... 52
Gambar 4.11	Perbandingan Antara Data dan Solusi Numerik untuk 10 Kali Percobaan Setiap Simulasi ..... 54

UNIVERSITAS BRAWIJAYA



## DAFTAR LAMPIRAN

Halaman

Lampiran 1. Proses Simulai Estimasi Parameter dengan $Pop\_size$ , $p_c$ , dan $p_m$ Berbeda untuk 1 Kali Percobaan Setiap Simulasi .....	59
Lampiran 2. Proses Simulai Estimasi Parameter dengan 10 Kali Percobaan Setiap Simulasi .....	71



UNIVERSITAS BRAWIJAYA



# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

*Acquired Immune Deficiency Syndrome* (AIDS) adalah sekumpulan gejala dan infeksi yang timbul karena rusaknya sistem kekebalan tubuh manusia akibat infeksi virus HIV (*Human Immunodeficiency Virus*). Infeksi HIV menyebabkan menurunnya tingkat kekebalan tubuh secara cepat, yang menyebabkan berkurangnya daya imun. (Anonimus<sup>3</sup>. 2008). Dalam perkembangannya, virus HIV membutuhkan sel-sel kekebalan untuk berkembang biak. Secara alamiah sel kekebalan akan dimanfaatkan oleh virus sebagai mesin fotokopi, namun virus ini akan merusak mesin fotokopinya setelah mendapatkan hasil *copy* virus baru dalam jumlah yang cukup banyak. Dan sehingga lama-kelamaan sel kekebalan akan habis dan jumlah virus menjadi sangat banyak.

Sebagai penyakit menular, HIV dapat menyebar pada beberapa bagian tubuh seperti aliran darah, semen, cairan vagina dan air susu ibu. Penularannya dapat melalui hubungan intim, transfusi darah, penggunaan jarum suntik yang telah terkontaminasi, kelahiran ataupun menyusui. Seseorang yang mengidap HIV tampak seperti orang normal. Ini berarti seseorang yang terinfeksi HIV tidak menunjukkan suatu gejala klinis sehingga kondisi ini dikatakan asimtomatik (tanpa gejala). Dengan kata lain, seseorang merasa sehat selama bertahun-tahun sebelum AIDS muncul. Di sinilah letak bahaya terselubung bagi penyebaran dan penularan HIV, karena tidak dapat diketahui bahwa seseorang terinfeksi atau tidak.

Kini AIDS telah menjadi wabah penyakit. Pada Januari 2006, UNAIDS bekerjasama dengan WHO memperkirakan bahwa AIDS telah menyebabkan kematian lebih dari 25 juta orang sejak pertama kali diakui pada tanggal 5 Juni 1981. Dengan kata lain, penyakit ini merupakan salah satu wabah paling mematikan dalam sejarah.

Pada umumnya, pemodelan matematika yang terkait dengan masalah HIV ada dua macam yaitu model epidemik penyakit AIDS dan model sistem imun HIV. Model yang dikaji dalam Skripsi ini adalah model sistem imun HIV dalam tubuh, dengan membentuk model menjadi tiga kompartemen yaitu jumlah sel T CD4<sup>+</sup> yang tidak terinfeksi, sel T CD4<sup>+</sup> yang terinfeksi dan virus yang hidup

bebas dalam darah. Model matematika yang terbentuk berupa sistem dinamik yang memuat parameter-parameter yang belum diketahui nilainya. Suatu model matematika tidak dapat diinterpretasikan dengan baik bila nilai parameter yang ada pada model tidak diketahui. Dengan demikian, maka perlu dilakukan estimasi parameter. Pada sistem persamaan diferensial, estimasi parameter merupakan suatu masalah yang sulit. Untuk melakukan proses estimasi, diperlukan data dan solusi numerik persamaan diferensial. Dengan menggunakan nilai parameter tebakan awal, sistem persamaan diferensial dapat diselesaikan secara numerik. Selanjutnya solusi numerik dicocokkan terhadap data menggunakan fungsi objektif yang berupa *error* dengan pendekatan metode kuadrat terkecil (*least squares method*). Kesulitannya adalah bagaimana meminimumkan fungsi objektif tersebut dengan menggunakan metode optimisasi. Metode optimisasi yang sering dipakai untuk estimasi adalah Newton Raphson, Levenberg-Marquardt, Nelder-Mead Simpleks dan *multiple shooting*.

Estimasi parameter pada Skripsi ini dilakukan menggunakan suatu metode optimisasi yang dikenal sebagai algoritma genetika (*genetic algorithm*). Algoritma genetika seringkali digunakan untuk menyelesaikan permasalahan-permasalahan yang sulit diselesaikan secara analitis. Pada umumnya, algoritma genetika dapat menentukan nilai optimum global suatu masalah optimasi. Algoritma genetika merupakan salah satu cabang kecerdasan buatan (*artificial intelligence*) yang berupa metode pencarian dan penyelesaian suatu masalah berdasarkan algoritma yang mengadaptasi mekanisme evolusi genetik dan seleksi alam yang didasari oleh Teori Darwin, di mana individu yang lebih kuat akan bertahan hidup dalam lingkungan yang berkompetisi.

## 1.2 Rumusan Masalah

Berdasarkan latar belakang di atas, maka permasalahan yang akan dibahas dalam Skripsi ini adalah sebagai berikut.

1. Bagaimana mengkonstruksi model sistem imun HIV?
2. Bagaimana mengestimasi parameter-parameter pada model sistem imun HIV dengan menggunakan algoritma genetika?

### 1.3 Batasan Masalah

Penulisan Skripsi ini difokuskan pada pembahasan dengan beberapa batasan masalah, yaitu:

1. Jumlah sel T CD4<sup>+</sup> yang dihasilkan oleh *thymus*, sumsum tulang dan organ lainnya pada individu sehat adalah konstan.
2. Nilai awal jumlah sel T CD4<sup>+</sup> yang tidak terinfeksi, sel T CD4<sup>+</sup> yang terinfeksi dan virus yang hidup bebas dalam darah pada proses estimasi, sama dengan nilai awal yang dipakai untuk pembangkitan data.

### 1.4 Tujuan Penelitian

Tujuan penulisan Skripsi ini adalah untuk:

1. Mengkonstruksi model sistem imun HIV.
2. Mengestimasi parameter-parameter pada model sistem imun HIV dengan menggunakan algoritma genetika.



UNIVERSITAS BRAWIJAYA



## BAB II TINJAUAN PUSTAKA

Pada bab ini diberikan beberapa definisi dan teori untuk membantu memahami permasalahan yang akan dibahas dan juga digunakan sebagai acuan dalam pembahasan.

### 2.1 Persamaan Diferensial

#### Definisi 2.1

Persamaan diferensial adalah persamaan yang memuat hubungan antara turunan dari satu atau beberapa fungsi tak diketahui yang disebut sebagai variabel tak bebas terhadap satu atau lebih variabel bebas (Finizio dan Ladas, 1982).

#### Definisi 2.2

Persamaan diferensial biasa adalah persamaan yang memuat turunan biasa dari satu atau lebih variabel tak bebas terhadap satu variabel bebas (Ross, 1984).

Persamaan diferensial dikatakan mempunyai orde (tingkat)  $n$  jika turunan tertinggi yang terdapat dalam persamaan diferensial tersebut adalah turunan ke  $n$ . Suatu persamaan diferensial disebut mempunyai derajat  $k$  jika turunan tertinggi dalam persamaan diferensial itu berderajat  $k$ . Dengan demikian persamaan diferensial biasa orde  $n$  adalah persamaan berbentuk  $F(x, y, y', y'', \dots, y^n) = 0$  yang menyatakan hubungan antara peubah bebas  $x$ , peubah tak bebas  $y(x)$  dan turunannya yaitu  $y, y', y'', \dots, y^n$ .

Persamaan diferensial biasa linier orde  $n$ , dalam variabel tak bebas  $y$  dan variabel bebas  $x$  adalah persamaan yang dapat dinyatakan dalam bentuk

$$a_0(x) \frac{d^n y}{dx^n} + a_1(x) \frac{d^{n-1} y}{dx^{n-1}} + \dots + a_{n-1}(x) \frac{dy}{dx} + a_n(x) y = f(x), \quad (2.1)$$

dengan  $a_0(x) \neq 0$ .

Pada persamaan (2.1) fungsi  $a_0(x), a_1(x), \dots, a_n(x)$  disebut koefisien persamaan diferensial dan  $f(x)$  disebut bagian non

homogen. Persamaan diferensial (2.1) disebut homogen jika  $f(x)$  sama dengan nol.

### Definisi 2.3

Sistem persamaan diferensial berdimensi  $n$  adalah suatu sistem yang memuat  $n$  buah persamaan diferensial dengan  $n$  buah fungsi yang tidak diketahui, di mana  $n$  merupakan bilangan bulat positif dan  $n \geq 2$  (Finizio-Ladas, 1982).

### Definisi 2.4

Persamaan diferensial biasa nonlinier adalah persamaan diferensial biasa yang variabel tak bebas dan turunannya berderajat lebih dari satu, dan atau memuat perkalian antara variabel tak bebas dan turunannya (Ross, 1984).

## 2.2 Sistem Otonomus

Suatu sistem persamaan diferensial yang berbentuk

$$\begin{cases} \frac{dx}{dt} = F(x, y) \\ \frac{dy}{dt} = G(x, y), \end{cases} \quad (2.2)$$

dengan fungsi-fungsi  $F$  dan  $G$  secara eksplisit tidak dipengaruhi oleh variabel waktu  $t$  disebut sebagai sistem otonomus atau dalam hal ini disebut bidang otonomus.

Diasumsikan bahwa fungsi  $F$  dan  $G$  adalah kontinu dalam suatu domain  $D$  di bidang- $xy$  yang disebut bidang fase untuk sistem (2.2). Jika diberikan kondisi awal pada saat  $t = t_0$  yaitu  $x(t_0) = x_0$ ,  $y(t_0) = y_0$  dengan  $(x_0, y_0) \in D$ , maka sistem (2.2) memiliki solusi tunggal  $x = x(t)$ ,  $y = y(t)$  yang terdefinisi pada suatu interval  $I$  yang memuat titik  $t_0$  (Boyce dan Dprima, 1992).

## 2.3 Optimasi

Optimasi merupakan masalah memaksimalkan atau meminimalkan suatu besaran tertentu, yang disebut dengan fungsi objektif. Fungsi objektif ini bergantung pada sejumlah variabel.

Variabel-variabel ini tidak saling berhubungan atau saling bergantung melalui satu atau lebih kendala (Bronson, 1996).

Optimasi terbagi menjadi dua, optimasi linier dan nonlinier. Suatu permasalahan optimasi disebut nonlinier jika fungsi objektif dan kendalanya mempunyai bentuk nonlinier pada salah satu atau keduanya. Optimasi nonlinier dapat dibagi menjadi empat berdasarkan kriteria permasalahannya yaitu optimasi nonlinier satu variabel tanpa kendala, multivariabel tanpa kendala, satu variabel dengan kendala, dan multivariabel dengan kendala (Luknanto, 2000).

Secara umum optimasi nonlinier multivariabel tanpa kendala untuk permasalahan minimasi didefinisikan sebagai

$$\min f(x) \text{ dengan } x = (x_1, x_2, \dots, x_n) \in I (I \subseteq \mathfrak{R}^n), \quad (2.3)$$

dan  $f : I \rightarrow \mathfrak{R}$ .

Sasaran penyelesaian optimasi secara umum untuk permasalahan minimasi adalah memperoleh solusi minimum global, yaitu suatu nilai  $x$  yang membuat fungsi objektif mencapai minimum global. Agar nilai fungsi objektif memenuhi minimum global, ada beberapa hal yang perlu diperhatikan.

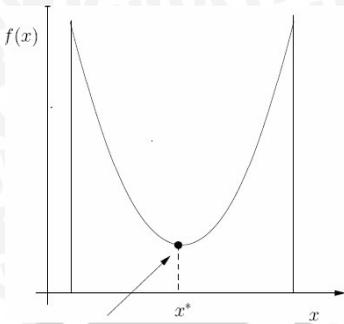
### **Definisi 2.5 (Minimum Lokal)**

Diberikan  $x^* \in I \subseteq \mathfrak{R}^n$ ,  $x^*$  dikatakan sebagai titik minimum lokal dari fungsi  $f$  dalam  $I$  jika terdapat persekitaran  $N(x^*)$  di dalam  $I$  sedemikian sehingga  $f(x^*) \leq f(x)$ ,  $\forall x \in N(x^*)$  (Leader, 2004).

### **Definisi 2.6 (Minimum Global)**

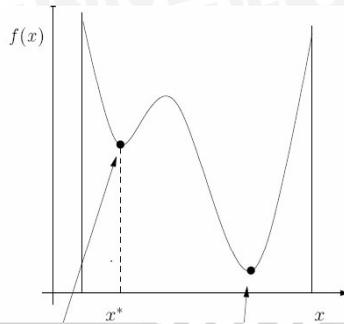
Diberikan  $x^* \in I \subseteq \mathfrak{R}^n$ ,  $x^*$  disebut sebagai titik minimum global fungsi  $f$  pada selang  $I$  jika  $f(x^*) \leq f(x)$ ,  $\forall x \in I$  (Bronshstein, dkk. 2007).

Ilustrasi tentang minimum lokal dan minimum global dapat dilihat pada Gambar 2.1 berikut.



Minimum global dan lokal

**a**



Minimum lokal

Minimum global

**b**

**Gambar 2.1** (a). Fungsi yang memuat titik minimum lokal sekaligus titik minimum global. (b). Fungsi yang memuat titik minimum lokal dan titik minimum global

## 2.4 Metode Runge-Kutta

Metode Runge-Kutta merupakan metode satu langkah, karena metode tersebut hanya memerlukan satu titik sebelumnya untuk menghitung nilai yang baru. Metode Runge-Kutta yang paling terkenal adalah Runge-Kutta orde empat.

Masalah syarat awal

$$x' = f(t, x) \quad x(t_0) = x_0$$

dapat diselesaikan dengan metode Runge-Kutta orde empat dengan rumus rekursif

$$t_{n+1} = t_n + h$$

$$x_{n+1} = x_n + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4), n = 0, 1, \dots, \quad (2.4)$$

dimana  $h$  adalah ukuran langkah dan

$$k_1 = hf(t_n, x_n)$$

$$k_2 = hf\left(t_n + \frac{h}{2}, x_n + \frac{1}{2}k_1\right)$$

$$k_3 = hf\left(t_n + \frac{h}{2}, x_n + \frac{1}{2}k_2\right)$$

$$k_4 = hf(t_n + h, x_n + k_3). \quad (2.5)$$

Misalkan  $x_1(t), \dots, x_m(t)$  adalah solusi sistem yang terdiri dari  $m$  persamaan orde satu:

$$\begin{aligned} x_1'(t) &= f_1(t, x_1, x_2, \dots, x_m), \\ x_2'(t) &= f_2(t, x_1, x_2, \dots, x_m), \\ &\vdots \\ x_m'(t) &= f_m(t, x_1, x_2, \dots, x_m), \end{aligned}$$

dan memenuhi syarat awal

$$x_1(t_0) = a_0, \quad x_2(t_0) = a_2, \quad \dots, \quad x_m(t_0) = a_m.$$

Misalkan  $x_{n,i}$  merupakan taksiran untuk  $x_i(t_n)$  untuk  $i = 1, 2, \dots, m$  dan  $t_n = t_0 + nh$  untuk  $n = 0, 1, \dots$ , maka rumus rekursif metode Runge-Kutta orde empat untuk sistem persamaan diperoleh dari rumus (2.5) dan (2.6) dengan memandang  $x_n, k_1, k_2, k_3$ , dan  $k_4$  sebagai vektor yaitu  $x_n = (x_{n,1}, \dots, x_{n,m})$ ,  $k_1 = (k_{1,1}, \dots, k_{1,m})$ ,  $k_2 = (k_{2,1}, \dots, k_{2,m})$ , dan seterusnya.

Oleh karena itu bentuk umum metode Runge-Kutta orde empat untuk sistem persamaan diferensial adalah

$$\begin{aligned} t_{n+1} &= t_n + h, \\ x_{n+1,i} &= x_{n,i} + \frac{1}{6}(k_{1,i} + 2k_{2,i} + 2k_{3,i} + k_{4,i}), \end{aligned} \tag{2.6}$$

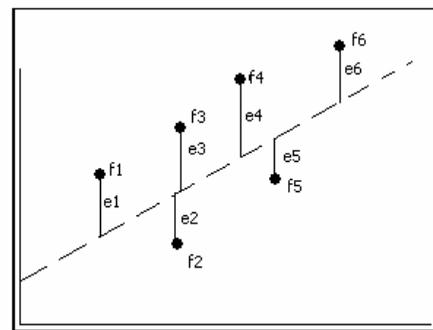
dimana  $h$  adalah ukuran langkah dan

$$\begin{aligned} k_{1,i} &= hf_i(t_n, x_{n,1}, x_{n,2}, \dots, x_{n,m}) \\ k_{2,i} &= hf_i\left(t_n + \frac{h}{2}, x_{n,1} + \frac{1}{2}k_{1,1}, x_{n,2} + \frac{1}{2}k_{1,2}, \dots, x_{n,m} + \frac{1}{2}k_{1,m}\right) \\ k_{3,i} &= hf_i\left(t_n + \frac{h}{2}, x_{n,1} + \frac{1}{2}k_{2,1}, x_{n,2} + \frac{1}{2}k_{2,2}, \dots, x_{n,m} + \frac{1}{2}k_{2,m}\right) \\ k_{4,i} &= hf_i(t_n + h, x_{n,1} + k_{3,1}, x_{n,2} + k_{3,2}, \dots, x_{n,m} + k_{3,m}) \end{aligned} \tag{2.7}$$

dengan  $n = 0, 1, \dots$ , dan  $i = 1, 2, \dots, m$  (Nagie dan Saff, 1993).

## 2.5 Metode Kuadrat Terkecil (*least squares method*)

Metode ini menggunakan asumsi bahwa kurva pendekatan terbaik adalah kurva yang mempunyai jumlah kuadrat kesalahan minimum (*least square error*) terhadap data. Misal diberikan data  $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$  dimana  $x$  adalah variabel bebas dan  $y$  adalah variabel tak bebas.



**Gambar 2.2** Analisis Kurva Data Pengamatan

Pencocokan kurva  $f(x)$  mempunyai deviasi  $e$  (*error*) pada setiap titik data  $e_1 = y_1 - f(x_1), e_2 = y_2 - f(x_2), \dots, e_n = y_n - f(x_n)$ . Menurut Spiegel (1988), kurva yang mendekati seperangkat data tertentu mempunyai ciri-ciri:

$$error = (e_1)^2 + (e_2)^2 + \dots + (e_n)^2 = \sum_{i=1}^n e_i^2 = \sum_{i=1}^n [y_i - f(x_i)]^2, \text{ minimum.} \quad (2.8)$$

Fungsi  $f(x)$  merupakan suatu polinomial pendekatan jika  $f(x) = a_0 + a_1x + a_2x^2 + \dots + a_nx^n$ , dengan  $n$  merupakan derajat polinomial yang dipergunakan.

## 2.6 Algoritma Genetika

### 2.6.1 Struktur Umum Algoritma Genetika

Algoritma genetika merupakan teknik pencarian nilai optimal berdasarkan mekanisme seleksi alam dan teori genetika. Algoritma ini menggunakan hukum probabilistik untuk memilih solusi sesuai dengan ketentuan yang diinginkan, sehingga proses pencarian solusi optimal menjadi lebih baik. Variabel permasalahan dikodekan menjadi struktur string dengan panjang berhingga.

Pencarian solusi melibatkan sejumlah populasi yang mengandung individu-individu. Individu dalam populasi merupakan representasi solusi dari permasalahan yang diberikan. Individu-individu dalam populasi tersebut berevolusi yaitu mengalami proses seleksi, pindah silang dan mutasi melalui suatu proses iterasi berkelanjutan yang disebut dengan generasi. Pada setiap generasi, dilakukan evaluasi terhadap individu berdasarkan fungsi objektif yang dioptimalkan.

Secara garis besar, algoritma genetika dapat dijelaskan dengan algoritma berikut:

1. [*Start*] buat secara acak sebuah populasi dengan  $n$  individu.
2. [*Fitness*] hitung nilai *fitness* untuk tiap individu dalam populasi tersebut.
3. [*New population*] buat sebuah populasi baru dengan mengulang langkah sebagai berikut:
  - a. [*Selection*] pilih individu dari populasi sebelumnya.
  - b. [*Crossover*] berdasar nilai probabilitas pindah silang, lakukan persilangan dua individu induk untuk membentuk dua individu baru.
  - c. [*Mutation*] berdasarkan nilai probabilitas mutasi, lakukan mutasi.
  - d. [*Accepting*] letakkan individu baru dalam populasi yang baru.
4. [*Replace*] gunakan populasi baru untuk menjalankan algoritma lebih lanjut.
5. [*Test*] jika kondisi akhir dipenuhi, stop. Jika tidak lanjutkan.
6. [*Loop*] kembali ke langkah 2.

(Kusumoputro, 2004)

### 2.6.2 Komponen-komponen Algoritma Genetika

Pada dasarnya algoritma genetika memiliki tujuh komponen yang harus dipenuhi, yaitu skema pengkodean, nilai *fitness*, seleksi, pindah silang, mutasi, elitisme, penggantian populasi.

#### 1. Skema Pengkodean

Terdapat tiga skema yang paling umum digunakan dalam pengkodean, yaitu *real number encoding*, *discrete decimal encoding*, *binary encoding*.

a. *Real number encoding*

Pada skema ini, nilai gen berada pada interval  $[0,R]$ , dimana  $R$  adalah bilangan riil positif dan biasanya  $R=1$ . Contoh *real number encoding* dapat dilihat pada Tabel 2.1.

**Tabel 2.1** Skema Pengkodean *Real Number Encoding*

$x_1$	$x_2$	$x_3$
0,2390	1,0000	0,0131
$g_1$	$g_2$	$g_3$

b. *Discrete decimal encoding*

Pada Tabel 2.2 berikut diberikan contoh skema pengkodean *discrete decimal encoding* dimana untuk setiap gen dapat bernilai salah satu dari bilangan bulat dalam interval  $[0,9]$ .

**Tabel 2.2** Skema Pengkodean *Discrete Decimal Encoding*

$x_1$			$x_2$			$x_3$		
2	3	9	9	9	9	0	1	3
$g_1$	$g_2$	$g_3$	$g_4$	$g_5$	$g_6$	$g_7$	$g_8$	$g_9$

c. *Binary encoding*

Pada Tabel 2.3 berikut diberikan contoh skema pengkodean *binary encoding* dimana untuk setiap gen hanya dapat bernilai 0 dan 1.

**Tabel 2.3** Skema Pengkodean *Binary Encoding*

$x_1$				$x_2$			$x_3$	
0	1	0	1	1	1	0	0	0
$g_1$	$g_2$	$g_3$	$g_4$	$g_5$	$g_6$	$g_7$	$g_8$	$g_9$

Pada contoh skema pengkodean dimisalkan terdapat tiga variabel, yaitu  $x_1$ ,  $x_2$  dan  $x_3$  yang dikodekan ke dalam sebuah kromosom yang terdiri dari 3 gen untuk *real number encoding*. Sedangkan pada *discrete decimal encoding* maupun *binary encoding*, ketiga variabel dikodekan ke dalam kromosom yang terdiri dari 9 gen, dimana masing-masing variabel dikodekan ke dalam 3 gen.

Dekode kromosom adalah mengubah kromosom dari *real number encoding*, *discrete decimal encoding* dan *binary encoding*

menjadi bilangan riil yang kemudian disubstitusikan sebagai nilai dari variabel  $x_j$  untuk dievaluasi ke dalam fungsi yang dicari nilai optimalnya.

Pada Skripsi ini, metode yang digunakan untuk mendekodekan sebanyak  $j$  variabel untuk satu kromosom / individu pada masing-masing *encoding* adalah:

a. *real number encoding* (2.9)  

$$x_j = r_b + (r_a - r_b)g_j$$

b. *discrete decimal encoding* (2.10)  

$$x_j = r_b + (r_a - r_b)(g_{j,1} \times 10^{-1} + g_{j,2} \times 10^{-2} + \dots + g_{j,p} \times 10^{-p})$$

c. *binary encoding* (2.11)  

$$x_j = r_b + (r_a - r_b)(g_{j,1} \times 2^{-1} + g_{j,2} \times 2^{-2} + \dots + g_{j,p} \times 2^{-p})$$

dimana:  $r_b$  = batas bawah interval  
 $r_a$  = batas atas interval  
 $j$  = 1, 2, ..., jumlah variabel  
 $g_j$  = nilai gen yang mengkodekan variabel ke- $j$   
 $p$  = jumlah gen dalam individu yang mengkodekan variabel ke- $j$ .

(Suyanto, 2005)

## 2. Nilai *Fitness*

Suatu individu dievaluasi berdasarkan suatu fungsi tertentu sebagai ukuran performansinya. Di dalam evolusi alam, individu yang bernilai *fitness* tinggi yang akan bertahan hidup. Sedangkan individu yang bernilai rendah akan mati. Pada masalah optimasi, nilai *fitness* yang digunakan antara lain:

a. untuk masalah maksimasi, nilai *fitness* ditentukan oleh fungsi berikut:

$$f = h \tag{2.12}$$

b. untuk masalah minimasi, nilai *fitness* ditentukan oleh fungsi berikut:

$$f = \frac{1}{h+a} \tag{2.13}$$

dimana,  $h$  = nilai fungsi objektif

$a$  = bilangan sangat kecil, untuk menghindari pembagian dengan 0.

*Linear fitness ranking* adalah mekanisme yang bertujuan untuk melakukan penskalaan nilai-nilai *fitness*. Metode yang digunakan adalah

$$f(i) = f_{\max} - (f_{\max} - f_{\min}) \left( \frac{R(i) - 1}{N - 1} \right) \quad (2.14)$$

di mana,  $i = 1, 2, 3, \dots, N$ .

$N$  = jumlah individu dalam populasi

$R(i)$  = ranking individu ke  $i$

$R(i) = 1$ , jika  $i$  individu bernilai *fitness* tertinggi

$R(i) = N$ , jika  $i$  individu bernilai *fitness* terendah.

(Suyanto, 2005)

### 3. Seleksi (*Selection*)

Metode seleksi umum yang digunakan untuk seleksi adalah *Roulette Wheel Selection* (seleksi roda *roulette*). Langkah-langkah perhitungan probabilitas menggunakan *roulette wheel* adalah:

1. hitung nilai *fitness* masing-masing individu  $f_i$  dengan

$$eval(f_i) = \frac{1}{h + a} \quad (2.15)$$

2. hitung total nilai *fitness* semua individu dalam populasi

$$Tot.Fitness = \sum_{i=1}^N eval(f_i) \quad (2.16)$$

3. hitung probabilitas seleksi  $p_i$  pada setiap individu

$$p_i = \frac{eval(f_i)}{Tot.Fitness} \quad (2.17)$$

4. hitung probabilitas kumulatif  $q_i$

$$q_i = \sum_{j=1}^i p_j \quad (2.18)$$

dimana,  $f_i$  = *fitness* individu ke- $i$

$i = 1, 2, 3, \dots, N$

$p_i$  = probabilitas individu ke- $i$

$q_i$  = probabilitas kumulatif individu ke- $i$

$N$  = jumlah individu dalam populasi.

Setiap individu diseleksi dengan prosedur sebagai berikut:

- bangkitkan bilangan acak riil ( $r$ ) dalam interval  $[0,1]$
- jika  $r < q_1$ , maka seleksi individu pertama  $f_1$ , jika tidak memenuhi, seleksi individu ke- $i$  yaitu  $f_i$  ( $2 \leq i \leq N$ ) dimana  $q_{i-1} < r \leq q_i$ .

(Gen dan Cheng, 1997)

#### 4. Pindah Silang (*Crossover*)

Proses pindah silang bertujuan untuk mendapatkan dua individu baru. Pindah silang dapat berakibat buruk jika ukuran populasinya sangat kecil. Untuk mengatasinya, maka digunakan suatu aturan bahwa pindah silang hanya bisa dilakukan dengan suatu probabilitas tertentu  $p_c$ , artinya, pindah silang dapat dilakukan hanya jika suatu bilangan *random*  $[0,1]$  yang dibangkitkan kurang dari  $p_c$ . Metode yang digunakan adalah pindah silang satu titik potong (*one-point crossover*). Suatu titik potong dipilih secara acak, kemudian bagian pertama dari induk 1 digabungkan dengan bagian kedua dari induk 2, dan juga sebaliknya.

Pemilihan individu induk hanya bisa dilakukan dengan suatu probabilitas tertentu,  $p_c$  sebagai berikut:

- bangkitkan bilangan acak riil ( $r$ ) dalam interval  $[0,1]$
- jika  $r < p_c$ , pilih individu untuk dilakukan *crossover*.

**Tabel 2.4** Contoh Proses *Crossover* untuk *Binary Encoding*

	titik potong									
Induk 1	1	1	0	1	1	1	0	1	0	
Induk 2	0	0	1	1	0	1	0	1	1	
Anak 1	0	0	0	1	1	1	0	1	0	
Anak 2	1	1	1	1	0	1	0	1	1	

Pada Tabel 2.4, titik potong pindah silang terletak antara gen ke dua dan gen ke tiga. Sehingga gen pertama dan ke dua pada Induk 1 digabungkan dengan gen ke tiga, gen ke empat sampai gen terakhir pada Induk 2. Sebaliknya gen pertama dan ke dua pada Induk 2 digabungkan dengan gen ke tiga, gen ke empat sampai gen terakhir pada Induk 1.

### 5. Mutasi (*mutation*)

Proses ini mengubah gen-gen pada setiap individu untuk mendapatkan individu baru berdasarkan probabilitas mutasi ( $p_m$ ).

Pada *binary encoding*, mutasi dilakukan dengan cara sebagai berikut:

- a. bangkitkan bilangan acak riil ( $r$ ) dalam interval  $[0,1]$
- b. jika  $r < p_m$ , maka nilai gen 0 diubah menjadi 1 dan 1 diubah menjadi 0.

Pada umumnya  $p_m$  dideklarasikan sebagai  $1/n$ , di mana  $n$  adalah jumlah gen dalam individu. Berikut diberikan contoh mutasi (untuk *binary encoding*) seperti tampak pada Tabel 2.5.

**Tabel 2.5** Contoh Proses Mutasi untuk *Binary Encoding*

Individu asal

0	0	0	1	1	1	0	1	0
---	---	---	---	---	---	---	---	---

$g_5$

Individu hasil mutasi

0	0	0	1	0	1	0	1	0
---	---	---	---	---	---	---	---	---

$g_5$

Pada Tabel 2.5 di atas, gen ke lima pada individu asal yang bernilai 1, diubah menjadi 0.

### 6. Elitisme

Proses ini bertujuan untuk menjaga agar individu bernilai *fitness* tertinggi tidak hilang selama evolusi dengan membuat satu atau beberapa kopi individu. Karena selama proses seleksi dilakukan secara *random*, maka tidak ada jaminan bahwa suatu individu bernilai *fitness* tertinggi akan selalu dipilih (Suyanto, 2005).

## 7. Penggantian Populasi

Ukuran populasi menyatakan banyaknya individu dalam satu generasi. Setelah dilakukan proses seleksi, pindah silang dan mutasi, kemudian dilakukan penggantian populasi. Skema penggantian populasi ini disebut *generational replacement*, yang berarti semua individu (misal  $N$  individu dalam satu populasi) dari suatu generasi digantikan sekaligus oleh  $N$  individu baru hasil pindah silang dan mutasi (Suyanto, 2005).

### 2.6.3 Parameter Algoritma Genetika

Terdapat beberapa parameter dari algoritma genetika yang perlu diberi nilai, yaitu ukuran populasi, jumlah generasi, probabilitas *crossover*, dan probabilitas mutasi.

#### 1. Ukuran Populasi

Ukuran populasi atau *pop\_size*, merupakan parameter yang berfungsi untuk menentukan jumlah individu yang berada pada populasi (untuk setiap generasi). Semakin banyak dan semakin beragam individu dalam populasi akan memberikan peluang lebih besar untuk menemukan individu yang mendekati sempurna. Jika individu yang ada terlalu sedikit, algoritma genetika hanya memiliki sedikit kemungkinan untuk melakukan *crossover* dan hanya sebagian kecil dari *search space* yang akan ditelusuri. Namun jika terdapat terlalu banyak individu, maka *performance* dari algoritma genetika juga akan menjadi lambat. Penelitian menunjukkan bahwa pada batasan tertentu, meningkatkan ukuran populasi akan menjadi sangat tidak berguna, karena pemecahan masalah tidak akan berjalan lebih cepat (Obitko, 1998).

#### 2. Jumlah Generasi

Jumlah generasi mempunyai andil yang besar dalam menemukan individu yang lebih baik, karena semakin besar generasi maka individu yang dihasilkan akan semakin baik dan sempurna. Tetapi tidak berarti semakin besar jumlah generasi maka individu yang dihasilkan selalu lebih baik karena ada suatu saat nilai *fitness* semua individu menjadi sama.

### 3. Probabilitas *Crossover* ( $P_c$ )

Parameter ini berfungsi untuk menentukan seberapa sering proses *crossover* dilakukan. Semakin besar probabilitas *crossover* akan memungkinkan pencapaian alternatif solusi yang lebih bervariasi dan mengurangi kemungkinan menghasilkan nilai optimum yang tidak dikehendaki. Tetapi bila nilai ini terlalu tinggi akan mengakibatkan pemborosan waktu untuk melakukan perhitungan di daerah solusi yang tidak menjanjikan hasil yang optimal. Nilai probabilitas *crossover* yang sering dipakai berkisar antara 0.65 sampai 0.95 (Obitko, 1998).

### 4. Probabilitas Mutasi ( $P_m$ )

Parameter ini menunjukkan seberapa sering sebuah individu akan dimutasi. Jika tidak terdapat mutasi, maka semua anak akan berasal dari proses *crossover* tanpa adanya perubahan. Jika terdapat proses mutasi, maka sebagian dari individu tersebut akan berubah. Nilai probabilitas mutasi yang terlalu rendah dapat mengakibatkan gen-gen yang berpotensi tidak dicoba. Sebaliknya jika terlalu tinggi, dapat menghilangkan sifat kemiripan dengan induknya, sehingga proses pencarian bisa salah dan menjauh dari daerah solusi yang diinginkan. Nilai probabilitas mutasi yang sering dipakai berkisar antara 0.01 sampai 0.1 (Obitko, 1998).

#### 2.6.4 Kekonvergenan Algoritma Genetika

Algoritma genetika berhenti pada suatu kondisi jika memenuhi kriteria yang telah ditentukan sebelumnya. Dalam hal ini, proses berhenti setelah iterasi berulang sebanyak  $n$  kali. Konvergen merupakan kondisi yang dicapai pada saat populasi kehilangan keanekaragaman, artinya sebagian besar individu dalam populasi mempunyai karakteristik yang sama. Algoritma genetika dapat mencapai kondisi konvergen meskipun tidak dapat dipastikan bahwa solusi yang diperoleh merupakan solusi optimal. Kesalahan yang sering terjadi adalah populasi memperoleh solusi yang belum optimal, sementara kriteria terminasi sudah dipenuhi (Setiawan, 2003).

## **2.7 HIV (*Human Immunodeficiency Virus*)**

### **2.7.1 Infeksi HIV dan AIDS**

HIV adalah singkatan dari *Human Immunodeficiency Virus*, virus yang menyebabkan rusaknya / melemahnya sistem kekebalan tubuh manusia (Anonimus<sup>3</sup>, 2008). AIDS adalah sekumpulan gejala dan infeksi (sindrom) yang timbul karena rusaknya sistem kekebalan tubuh manusia akibat infeksi virus HIV atau infeksi virus-virus lain sejenis yang menyerang spesies lainnya (SIV, FIV, dan lain-lain).

AIDS merupakan bentuk terparah akibat infeksi HIV. HIV adalah retrovirus yang biasanya menyerang organ-organ vital sistem kekebalan manusia, seperti sel T CD4<sup>+</sup> (sejenis sel T), makrofag, dan sel dendritik. HIV merusak sel T CD4<sup>+</sup> secara langsung dan tidak langsung, padahal sel T CD4<sup>+</sup> dibutuhkan agar sistem kekebalan tubuh dapat berfungsi baik. Jika HIV telah membunuh sel T CD4<sup>+</sup> hingga jumlahnya menyusut kurang dari 200 per mikroliter ( $\mu\text{L}$ ) darah, maka kekebalan selular akan hilang, dan kondisi ini disebut AIDS. Infeksi akut HIV akan berlanjut menjadi infeksi laten klinis, kemudian timbul gejala infeksi HIV awal, dan akhirnya AIDS, yang diidentifikasi dengan memeriksa jumlah sel T CD4<sup>+</sup> di dalam darah serta adanya infeksi tertentu (Anonimus<sup>3</sup>, 2008).

### **2.7.2 Cara Kerja Infeksi HIV**

Tanpa terapi antiretrovirus, rata-rata lamanya perkembangan infeksi HIV menjadi AIDS adalah 9 sampai 10 tahun, dan rata-rata waktu hidup setelah mengalami AIDS hanya sekitar 9,2 bulan. Namun demikian, laju perkembangan penyakit ini pada setiap orang sangat bervariasi, yaitu dari dua minggu sampai 20 tahun. Banyak faktor yang mempengaruhinya, di antaranya ialah kekuatan tubuh untuk bertahan melawan HIV (seperti fungsi kekebalan tubuh) dari orang yang terinfeksi (Anonimus<sup>3</sup>, 2008).

Sistem kekebalan dapat mempertahankan tubuh terhadap infeksi. Sistem ini terdiri dari banyak jenis sel. Dari sel-sel tersebut, sel T CD4<sup>+</sup> (*T-helper*) sangat krusial karena ia mengkoordinasi semua sistem kekebalan sel lainnya. Sel *T-helper* memiliki protein pada permukaannya yang disebut CD4.

HIV masuk ke dalam darah dan mendekati sel *T-helper* dengan melekatkan dirinya pada protein CD4. Sekali ia berada di dalam, materi viral (jumlah virus dalam tubuh penderita) turunan yang

disebut RNA (*ribonucleic acid*) berubah menjadi viral DNA (*deoxyribonucleic acid*) dengan suatu enzim yang disebut *reverse transcriptase*. Viral DNA tersebut menjadi bagian dari DNA manusia, yang mulai menghasilkan virus-virus HIV daripada menghasilkan lebih banyak sel jenisnya.

Selain enzim *reverse transcriptase*, terdapat enzim protease yang mengatur viral kimia untuk membentuk virus-virus yang baru. Virus-virus baru tersebut keluar dari sel tubuh dan bergerak bebas dalam aliran darah, dan berhasil menulari lebih banyak sel. Ini adalah sebuah proses yang sedikit demi sedikit merusak sistem kekebalan tubuh dan membuat tubuh mudah diserang oleh infeksi dan penyakit-penyakit yang lain. Dibutuhkan waktu untuk menularkan virus tersebut dari orang ke orang.

Respons tubuh secara alamiah terhadap suatu infeksi adalah untuk melawan sel-sel yang terinfeksi dan menggantikan sel-sel yang telah hilang. Respons tersebut mendorong virus untuk menghasilkan kembali dirinya. Jumlah normal dari sel-sel T CD4<sup>+</sup> pada seseorang yang sehat adalah 800-1200 sel /  $\mu$ L kubik darah. Ketika seorang pengidap HIV yang sel-sel T CD4<sup>+</sup> terhitung di bawah 200 sel /  $\mu$ L kubik darah, dia menjadi semakin mudah diserang oleh infeksi-infeksi oportunistik. Pada seseorang dengan sistem kekebalan yang sehat, infeksi-infeksi tersebut biasanya tidak mengancam hidup mereka, tetapi bagi seorang pengidap HIV hal tersebut dapat menjadi fatal. Tanpa perawatan, *viral load*, yang menunjukkan jumlah relatif dari virus bebas bergerak di dalam plasma darah, akan meningkat mencapai titik dimana tubuh tidak akan mampu melawannya (Anonimus<sup>2</sup>, 2007).

### 2.7.3 Cara Penularan HIV

HIV berada terutama dalam cairan tubuh manusia, seperti darah, cairan sperma, cairan vagina, dan air susu ibu. Sedangkan cairan yang tidak berpotensi untuk menularkan virus HIV adalah cairan keringat, air liur, air mata dan lain-lain (Anonimus<sup>1</sup>, 2006).

Jalan utama penularan HIV adalah kontak seksual (homoseksual atau heteroseksual), tranmisi jarum suntik dan alat kesehatan lain, tranmisi perinatal (dari ibu ke anak dalam persalinan), tranmisi darah dan produk darah serta tranmisi dalam pelayanan kesehatan yaitu pada pekerja rumah sakit yang melakukan kontak

dengan darah atau cairan tubuh pasien yang terinfeksi HIV (Pintauli, 2000).

UNIVERSITAS BRAWIJAYA



UNIVERSITAS BRAWIJAYA



## BAB III METODOLOGI PENELITIAN

### 3.1 Konstruksi Model Sistem Imun HIV

Model sistem imun HIV / AIDS melibatkan tiga variabel, yang terdiri atas populasi sel T CD4<sup>+</sup> yang tidak terinfeksi ( $X_1$ ), sel T CD4<sup>+</sup> yang terinfeksi ( $X_2$ ), dan virus yang hidup bebas di dalam darah ( $X_3$ ).

### 3.2 Mengestimasi Parameter dengan Algoritma Genetika

Data dibangkitkan dari model sistem imun HIV dengan memberikan nilai parameter awal  $S=1$ ,  $d=0.8$ ,  $\beta=1$ ,  $\mu_1=0.8$ ,  $k=1$ ,  $\mu_2=0.01078$ ,  $X_1=1$ ,  $X_2=0.2$ , dan  $X_3=0.8$ , untuk  $t=0$  sampai dengan  $t=1$  dan  $\Delta t=0.1$ . Estimasi parameter model dilakukan dengan mencocokkan data terhadap penyelesaian persamaan diferensial menggunakan metode kuadrat terkecil (*least squares method*), sehingga fungsi objektif yang akan diminimumkan adalah:

$$error = \sum_{i=1}^3 \sum_{j=1}^n [y_{i,j} - y_{i,j}^*]^2,$$

dengan  $n$  adalah banyaknya data,  $y_{i,j}$  adalah komponen data dan  $y_{i,j}^*$  adalah komponen penyelesaian persamaan diferensial.

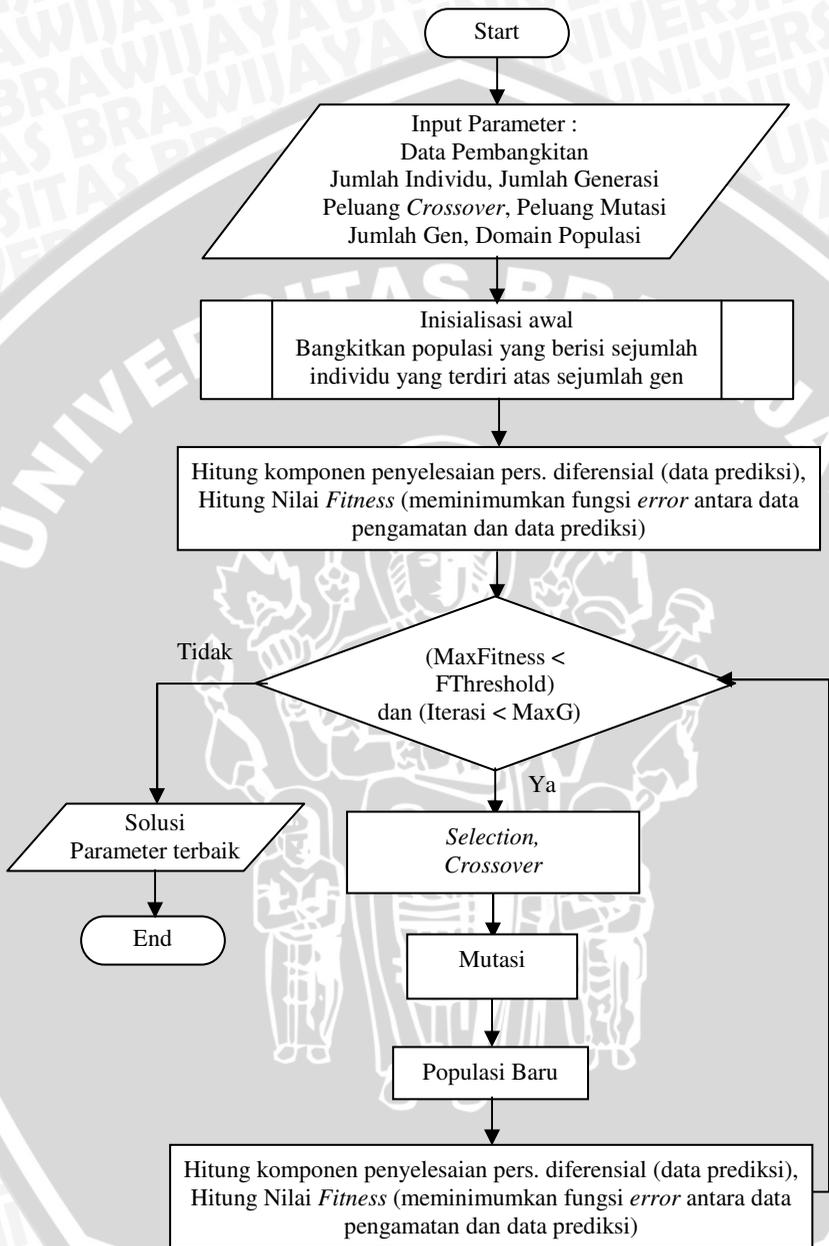
Untuk meminimumkan fungsi objektif digunakan metode optimasi algoritma genetika. Kriteria pemberhentian (terminasi) yang digunakan adalah jika *fitness* tertinggi (MaxFitness)  $\geq$  fungsi *threshold* (FThreshold) dan iterasi telah mencapai maksimum generasi (MaxG), dengan fungsi *threshold* adalah  $\frac{1}{a}$  dan  $a$  bilangan kecil. Model yang memuat parameter yang diperoleh dari algoritma genetika diselesaikan secara numerik menggunakan metode Runge-Kutta orde empat.

Langkah awal yang sangat penting dilakukan dari algoritma genetika adalah menentukan masukan semua parameter algoritma genetika yaitu jumlah individu, jumlah generasi, probabilitas pindah silang, probabilitas mutasi, jumlah gen dan domain populasi.

Secara garis besar rancangan penyelesaian permasalahan estimasi parameter dengan algoritma genetika dapat digambarkan dengan diagram alir (*flowchart*) pada Gambar 3.1.

UNIVERSITAS BRAWIJAYA





**Gambar 3.1** Diagram Alir Estimasi Parameter Menggunakan Algoritma Genetika

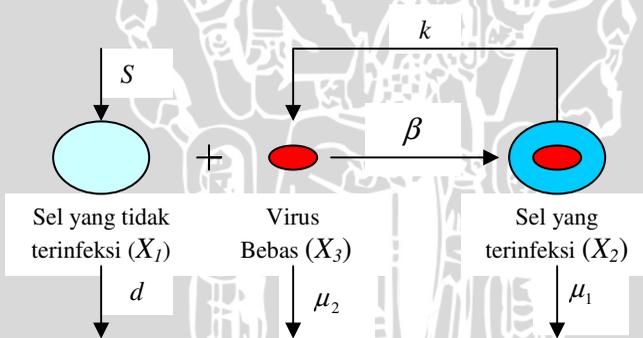
Langkah-langkah spesifik algoritma genetika yang digunakan antara lain:

1. Tentukan semua parameter awal algoritma genetika yang dibutuhkan yaitu:
  - Jumlah individu
  - Jumlah generasi
  - Probabilitas *crossover*
  - Probabilitas mutasi
  - Jumlah gen
  - Domain populasi
2. Lakukan proses inisialisasi awal yaitu dengan membangkitkan bilangan acak yang disusun menjadi sebuah individu. Sejumlah individu yang terbentuk, dimasukkan ke dalam populasi awal.
3. Hitung nilai prediksi dengan mencari solusi numerik sistem persamaan diferensial dengan Runge-Kutta orde empat. Kemudian dihitung nilai *fitness* dengan meminimumkan fungsi objektif.
4. Lakukan seleksi untuk menentukan individu-individu yang akan *dicrossover* untuk menghasilkan individu baru. Selanjutnya dipilih individu dengan *fitness* yang lebih baik dari individu-individu yang sudah ada untuk menggantikannya.
5. Lakukan proses seleksi untuk memilih individu yang akan mengalami proses mutasi. Untuk setiap individu baru yang dihasilkan akan dilakukan pengecekan *fitness* untuk menentukan apakah individu baru yang dihasilkan pada proses mutasi bisa menggantikan individu yang sudah ada dalam populasi.
6. Proses *crossover* dan mutasi akan terus dilakukan sampai kriteria pemberhentian berlaku. Kriteria yang digunakan adalah berhenti jika *fitness* terbesar sudah mencapai  $F_{\text{Threshold}}$  dan sampai batas generasi yang ditentukan.

## BAB IV HASIL DAN PEMBAHASAN

### 4.1 Struktur Model

Dalam infeksi HIV, banyak model matematika yang telah dikembangkan untuk menggambarkan pertumbuhan sel T CD4<sup>+</sup> dalam waktu tertentu, interaksi antara HIV dan infeksi oportunistik yang lain, kemunculan obat yang resisten terhadap virus, dan sebagainya. Berdasarkan studi yang dilakukan oleh Nowak dan Bangham tahun 1996, telah dirumuskan suatu model paling sederhana pada dinamika penyebaran virus HIV. Tampak pada Gambar 4.1 bahwa dalam model terbagi menjadi 3 variabel, yang terdiri atas populasi sel T CD4<sup>+</sup> yang tidak terinfeksi ( $X_1$ ), sel T CD4<sup>+</sup> yang terinfeksi ( $X_2$ ), dan virus yang hidup bebas di dalam darah ( $X_3$ ). Diasumsikan bahwa sel T CD4<sup>+</sup> yang tidak terinfeksi bertemu dengan virus bebas dalam darah dan menghasilkan sel T CD4<sup>+</sup> yang terinfeksi.



**Gambar 4.1** Diagram Interaksi Virus HIV

dengan:

$X_1(t)$  = jumlah sel T CD4<sup>+</sup> yang tidak terinfeksi pada waktu  $t$

$X_2(t)$  = jumlah sel T CD4<sup>+</sup> yang terinfeksi pada waktu  $t$

$X_3(t)$  = jumlah virus bebas dalam darah pada waktu  $t$

$S$  = laju produksi sel T CD4<sup>+</sup> yang tidak terinfeksi

$d$  = laju kematian sel T CD4<sup>+</sup> yang tidak terinfeksi

$\beta$  = laju infeksi pada sel T CD4<sup>+</sup> oleh virus bebas

$\mu_1$  = laju kematian sel T CD4<sup>+</sup> yang terinfeksi

$\mu_2$  = laju kematian virus bebas karena antibodi

$k$  = laju produksi virus bebas

dimana  $S, d, \beta, \mu_1, \mu_2, k$  adalah parameter tidak negatif.

## 4.2 Konstruksi Model

Berdasarkan Gambar 4.1, misalkan  $X_1(t)$ ,  $X_2(t)$ , dan  $X_3(t)$  berturut-turut menyatakan jumlah populasi sel T CD4<sup>+</sup> yang tidak terinfeksi, sel T CD4<sup>+</sup> yang terinfeksi dan virus yang hidup bebas dalam darah pada saat  $t$ . Laju perubahan dari ketiga populasi tersebut diperoleh dengan asumsi berikut.

1. Misalkan  $S$  adalah banyaknya sel T CD4<sup>+</sup> baru yang dihasilkan oleh *thymus*, sumsum tulang ataupun sumber lainnya. Laju perubahan  $X_1$  pada waktu  $t$  berbanding lurus dengan banyak sel baru yang dihasilkan:

$$\frac{dX_1(t)}{dt} = S. \quad (4.1)$$

2. Sel T CD4<sup>+</sup> mempunyai masa hidup terbatas, misalkan laju kematian sel T CD4<sup>+</sup> adalah  $d$ , laju perubahan  $X_1$  akibat kematian alami adalah:

$$\frac{dX_1(t)}{dt} = -d X_1(t). \quad (4.2)$$

3. Jumlah sel T CD4<sup>+</sup> yang terinfeksi virus ditentukan oleh banyaknya pertemuan yang terjadi antara sel T CD4<sup>+</sup> dengan virus. Misalkan virus dalam darah menginfeksi sel T CD4<sup>+</sup> yang tidak terinfeksi dengan laju infeksi  $\beta$ , sehingga laju perubahan sel T CD4<sup>+</sup> yang tidak terinfeksi berbanding lurus dengan jumlah sel T CD4<sup>+</sup> tidak terinfeksi yang melakukan kontak dengan virus, yaitu:

$$\frac{dX_1(t)}{dt} = -\beta X_1(t) X_3(t). \quad (4.3)$$

Dari persamaan (4.1)-(4.3) dapat dibuat model laju perubahan sel T CD4<sup>+</sup> terhadap waktu  $t$  yaitu:

$$\frac{dX_1(t)}{dt} = S - d X_1(t) - \beta X_1(t)X_3(t). \quad (4.4)$$

4. Berdasarkan persamaan (4.3) laju perubahan sel T CD4<sup>+</sup> yang terinfeksi adalah:

$$\frac{dX_2(t)}{dt} = \beta X_1(t)X_3(t). \quad (4.5)$$

5. Sel T CD4<sup>+</sup> yang terinfeksi akan mengalami kematian secara alami. Diasumsikan laju kematian konstan  $\mu_1$ , maka laju perubahan  $X_2$  karena kematian alami adalah:

$$\frac{dX_2(t)}{dt} = -\mu_1 X_2(t). \quad (4.6)$$

Dari persamaan (4.5)-(4.6) dapat dibuat model laju perubahan sel T CD4<sup>+</sup> yang terinfeksi terhadap waktu  $t$  adalah:

$$\frac{dX_2(t)}{dt} = \beta X_1(t)X_3(t) - \mu_1 X_2(t). \quad (4.7)$$

6. Sel T CD4<sup>+</sup> yang terinfeksi akan menghasilkan virus-virus baru. Misalkan  $k$  adalah tingkat produksi virus yang bebas di dalam darah, maka laju perubahan virus dalam darah akan berbanding lurus dengan sel T CD4<sup>+</sup> yang terinfeksi:

$$\frac{dX_3(t)}{dt} = kX_2(t). \quad (4.8)$$

7. Keberadaan virus menimbulkan respon imun yang spesifik (antibodi) untuk membunuh virus. Jika diasumsikan laju kematian konstan  $\mu_2$ , maka laju perubahan  $X_3$  adalah:

$$\frac{dX_3(t)}{dt} = -\mu_2 X_3(t). \quad (4.9)$$

Dari persamaan (4.8)-(4.9) dapat dibuat model laju perubahan virus terhadap waktu  $t$  yaitu:

$$\frac{dX_3(t)}{dt} = kX_2(t) - \mu_2 X_3(t). \quad (4.10)$$

Dengan demikian laju perubahan dari sel T CD4<sup>+</sup> yang tidak terinfeksi  $X_1(t)$ , sel T CD4<sup>+</sup> yang terinfeksi  $X_2(t)$  dan virus yang

hidup bebas dalam darah  $X_3(t)$  memenuhi sistem persamaan diferensial nonlinier berikut:

$$\begin{cases} \frac{dX_1(t)}{dt} = S - d X_1(t) - \beta X_1(t)X_3(t) \\ \frac{dX_2(t)}{dt} = \beta X_1(t)X_3(t) - \mu_1 X_2(t) \\ \frac{dX_3(t)}{dt} = kX_2(t) - \mu_2 X_3(t). \end{cases} \quad (4.11)$$

Selanjutnya akan dibahas mengenai penyelesaian secara numerik dan estimasi parameter model sistem imun HIV, seperti yang tampak pada persamaan (4.11), dimana  $S$ ,  $d$ ,  $\beta$ ,  $\mu_1$ ,  $\mu_2$ ,  $k$  adalah parameter positif.

### 4.3 Estimasi Parameter Model Sistem Imun HIV

#### 4.3.1 Solusi Numerik pada Kondisi Awal

Andaikan persamaan (4.11) mempunyai nilai awal  $S=1$ ,  $d=0.8$ ,  $\beta=1$ ,  $\mu_1=0.8$ ,  $\mu_2=0.01078$ ,  $k=1$  dan  $X_1=1$ ,  $X_2=0.2$ ,  $X_3=0.8$ , maka sistem persamaan dapat dicari solusi numeriknya dengan menggunakan metode Runge-Kutta orde empat, sehingga persamaan (4.11) dapat ditulis:

$$\begin{cases} \frac{dX_1(t)}{dt} = f_1(t, X_1, X_2, X_3) = 1 - 0.8X_1(t) - X_1(t)X_3(t) \\ \frac{dX_2(t)}{dt} = f_2(t, X_1, X_2, X_3) = X_1(t)X_3(t) - 0.8X_2(t) \\ \frac{dX_3(t)}{dt} = f_3(t, X_1, X_2, X_3) = X_2(t) - 0.01078X_3(t). \end{cases} \quad (4.12)$$

Misal  $t = [0,1]$  dan  $\Delta t = 0.1$ , dengan menggunakan rumus rekursif metode Runge-Kutta yang sesuai dengan persamaan (4.12) adalah sebagai berikut:

$$X_{n+1,i} = X_{n,i} + \frac{1}{6}(k_{1,i} + 2k_{2,i} + 2k_{3,i} + k_{4,i})$$

dimana,

$$k_{1,i} = \Delta t \times f_i(t_n, x_{n,1}, x_{n,2}, \dots, x_{n,m})$$

$$k_{2,i} = \Delta t \times f_i \left( t_n + \frac{h}{2}, x_{n,1} + \frac{1}{2} k_{1,1}, x_{n,2} + \frac{1}{2} k_{1,2}, \dots, x_{n,m} + \frac{1}{2} k_{1,m} \right)$$

$$k_{3,i} = \Delta t \times f_i \left( t_n + \frac{h}{2}, x_{n,1} + \frac{1}{2} k_{2,1}, x_{n,2} + \frac{1}{2} k_{2,2}, \dots, x_{n,m} + \frac{1}{2} k_{2,m} \right)$$

$$k_{4,i} = \Delta t \times f_i \left( t_n + h, x_{n,1} + k_{3,1}, x_{n,2} + k_{3,2}, \dots, x_{n,m} + k_{3,m} \right)$$

dengan  $n = 0, 1, \dots, 10$  dan  $i = 1, 2, 3$ .

Untuk proses iterasi pertama:

$$t_0 = 0, X_{0,1} = 1, X_{0,2} = 0.2, X_{0,3} = 0.8,$$

$$f_1(t_0, X_{0,1}, X_{0,2}, X_{0,3}) = 1 - 0.8 \times 1 - 1 \times 1 \times 0.8 = -0.6$$

$$f_2(t_0, X_{0,1}, X_{0,2}, X_{0,3}) = 1 \times 1 \times 0.8 - 0.8 \times 0.2 = 0.64$$

$$f_3(t_0, X_{0,1}, X_{0,2}, X_{0,3}) = 1 \times 0.2 - 0.01078 \times 0.8 = 0.191376$$

$$k_{1,1} = \Delta t \times f_1(t_0, X_{0,1}, X_{0,2}, X_{0,3}) = -0.06$$

$$k_{1,2} = \Delta t \times f_2(t_0, X_{0,1}, X_{0,2}, X_{0,3}) = 0.064$$

$$k_{1,3} = \Delta t \times f_3(t_0, X_{0,1}, X_{0,2}, X_{0,3}) = 0.0191376$$

$$k_{2,1} = \Delta t \times f_1 \left( t_0 + \frac{\Delta t}{2}, X_{0,1} + \frac{k_{1,1}}{2}, X_{0,2} + \frac{k_{1,2}}{2}, X_{0,3} + \frac{k_{1,3}}{2} \right) \\ = -0.0561282$$

$$k_{2,2} = \Delta t \times f_2 \left( t_0 + \frac{\Delta t}{2}, X_{0,1} + \frac{k_{1,1}}{2}, X_{0,2} + \frac{k_{1,2}}{2}, X_{0,3} + \frac{k_{1,3}}{2} \right) \\ = 0.05996817$$

$$k_{2,3} = \Delta t \times f_3 \left( t_0 + \frac{\Delta t}{2}, X_{0,1} + \frac{k_{1,1}}{2}, X_{0,2} + \frac{k_{1,2}}{2}, X_{0,3} + \frac{k_{1,3}}{2} \right) \\ = 0.02232728$$

$$k_{3,1} = \Delta t \times f_1 \left( t_0 + \frac{\Delta t}{2}, X_{0,1} + \frac{k_{2,1}}{2}, X_{0,2} + \frac{k_{2,2}}{2}, X_{0,3} + \frac{k_{2,3}}{2} \right) \\ = -0.05659477$$

$$k_{3,2} = \Delta t \times f_2 \left( t_0 + \frac{\Delta t}{2}, X_{0,1} + \frac{k_{2,1}}{2}, X_{0,2} + \frac{k_{2,2}}{2}, X_{0,3} + \frac{k_{2,3}}{2} \right) \\ = 0.06044117$$

$$k_{3,3} = \Delta t \times f_3 \left( t_0 + \frac{\Delta t}{2}, X_{0,1} + \frac{k_{2,1}}{2}, X_{0,2} + \frac{k_{2,2}}{2}, X_{0,3} + \frac{k_{2,3}}{2} \right) \\ = 0.02212398$$

$$k_{4,1} = \Delta t \times f_1(t_0 + \Delta t, X_{0,1} + k_{3,1}, X_{0,2} + k_{3,2}, X_{0,3} + k_{3,3})$$

$$= -0.05303207$$

$$k_{4,2} = \Delta t \times f_2(t_0 + \Delta t, X_{0,1} + k_{3,1}, X_{0,2} + k_{3,2}, X_{0,3} + k_{3,3}) \\ = 0.05672434$$

$$k_{4,3} = \Delta t \times f_3(t_0 + \Delta t, X_{0,1} + k_{3,1}, X_{0,2} + k_{3,2}, X_{0,3} + k_{3,3}) \\ = 0.02515787$$

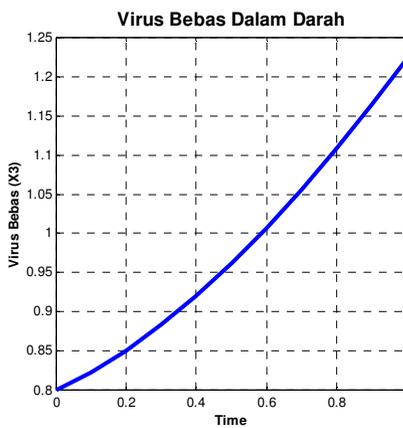
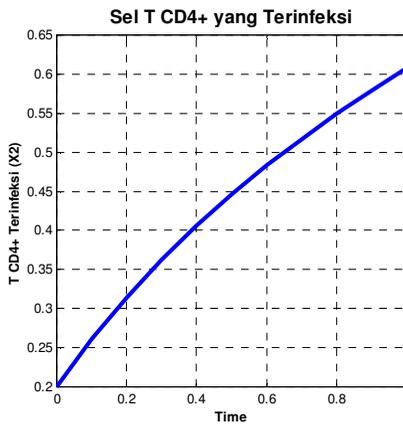
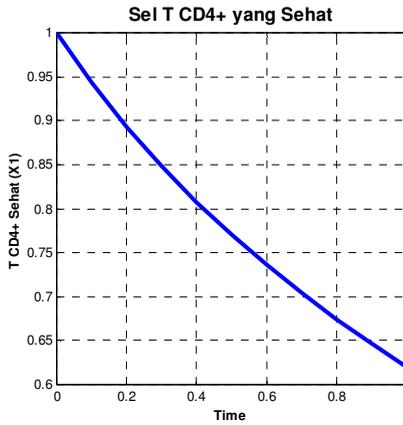
Maka diperoleh:

$$X_{1,1} = X_{0,1} + \frac{1}{6}(k_{1,1} + 2k_{2,1} + 2k_{3,1} + k_{4,1}) \\ = 1 + \frac{1}{6}(-0.06 + 2(-0.0561282) + 2(-0.05659477) + (-0.05303207)) \\ = 0.943587$$

$$X_{1,2} = X_{0,2} + \frac{1}{6}(k_{1,2} + 2k_{2,2} + 2k_{3,2} + k_{4,2}) \\ = 0.2 + \frac{1}{6}(0.064 + 2 \times 0.05996817 + 2 \times 0.06044117 + 0.05672434) \\ = 0.26025717$$

$$X_{1,3} = X_{0,3} + \frac{1}{6}(k_{1,3} + 2k_{2,3} + 2k_{3,3} + k_{4,3}) \\ = 0.8 + \frac{1}{6}(0.0191376 + 2 \times 0.02332728 + 2 \times 0.02212398 + 0.02515787) \\ = 0.82219967 .$$

Proses berulang hingga mencapai  $t=1$  atau proses ke-10. Kurva solusi dari persamaan (4.12) merupakan hubungan antara waktu ( $t$ ) dengan populasi sel T CD4<sup>+</sup> yang sehat ( $X_1$ ), sel T CD4<sup>+</sup> yang terinfeksi ( $X_2$ ), dan virus yang hidup bebas di dalam darah ( $X_3$ ) dapat dilihat pada Gambar 4.2.



**Gambar 4.2** Solusi Numerik ( $X_1$ ), ( $X_2$ ), dan ( $X_3$ ) Terhadap Waktu  $t$

### 4.3.2 Contoh Implementasi Algoritma Genetika

Pada subbab ini dijelaskan bagaimana implementasi algoritma genetika sebagai alat estimasi untuk mencari nilai parameter yang optimal (dengan *error* minimum). Fungsi objektif yang digunakan adalah jumlah kuadrat *error* (*least squares method*), dimana daerah interval  $S, d, \beta, \mu_1, k, \mu_2 \in [0,1]$ . Langkah-langkah yang dilakukan adalah:

#### 1. Inisialisasi Awal

Pada inisialisasi awal, pencarian nilai parameter-parameter optimum dimulai dengan membangkitkan sebuah populasi yang berisi sejumlah kromosom / individu. Masing-masing individu berisi sejumlah gen, yang mengkodekan informasi yang disimpan dalam individu. Individu diperoleh dengan membangkitkan sejumlah bilangan riil acak / *random* dalam interval  $[0,1]$ . Proses pembangkitan individu dilakukan gen per gen. Sebagai contoh, diberikan panjang individu (jumlah gen) = 18, jumlah individu dalam satu populasi = 10, jumlah parameter = 6, dengan setiap parameter dikodekan menjadi 3 gen. *Encoding* yang digunakan adalah *binary encoding*. Untuk lebih jelasnya, individu dalam satu populasi yang diperoleh dapat dilihat pada Tabel 4.1.

**Tabel 4.1** Jumlah Individu dalam Satu Populasi

Individu	$S$			$d$			$\beta$			$\mu_1$			$k$			$\mu_2$		
1	0	1	0	1	1	1	1	0	0	0	0	1	0	0	0	1	1	0
2	0	1	1	1	0	0	1	1	0	1	1	1	1	0	0	1	0	1
3	1	1	0	0	1	0	0	1	1	0	1	0	1	1	1	1	0	0
4	0	0	1	0	1	1	1	0	0	1	0	0	0	1	0	0	0	0
5	1	0	1	1	1	0	0	0	1	0	1	1	0	0	1	1	1	1
6	1	0	0	1	0	1	1	0	0	1	0	1	1	0	1	0	1	1
7	0	0	0	0	0	1	0	1	0	0	1	1	1	1	0	1	1	0
8	1	1	1	0	1	0	0	1	1	0	1	1	0	1	1	1	1	0
9	0	1	0	1	1	1	1	0	1	1	1	0	1	0	0	1	0	1
10	1	1	0	0	0	0	1	1	1	0	1	0	1	1	0	1	0	1

Dari Tabel 4.1, dapat dihitung nilai parameter ( $S, d, \beta, \mu_1, k, \mu_2$ ) dengan mendekodekan individu yang berisi bilangan biner

menggunakan *binary encoding*. Sebagai contoh perhitungan parameter pada individu pertama yaitu:

$$x_j = r_b + (r_a - r_b)(g_{j,1} \times 2^{-1} + g_{j,2} \times 2^{-2} + \dots + g_{j,p} \times 2^{-p}),$$

dengan  $r_b = 0$ ,  $r_a = 1$  dan  $x_1 = S$ ,  $x_2 = d$ ,  $x_3 = \beta$ ,  $x_4 = \mu_1$ ,  $x_5 = k$ ,  $x_6 = \mu_2$ .

$$x_1 = S = 0 + (1 - 0)(0 \times 2^{-1} + 1 \times 2^{-2} + 0 \times 2^{-3}) = 0.25,$$

$$x_2 = d = 0 + (1 - 0)(1 \times 2^{-1} + 1 \times 2^{-2} + 1 \times 2^{-3}) = 0.875,$$

$$x_3 = \beta = 0 + (1 - 0)(1 \times 2^{-1} + 0 \times 2^{-2} + 0 \times 2^{-3}) = 0.5,$$

$$x_4 = \mu_1 = 0 + (1 - 0)(0 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3}) = 0.125,$$

$$x_5 = k = 0 + (1 - 0)(0 \times 2^{-1} + 0 \times 2^{-2} + 0 \times 2^{-3}) = 0,$$

$$x_6 = \mu_2 = 0 + (1 - 0)(1 \times 2^{-1} + 1 \times 2^{-2} + 0 \times 2^{-3}) = 0.75,$$

sehingga pada individu pertama diperoleh nilai parameter  $(S, d, \beta, \mu_1, k, \mu_2) = (0.25, 0.875, 0.5, 0.125, 0, 0.75)$ . Nilai parameter untuk masing-masing individu selengkapnya dapat dilihat pada Tabel 4.2.

**Tabel 4.2** Nilai Parameter dalam Satu Populasi

Individu	$S$	$d$	$\beta$	$\mu_1$	$k$	$\mu_2$
1	0.25	0.875	0.5	0.125	0	0.75
2	0.375	0.5	0.75	0.875	0.5	0.625
3	0.75	0.25	0.375	0.25	0.875	0.5
4	0.125	0.375	0.5	0.5	0.375	0
5	0.625	0.75	0.125	0.375	0.125	0.875
6	0.5	0.625	0.5	0.625	0.625	0.375
7	0	0.125	0.25	0.375	0.75	0.75
8	0.875	0.25	0.375	0.375	0.375	0.75
9	0.25	0.875	0.625	0.75	0.5	0.625
10	0.75	0	0.875	0.25	0.75	0.625

Berdasarkan nilai-nilai parameter yang diperoleh seperti pada Tabel 4.2, maka penyelesaian persamaan (4.11) untuk masing-masing individu dapat dicari secara numerik menggunakan metode Runge-Kutta orde empat dengan nilai awal  $X_1 = 1$ ,  $X_2 = 0.2$ ,  $X_3 = 0.8$ , yang hasilnya dapat dilihat pada Tabel 4.3.

**Tabel 4.3** Data Solusi Numerik dalam Satu Populasi

Individu ke-	$t$	$X_1$	$X_2$	$X_3$
1	0	1	0.2	0.8
	0.1	0.9051	0.2367	0.7422
	0.2	0.8237	0.2676	0.6886
	0.3	0.7536	0.2937	0.6388
	0.4	0.6931	0.3159	0.5927
	0.5	0.6408	0.3350	0.5498
	0.6	0.5954	0.3514	0.5101
	0.7	0.5559	0.3655	0.4732
	0.8	0.5215	0.3778	0.4390
	0.9	0.4915	0.3885	0.4073
2	1	0.4653	0.3979	0.3779
	0	1	0.2	0.8
	0.1	0.9327	0.2373	0.7622
	0.2	0.8746	0.2657	0.7282
	0.3	0.8243	0.2869	0.6975
	0.4	0.7806	0.3022	0.6696
	0.5	0.7425	0.3127	0.6439
	0.6	0.7093	0.3194	0.6203
	0.7	0.6804	0.3230	0.5983
	0.8	0.6551	0.3241	0.5777
3	0.9	0.6330	0.3232	0.5584
	1	0.6136	0.3207	0.5402
	0	1	0.2	0.8
	0.1	1.0199	0.2246	0.7791
	0.2	1.0394	0.2484	0.7613
	0.3	1.0585	0.2715	0.7464
	0.4	1.0772	0.2941	0.7341
	0.5	1.0953	0.3162	0.7244
	0.6	1.1129	0.3378	0.7170
	0.7	1.1298	0.3592	0.7118
4	0.8	1.1461	0.3802	0.7086
	0.9	1.1616	0.4011	0.7074
	1	1.1764	0.4218	0.7080
	0	1	0.2	0.8
	0.1	0.9373	0.2282	0.8080
	0.2	0.8789	0.2530	0.8171
	0.3	0.8244	0.2748	0.8270
	0.4	0.7738	0.2938	0.8376
	0.5	0.7265	0.3103	0.8490

	0.6	0.6825	0.3245	0.8609
	0.7	0.6415	0.3367	0.8733
	0.8	0.6033	0.3470	0.8861
	0.9	0.5677	0.3555	0.8993
	1	0.5346	0.3625	0.9128
5	0	1	0.2	0.8
	0.1	0.9788	0.2019	0.7354
	0.2	0.9601	0.2029	0.6762
	0.3	0.9435	0.2030	0.6220
	0.4	0.9288	0.2024	0.5723
	0.5	0.9158	0.2011	0.5268
	0.6	0.9043	0.1994	0.4850
	0.7	0.8941	0.1972	0.4468
	0.8	0.8852	0.1946	0.4117
	0.9	0.8772	0.1917	0.3795
	1	0.8702	0.1886	0.3500
	6	0	1	0.2
0.1		0.9505	0.2253	0.7836
0.2		0.9065	0.2465	0.7693
0.3		0.8672	0.2644	0.7567
0.4		0.8323	0.2793	0.7455
0.5		0.8010	0.2916	0.7356
0.6		0.7731	0.3018	0.7267
0.7		0.7481	0.3102	0.7188
0.8		0.7257	0.3169	0.7115
0.9		0.7056	0.3223	0.7050
1	0.6877	0.3264	0.6989	
7	0	1	0.2	0.8
	0.1	0.9686	0.2114	0.7571
	0.2	0.9391	0.2209	0.7180
	0.3	0.9113	0.2286	0.6824
	0.4	0.8851	0.2349	0.6498
	0.5	0.8604	0.2398	0.6200
	0.6	0.8369	0.2436	0.5927
	0.7	0.8146	0.2464	0.5676
	0.8	0.7934	0.2483	0.5445
	0.9	0.7731	0.2494	0.5231
	1	0.7538	0.2499	0.5034
8	0	1	0.2	0.8
	0.1	1.0326	0.2216	0.7498
	0.2	1.0653	0.2415	0.7040

	0.3	1.0980	0.2598	0.6622
	0.4	1.1308	0.2766	0.6241
	0.5	1.1636	0.29200	0.5892
	0.6	1.1962	0.3061	0.5575
	0.7	1.2287	0.3191	0.5285
	0.8	1.2611	0.3309	0.5021
	0.9	1.2932	0.3418	0.4779
	1	1.3250	0.3517	0.4559
	9	0	1	0.2
0.1		0.8960	0.2300	0.7620
0.2		0.8070	0.2515	0.7275
0.3		0.7306	0.2662	0.6960
0.4		0.6650	0.2756	0.6670
0.5		0.6083	0.2806	0.6401
0.6		0.5594	0.2824	0.6150
0.7		0.5171	0.2815	0.5914
0.8		0.4804	0.2786	0.5691
0.9		0.4486	0.2740	0.5481
1	0.4210	0.2683	0.5280	
10	0	1	0.2	0.8
	0.1	1.0062	0.2630	0.7684
	0.2	1.0145	0.3224	0.7432
	0.3	1.0241	0.3790	0.7237
	0.4	1.0346	0.4333	0.7094
	0.5	1.0455	0.4859	0.6999
	0.6	1.0564	0.5372	0.6947
	0.7	1.0670	0.5876	0.6935
	0.8	1.0768	0.6375	0.6960
	0.9	1.0857	0.6870	0.7020
1	1.0934	0.7366	0.7113	

## 2. Evaluasi Individu

Evaluasi individu ini bertujuan untuk menghitung nilai *fitness* dari suatu individu, dimana fungsi evaluasi ini bergantung pada permasalahan yang akan diselesaikan. Dalam Skripsi ini, fungsi yang

akan diminimumkan adalah  $error = h = \sum_{i=1}^3 \sum_{j=1}^n [y_{i,j} - y_{i,j}^*]^2$ . Fungsi

*fitnessnya* adalah  $f = \frac{1}{(h + a)}$ , dengan bilangan kecil  $a = 0.1$ .

Dalam proses perhitungan nilai *error* diperlukan data (data pembangkitan dengan nilai awal) dan solusi persamaan diferensial secara numerik. Oleh karena itu, dibangkitkan sejumlah data dari sistem persamaan (4.11) dengan nilai awal  $S=1$ ,  $d=0.8$ ,  $\beta=1$ ,  $\mu_1=0.8$ ,  $\mu_2=0.01078$ ,  $k=1$  dan  $X_1=1$ ,  $X_2=0.2$ ,  $X_3=0.8$ , mulai  $t=0$  sampai dengan  $t=1$  dan  $\Delta t=0.1$ . Data hasil pembangkitan dapat dilihat pada Tabel 4.4.

**Tabel 4.4** Data yang Dibangkitkan dengan Nilai Awal

Waktu ( $t$ )	Sel T CD4 <sup>+</sup> Sehat ( $X_1$ )	Sel T CD4 <sup>+</sup> Terinfeksi ( $X_2$ )	Virus Bebas ( $X_3$ )
0	1.00	0.20	0.80
0.1	0.94	0.26	0.82
0.2	0.89	0.31	0.85
0.3	0.85	0.36	0.88
0.4	0.81	0.41	0.92
0.5	0.77	0.45	0.96
0.6	0.74	0.48	1.01
0.7	0.70	0.52	1.06
0.8	0.67	0.55	1.12
0.9	0.65	0.58	1.16
1	0.62	0.61	1.22

Sebagai contoh perhitungan *fitness* untuk individu pertama:

$$h = \text{Error}(S, d, \beta, \mu_1, k, \mu_2) = \sum_{i=1}^3 \sum_{j=1}^{11} [y_{i,j} - y_{i,j}^*]^2$$

$$\begin{aligned}
 h &= (1.00-1.00)^2 + (0.94-0.9051)^2 + (0.89-0.8237)^2 + (0.85-0.7536)^2 + \\
 & (0.81-0.6931)^2 + (0.77-0.6408)^2 + (0.74-0.5954)^2 + (0.70-0.5559)^2 + \\
 & (0.67-0.5215)^2 + (0.65-0.4915)^2 + (0.62-0.4653)^2 + (0.20-0.20)^2 + \\
 & (0.26-0.2367)^2 + (0.31-0.2676)^2 + (0.36-0.2937)^2 + (0.41-0.3159)^2 + \\
 & (0.45-0.3350)^2 + (0.48-0.3514)^2 + (0.52-0.3655)^2 + (0.55-0.3778)^2 + \\
 & (0.58-0.3885)^2 + (0.61-0.3979)^2 + (0.80-0.80)^2 + (0.82-0.7422)^2 + \\
 & (0.85-0.6886)^2 + (0.88-0.6388)^2 + (0.92-0.5927)^2 + (0.96-0.5498)^2 + \\
 & (1.01-0.5101)^2 + (1.06-0.4732)^2 + (1.12-0.4390)^2 + (1.16-0.4073)^2 + \\
 & (1.22-0.3779)^2 \\
 & = 3.0379
 \end{aligned}$$

$$fitness = \frac{1}{(3.0379 + 0.1)} = 0.31868 .$$

Dengan cara yang sama, diperoleh nilai *fitness* masing-masing individu dalam satu populasi seperti pada Tabel 4.5.

**Tabel 4.5** Nilai *Fitness* Setiap Individu dalam Satu Populasi

Individu	<i>Fitness</i>
1	0.31868
2	0.47499
3	0.38191
4	1.46686
5	0.24450
6	0.71643
7	0.37442
8	0.21170
9	0.39318
10	0.46112

Setelah diperoleh nilai *fitness*, dilakukan proses *elitisme* dengan cara membuat satu atau dua kopi dari individu yang bernilai *fitness* tertinggi. Salinan satu individu terbaik dari populasi diambil jika ukuran populasinya ganjil dan salinan dua individu terbaik dari populasi diambil jika ukuran populasinya genap. Hal ini dilakukan untuk menjaga agar individu bernilai *fitness* tertinggi tidak hilang selama evolusi. Dari Tabel 4.5 diperoleh nilai *fitness* tertinggi ( $f_{max}$ ) adalah 1.46686 dan *fitness* terendah ( $f_{min}$ ) adalah 0.21170, dengan indeks individu terbaik = 4.

### 3. *Linear Fitness Ranking*

Langkah selanjutnya untuk menghindari kecenderungan konvergen pada optimum lokal, dilakukan *linear fitness ranking* dengan cara penskalaan nilai *fitness* seperti pada persamaan (2.15), sehingga diperoleh nilai *fitness* baru yang lebih baik. Nilai *fitness* diurutkan terlebih dahulu sebelum dilakukan *linear fitness ranking* seperti pada Tabel 4.6.

**Tabel 4.6** Nilai *Fitness* Setelah Diurutkan

Urutan	Individu	<i>Fitness</i>
1	8	0.21170
2	5	0.24450
3	1	0.31868
4	7	0.37442
5	3	0.38191
6	9	0.39318
7	10	0.46112
8	2	0.47499
9	6	0.71643
10	4	1.46686

Sebagai contoh perhitungan *linear fitness ranking* untuk individu *fitness* terbaik:

$$f(i) = f_{\max} - (f_{\max} - f_{\min}) \left( \frac{R(i) - 1}{N - 1} \right)$$

$$f(10) = 1.46686 - (1.46686 - 0.21170) \left( \frac{1 - 1}{10 - 1} \right) = 1.46686 .$$

Dengan cara yang sama diperoleh penskalaan nilai-nilai *fitness* seperti pada Tabel 4.7.

**Tabel 4.7** *Linear Fitness Ranking*

Urutan	Individu	<i>Fitness</i>
1	8	0.21170
2	5	0.35116
3	1	0.49062
4	7	0.63009
5	3	0.76955
6	9	0.90901
7	10	1.04847
8	2	1.18794
9	6	1.32740
10	4	1.46686

#### 4. Seleksi Orang Tua

Seleksi orang tua dilakukan untuk memilih dua buah individu sebagai induk yang akan *dicrossover* secara proporsional sesuai dengan nilai *fitness*nya. Metode yang digunakan adalah *roulette*

*wheel selection*. Nilai *fitness* pada Tabel 4.7, dibuat interval nilai kumulatif berdasarkan persamaan (2.15)-(2.18). Sebagai contoh penggunaan metode seleksi *roulette wheel selection*, dapat dilihat pada Tabel 4.8.

**Tabel 4.8** Nilai Kumulatif pada *Roulette Wheel Selection*

Individu ( <i>i</i> )	Fitness ( <i>f<sub>i</sub></i> )	Probabilitas seleksi ( <i>p<sub>i</sub></i> )	Probabilitas kumulatif ( <i>q<sub>i</sub></i> )	Interval Nilai Kumulatif
1	0.21170	0.02522	0.02522	[0 , 0.02522]
2	0.35116	0.04184	0.06707	(0.02522, 0.06707]
3	0.49062	0.05846	0.12552	(0.06707, 0.12552]
4	0.63009	0.07508	0.20059	(0.12552, 0.20059]
5	0.76955	0.09169	0.29229	(0.20059, 0.29229]
6	0.90901	0.10831	0.40059	(0.29229, 0.40059]
7	1.04847	0.12492	0.52552	(0.40059, 0.52552]
8	1.18794	0.14154	0.66707	(0.52552, 0.66707]
9	1.32740	0.15816	0.82522	(0.66707, 0.82522]
10	1.46686	0.17478	1	(0.82522, 1]
<b>Total</b>	8.39280			

Sebuah individu akan terpilih jika bilangan acak yang dibangkitkan berada dalam interval kumulatifnya. Misalkan bilangan acak yang dibangkitkan adalah 0,4 maka individu 9 terpilih sebagai orang tua. Tetapi jika bilangan acak yang dibangkitkan 0,89 maka individu 4 yang terpilih sebagai orang tua.

### 5. Pindah Silang (*Crossover*)

Proses pindah silang dilakukan setelah memperoleh dua buah individu sebagai orang tua. Pindah silang yang digunakan adalah pindah silang satu titik potong (*one-point crossover*). Selanjutnya dibangkitkan satu bilangan acak untuk menentukan letak gen yang akan dipotong dengan rumus perhitungan sebagai berikut:

$$TP = 1 + \text{fix}(\text{rand} * (\text{JumlahGen} - 1)).$$

Misalkan bilangan acak yang dibangkitkan adalah 0.5 maka:

$$TP = 1 + \text{fix}(0.5 * (18 - 1)) = 9.$$

Proses *crossover* dapat dilihat pada Gambar 4.3.

Induk 1

Titik Potong

0	1	0	1	1	1	1	0	0	0	0	1	0	0	0	1	1	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Induk 2

0	1	1	1	0	0	1	1	0	1	1	1	1	0	0	1	0	1
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Anak 1

0	1	0	1	1	1	1	0	0	1	1	1	1	0	0	1	0	1
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Anak 2

0	1	1	1	0	0	1	1	0	0	0	1	0	0	0	1	1	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

**Gambar 4.3** Proses Pindah Silang

## 6. Mutasi (*Mutation*)

Mutasi bisa terjadi secara acak pada setiap gen dalam individu. Jika suatu bilangan acak  $[0,1]$  yang dibangkitkan kurang dari peluang mutasi, maka gen yang bersesuaian akan diganti dengan nilai kebalikannya (nilai 0 diubah menjadi 1 dan 1 diubah menjadi 0). Misalkan diberikan probabilitas mutasi adalah 0.1, maka setiap gen dibangkitkan bilangan acak. Andaikan bilangan acak yang dibangkitkan pada saat gen ke-8 adalah 0.01, maka gen ke-8 diubah nilainya dari 1 menjadi 0. Prosesnya dapat dilihat pada Gambar 4.4.

Individu 1

0	1	1	1	0	0	1	1	0	0	0	1	0	0	0	1	1	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

0.01 Bilangan acak

Setelah Mutasi

0	1	1	1	0	0	1	0	0	0	1	0	0	0	1	1	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

**Gambar 4.4** Proses Mutasi

Setelah melalui proses *crossover* dan mutasi, populasi baru terbentuk berdasarkan nilai *fitness* yang lebih baik dari populasi yang memuat individu sebelumnya. Proses dari algoritma genetika ini berhenti sampai *threshold* nilai *fitness* terpenuhi atau sampai iterasi

mencapai maksimum generasi. Hasil yang diperoleh adalah parameter yang memiliki nilai *fitness* tertinggi dan nilai *error* minimum.

### 4.3.3 Simulasi Estimasi Parameter dengan Menggunakan Data yang Dibangkitkan

Secara garis besar proses estimasi parameter model dilakukan dengan:

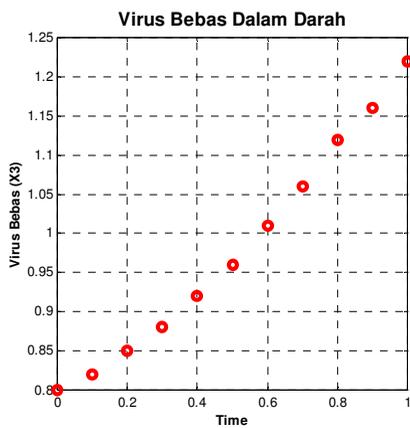
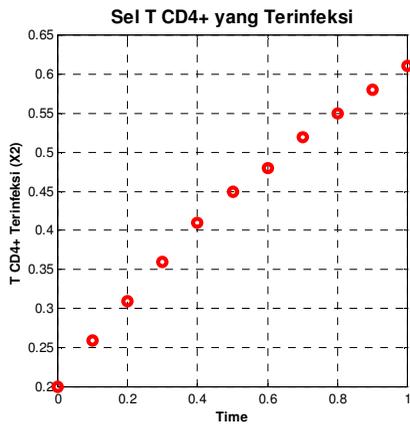
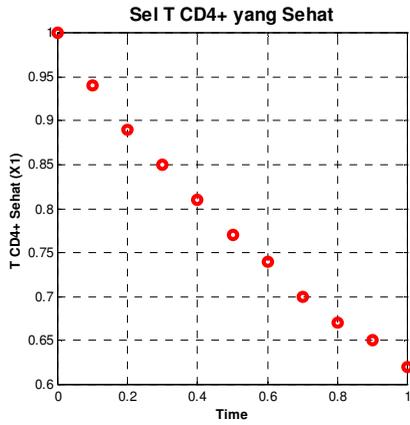
1. Menentukan parameter awal algoritma genetika yang meliputi jumlah individu, jumlah generasi, probabilitas *crossover*, probabilitas mutasi, jumlah gen, domain populasi (batas atas dan bawah solusi).
2. Membangkitkan populasi awal yang berisi parameter-parameter model yang akan diestimasi yaitu  $S, d, \beta, \mu_1, k, \mu_2$ .
3. Mencari solusi numerik model dengan metode Runge-Kutta orde 4 berdasar pada parameter yang diperoleh dari estimasi.
4. Menghitung nilai fungsi objektif yang berupa *error* antara data dengan data prediksi (penyelesaian sistem persamaan diferensial secara numerik dengan Runge-Kutta) dengan metode kuadrat terkecil,

$$h = \text{error}(S, d, \beta, \mu_1, k, \mu_2) = \sum_{i=1}^3 \sum_{j=1}^n [y_{i,j} - y_{i,j}^*]^2.$$

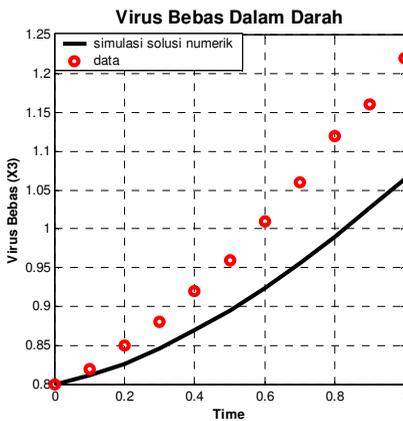
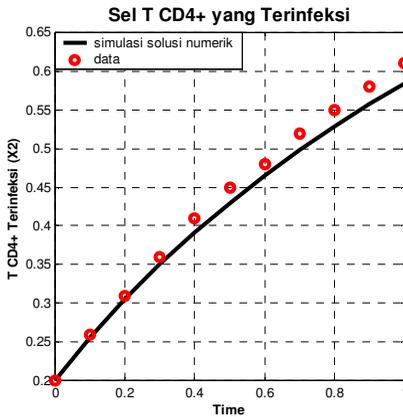
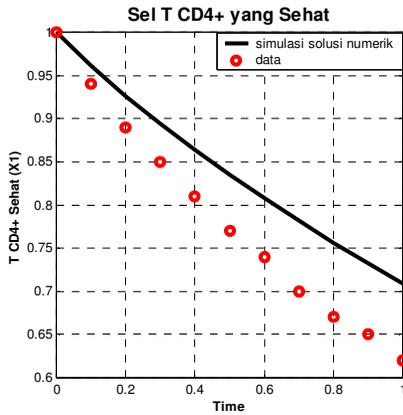
5. Meminimumkan fungsi  $h$  menggunakan algoritma genetika.

Data yang digunakan ada pada Tabel 4.4, yang solusi numeriknya dapat digambarkan seperti pada Gambar 4.5. Berdasarkan data, dilakukan proses estimasi dengan menggunakan algoritma genetika, dengan nilai parameter  $S, d, \beta, \mu_1, k, \mu_2$  terletak pada interval  $[0,1]$ , ukuran populasi sebesar 50 individu, jumlah generasi sebesar 50, probabilitas *crossover* 0.80, probabilitas mutasi 0.01, dan panjang individu sebanyak 60 gen, yang setiap parameter dikodekan menjadi 10 gen.

Dengan 1 kali percobaan, hasil estimasi yang diperoleh pada saat generasi ke-1 adalah nilai *fitness* sebesar 4,2917, dengan parameter  $S = 0.7754, d = 0.4492, \beta = 0.9092, \mu_1 = 0.7285, k = 0.8584, \mu_2 = 0.1035$  dan nilai minimum (*error*) sebesar 0.1330. Pada Gambar 4.6 ditampilkan solusi numerik yang dihasilkan dibandingkan terhadap data.

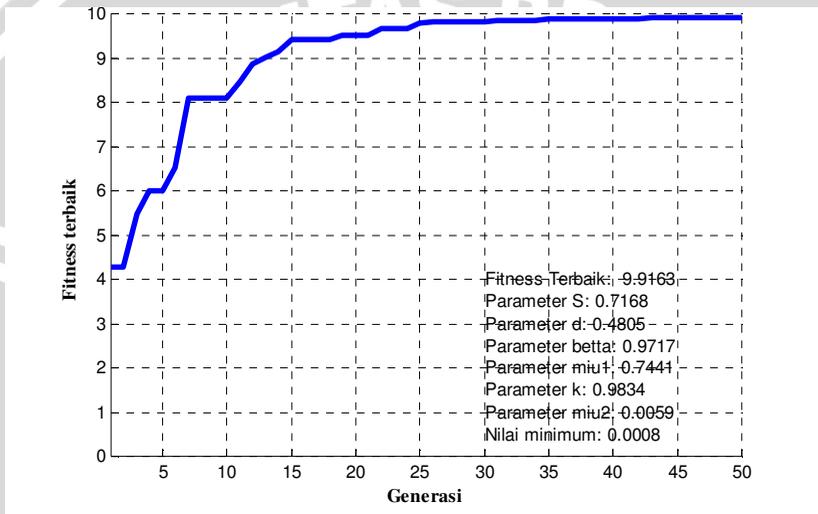


**Gambar 4.5** Data yang Dibangkitkan dengan Nilai Parameter Awal

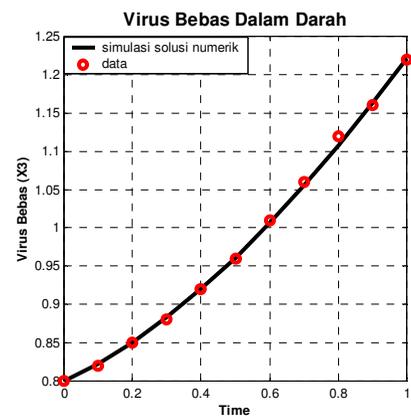
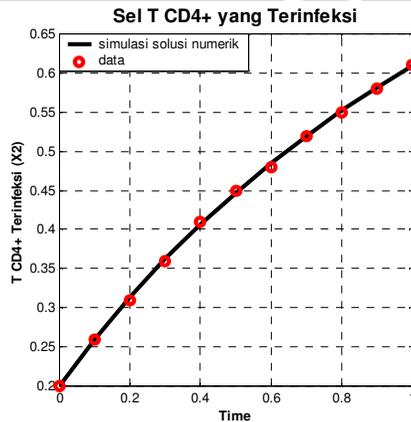
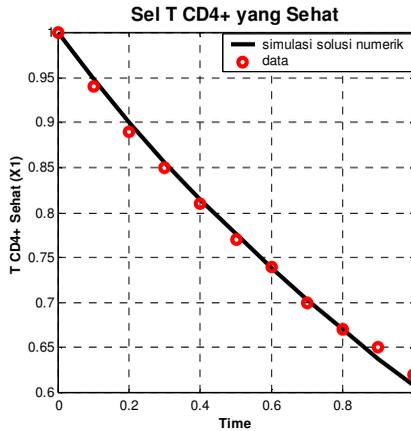


**Gambar 4.6** Perbandingan antara Data dan Solusi Numerik pada Generasi ke-1

Setelah mencapai generasi ke-50, seperti pada Gambar 4.7 dapat dijelaskan bahwa *fitness* terbaik yang dicapai adalah sebesar 9.9163. Nilai parameter yang diperoleh adalah  $S = 0.7168$ ,  $d = 0.4805$ ,  $\beta = 0.9717$ ,  $\mu_1 = 0.7441$ ,  $k = 0.9834$ ,  $\mu_2 = 0.0059$ , dan nilai minimum (*error*) sebesar 0.0008. Pada Gambar 4.8 ditampilkan perbandingan antara data terhadap solusi persamaan diferensial secara numerik dengan hasil estimasi parameter yang diperoleh.



**Gambar 4.7** Estimasi Parameter Menggunakan Algoritma Genetika



**Gambar 4.8** Perbandingan antara Data dan Solusi Numerik pada Generasi ke-50

Berikut ini simulasi numerik dilakukan dengan percobaan sebanyak 1 kali untuk masing-masing simulasi, dengan menggunakan probabilitas *crossover* ( $p_c$ ) dan ukuran populasi (*pop\_size*) yang berbeda-beda, dengan jumlah generasi sebesar 50, probabilitas mutasi ( $p_m$ ) adalah 0.01, dan jumlah gen sebesar 60. Hasil estimasi berupa nilai *fitness* dan *error* disajikan seperti pada Tabel 4.9, dan hasil selengkapnya dapat dilihat pada bagian Lampiran 1.

**Tabel 4.9** Hasil Estimasi untuk 1 Kali Percobaan pada Berbagai Ukuran Populasi dan Probabilitas *Crossover*

Ukuran Populasi	Probabilitas <i>crossover</i>	Probabilitas mutasi	<i>Fitness</i>	Nilai <i>Error</i>
50	0.65	0.01	9.7289	0.0028
50	0.75		9.9059	0.0010
50	0.85		9.9383	0.0006
50	0.95		9.9236	0.0008
100	0.65		9.7283	0.0028
100	0.75		9.8229	0.0018
100	0.85		9.8121	0.0019
<b>100</b>	<b>0.95</b>		<b>9.9588</b>	<b>0.0004</b>
200	0.65		9.9337	0.0007
200	0.75		9.9004	0.0010
200	0.85		9.8943	0.0011
200	0.95		9.8541	0.0015

Tampak pada Tabel 4.9, bahwa dengan ukuran populasi sebanyak 100 individu dan probabilitas *crossover* sebesar 0.95 merupakan nilai yang paling optimal (*error* terkecil) dengan nilai *fitness* sebesar 9.9588, dan nilai *error* sebesar 0.0004. Dalam hal ini probabilitas *crossover* yang digunakan adalah antara 0.65-0.95.

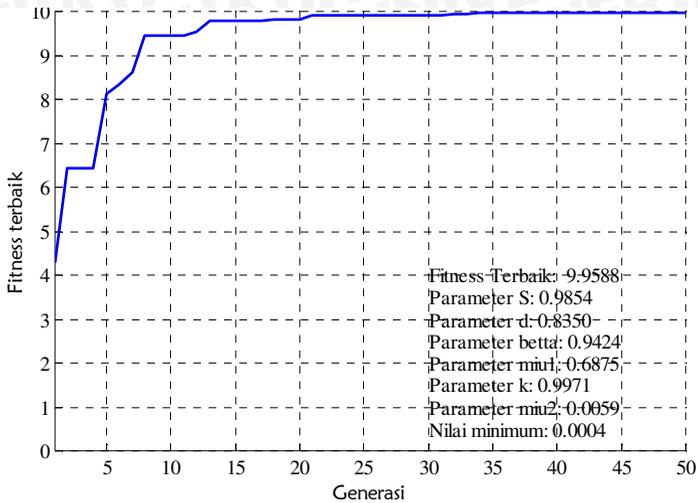
Selanjutnya juga dilakukan simulasi dengan melakukan percobaan sebanyak 1 kali untuk masing-masing simulasi, menggunakan probabilitas mutasi ( $p_m$ ) dan ukuran populasi (*pop\_size*) yang berbeda-beda, dengan jumlah generasi sebesar 50, probabilitas *crossover* adalah 0.95, jumlah gen sebesar 60. Hasil estimasi diberikan seperti pada Tabel 4.10.

**Tabel 4.10** Hasil Estimasi untuk 1 Kali Percobaan pada Berbagai Ukuran Populasi dan Probabilitas Mutasi

Ukuran Populasi	Probabilitas mutasi	Probabilitas <i>crossover</i>	<i>Fitness</i>	Nilai <i>Error</i>
50	0.01	0.95	9.9236	0.0008
50	0.03		9.5947	0.0042
50	0.05		9.7525	0.0025
50	0.08		9.6139	0.0040
50	0.1		9.7459	0.0026
<b>100</b>	<b>0.01</b>		<b>9.9588</b>	<b>0.0004</b>
100	0.03		9.9161	0.0008
100	0.05		9.7915	0.0021
100	0.08		9.9428	0.0006
100	0.1		9.6917	0.0032
200	0.01		9.8541	0.0015
200	0.03		9.9584	0.0004
200	0.05		9.9080	0.0009
200	0.08		9.5934	0.0042
200	0.1		9.7022	0.0031

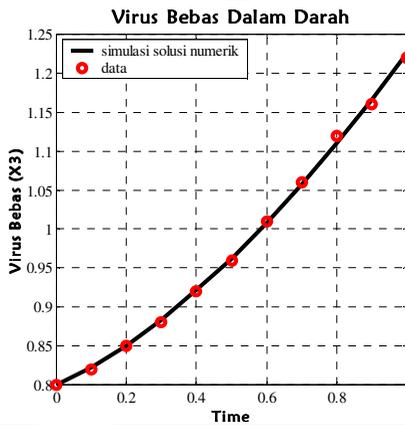
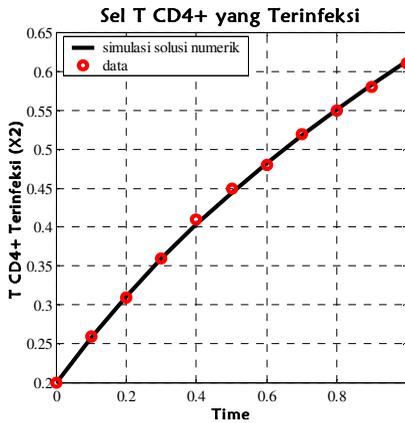
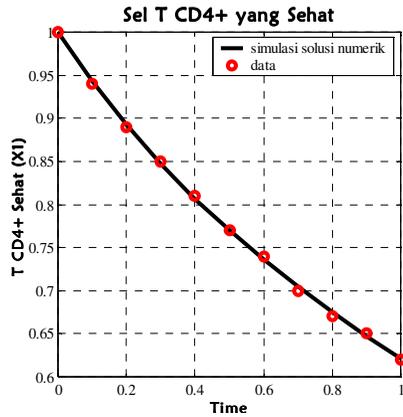
Pada Tabel 4.10 tampak bahwa dengan ukuran populasi sebanyak 100 individu dan probabilitas mutasi sebesar 0.01 merupakan nilai yang paling optimal (*error* terkecil) dengan nilai *fitness* sebesar 9.9588, dan nilai *error* sebesar 0.0004. Dalam hal ini probabilitas mutasi yang digunakan adalah antara 0.01-0.1.

Berdasarkan pada Tabel 4.9 dan Tabel 4.10 dengan melakukan percobaan sebanyak 1 kali untuk masing-masing simulasi diperoleh nilai yang optimal (*error* terkecil) dengan ukuran populasi sebanyak 100 individu, probabilitas *crossover* sebesar 0.95 dan probabilitas mutasi 0.01. Nilai-nilai parameter yang diperoleh adalah  $S = 0.9854$ ,  $d = 0.8350$ ,  $\beta = 0.9424$ ,  $\mu_1 = 0.6875$ ,  $k = 0.9971$ , dan  $\mu_2 = 0.0059$  seperti pada Gambar 4.9.



**Gambar 4.9** Estimasi Parameter Menggunakan Algoritma Genetika dengan  $Pop\_size$  100,  $p_c$  0.95,  $p_m$  0.01

Berdasarkan nilai parameter hasil estimasi, pencocokan penyelesaian sistem persamaan diferensial (4.11) terhadap data tampak seperti pada Gambar 4.10. Dari Gambar 4.10, data dan solusi numerik sudah saling berhimpit meskipun ada sedikit selisih jarak.



Gambar 4.10 Perbandingan antara Data dan Solusi Numerik untuk 1 Kali Percobaan Setiap Simulasi

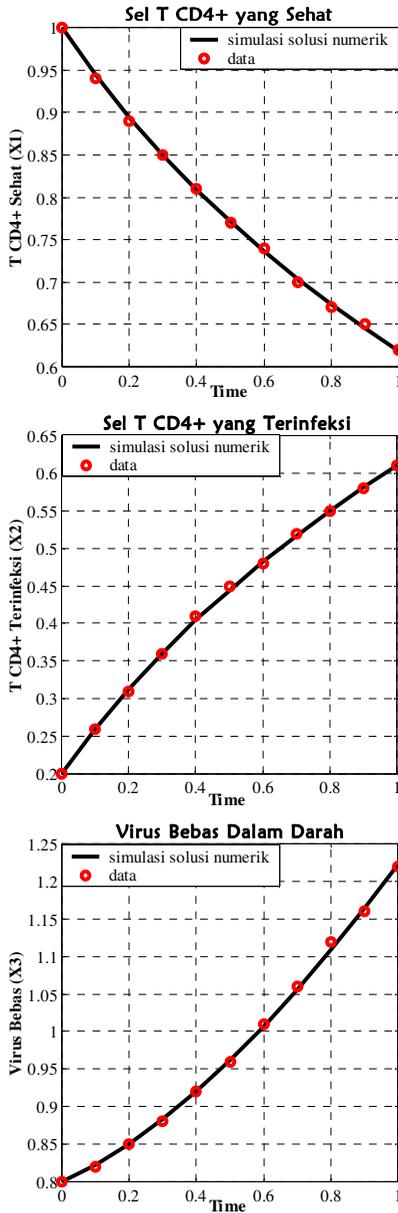
Simulasi numerik juga dilakukan untuk percobaan sebanyak 10 kali. Pada masing-masing simulasi diambil nilai *fitness* tertinggi. Simulasi dilakukan dengan menggunakan probabilitas *crossover* ( $p_c$ ) dan ukuran populasi (*pop\_size*) yang berbeda-beda, dengan jumlah generasi sebesar 50, probabilitas mutasi ( $p_m$ ) adalah 0.01, dan jumlah gen sebesar 60. Berdasarkan hasil simulasi, diperoleh nilai *fitness* dan *error* yang disajikan seperti pada Tabel 4.11.

**Tabel 4.11** Hasil Estimasi untuk Setiap 10 Kali Percobaan

Ukuran Populasi	Probabilitas <i>crossover</i>	Probabilitas mutasi	<i>Fitness</i>	Nilai <i>Error</i>
50	0.65	0.01	9.9594	0.0004
50	0.75		9.9284	0.0007
50	0.85		9.9579	0.0004
50	0.95		9.9591	0.0004
100	0.65		9.9584	0.0004
100	0.75		9.9440	0.0006
100	0.85		<b>9.9616</b>	0.0004
100	0.95		9.9588	0.0004

Berdasarkan Tabel 4.11 tampak bahwa dengan ukuran populasi 100 individu, probabilitas *crossover* 0.85, probabilitas mutasi 0.01 merupakan nilai yang paling optimal dengan nilai *fitness* sebesar 9.9616. Nilai-nilai parameter yang diperoleh adalah  $S = 0.9326$ ,  $d = 0.7549$ ,  $\beta = 0.9600$ ,  $\mu_1 = 0.7236$ ,  $k = 0.9883$ , dan  $\mu_2 = 0.0049$ . Pencocokan solusi numerik dari estimasi parameter terhadap data dapat ditampilkan seperti pada Gambar 4.11. Hasil estimasi selengkapnya dapat dilihat pada Lampiran 2.

Dari berbagai hasil simulasi yang diperoleh, dapat dijelaskan bahwa penentuan parameter algoritma genetika yang terdiri atas ukuran populasi, jumlah generasi, probabilitas *crossover*, dan probabilitas mutasi sangat berpengaruh pada hasil estimasi. Selain itu, karena semua komponen algoritma genetika berbasis pada fungsi *random* (bilangan acak), sehingga setiap kali menjalankan program akan menghasilkan solusi yang berbeda-beda. Karena tidak terdapat teori umum tentang pemilihan parameter algoritma genetika, maka dalam Skripsi ini dilakukan berkali-kali percobaan untuk memperoleh parameter yang tepat.



**Gambar 4.11** Perbandingan antara Data dan Solusi Numerik untuk 10 Kali Percobaan Setiap Simulasi

## BAB V PENUTUP

### 5.1 Kesimpulan

Berdasarkan pembahasan Skripsi ini dapat diambil beberapa kesimpulan antara lain:

1. Dengan menggunakan model kompartemen, diperoleh model sistem imun HIV berupa sistem persamaan diferensial biasa nonlinier.
2. Algoritma genetika digunakan sebagai metode estimasi parameter pada model sistem imun HIV mampu mencapai hasil yang cukup bagus, meskipun solusi yang tercapai bukan optimum global. Berdasarkan hasil simulasi, diperoleh nilai estimasi parameter yang paling optimal dengan 10 kali percobaan sebagai berikut:

$$S = 0.9326,$$

$$d = 0.7549,$$

$$\beta = 0.9600,$$

$$\mu_1 = 0.7236,$$

$$k = 0.9883,$$

$$\mu_2 = 0.0049,$$

dengan nilai *error* sebesar 0.0004.

### 5.2 Saran

Pada pembahasan selanjutnya, perlu dilakukan estimasi model dengan menggunakan data percobaan yang sebenarnya agar lebih menunjukkan ketepatan model secara umum.

UNIVERSITAS BRAWIJAYA



## DAFTAR PUSTAKA

- Anonimus<sup>1</sup>. 2006. *Informasi Dasar HIV-AIDS*, <http://www.aids-ina.org>, tanggal akses : 18 November 2008.
- Anonimus<sup>2</sup>. 2007. *Apakah HIV-AIDS Itu?*, <http://spiritia.or.id/li/bacali.php?lino=101>, tanggal akses : 6 Februari 2009.
- Anonimus<sup>3</sup>. 2008. *AIDS*, <http://id.wikipedia.org/wiki/AIDS>, tanggal akses : 18 November 2008.
- Boyce, W. E. dan R. C. Dprima. 1992. *Elementary Differential Equation and Boundary Value Problems*. Fifth edition. John Wiley and Sons. New York.
- Bronshtein, I.N. dkk. 2007. *Handbook of Mathematics*. Springer. Berlin.
- Bronson, R. 1996. *Teori dan Soal-Soal Operation Research*, Alih Bahasa : Drs. Hans J. Waspakrik. Erlangga. Jakarta.
- Finizio, N. dan G. Ladas. 1982. *Persamaan Diferensial Biasa dengan Penerapan Modern*. Erlangga. Jakarta.
- Gen, M., dan R. Cheng. 1997. *Genetic Algorithms and Engineering Design*. John Wiley and Sons.
- Kusumoputro, B. 2004. *Pengembangan Hemisphere Structure Of Hidden Layer Neural Networks Dan Optimasi Strukturnya Menggunakan Algoritma Genetika : Prosiding Semiloka Teknologi Simulasi dan Komputasi serta Aplikasi 2004*, Fakultas Ilmu Komputer : Universitas Indonesia.
- Leader, J.J. 2004. *Numerical Analysis and Scientific Computation International Edition*. Pearson Addison-Weasley. New York.

Luknanto, D. 2000. *Pengantar Optimasi Nonlinier*. Jurusan Teknik Sipil Fakultas Teknik Universitas Gajah Mada.

Manseur, S. dan N. Messaoudi. 2006. *Parameter Identification of an HIV Model by the Combined Adomian / Alienor Method*. *Kybernetes*, Vol. 35 No. 10 pp. 1725-1734.

Nagie, R. K. dan E. B. Saff. 1993. *Fundamentals of Differential Equations and Boundary Value Problems*. Addison-Wesley Publishing Company. USA.

Nowak, M. A. and C. R. M. Bangham. 1996. *Population Dynamics of Immune Responses to Persistent Viruses*. *Science*, 272, 74-79

Obitko, M. 1998. *Introduction to Genetic Algorithm*, <http://cs.felk.cvut.cz/~xobitko/ga/>, tanggal akses: 11 Desember 2008.

Pintauli, S. 2000. *AIDS dan Pencegahan Penularannya pada Praktek Dokter Gigi*. Universitas Sumatera Utara.

Ross, S. L. 1984. *Differential Equation*, 3<sup>rd</sup> ed. John Wiley and Sons. New York.

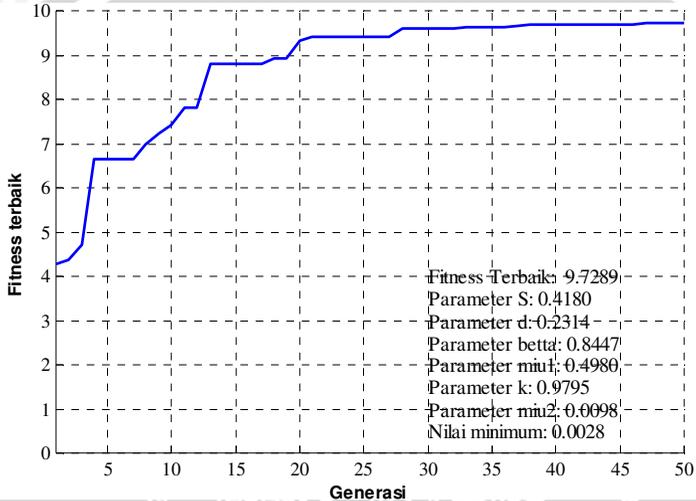
Setiawan, K. 2003. *Paradigma Sistem Cerdas : Jaringan Saraf Tiruan, Logika Fazi, dan Algoritma Genetika*. Bayumedia Publishing. Malang.

Spiegel, M.R. 1988. *Statistika*. Edisi Kedua. Erlangga. Jakarta.

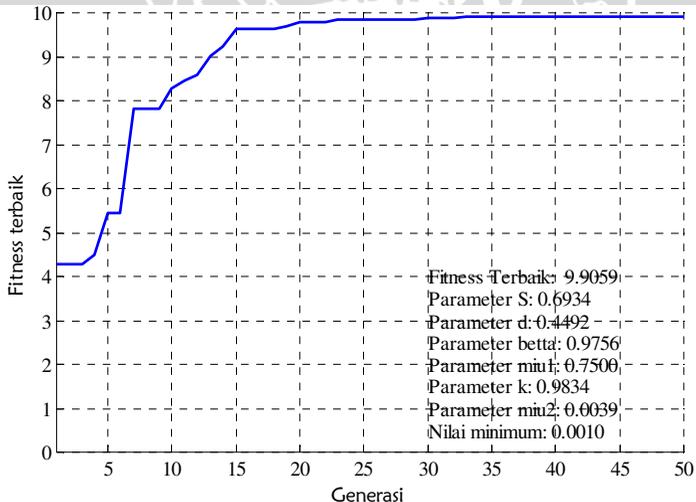
Suyanto. 2005. *Algoritma Genetika dalam Matlab*. Andi. Yogyakarta.

## LAMPIRAN

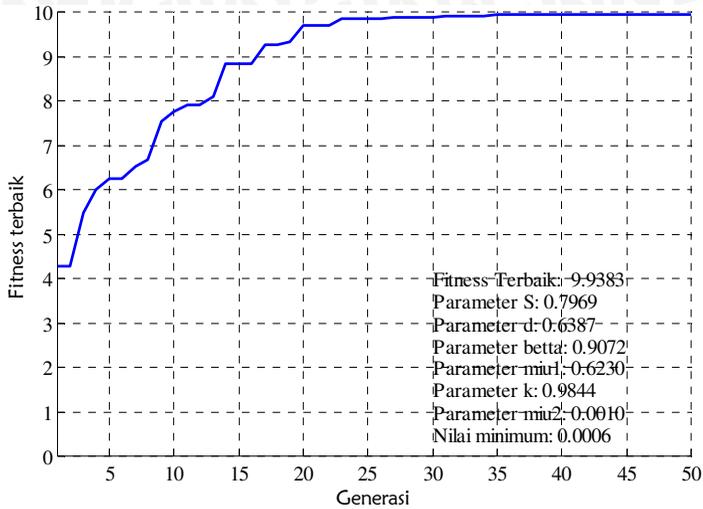
### Lampiran 1. Proses Simulasi Estimasi Parameter dengan $Pop\_size$ , $p_c$ , dan $p_m$ Berbeda untuk 1 Kali Percobaan Setiap Simulasi



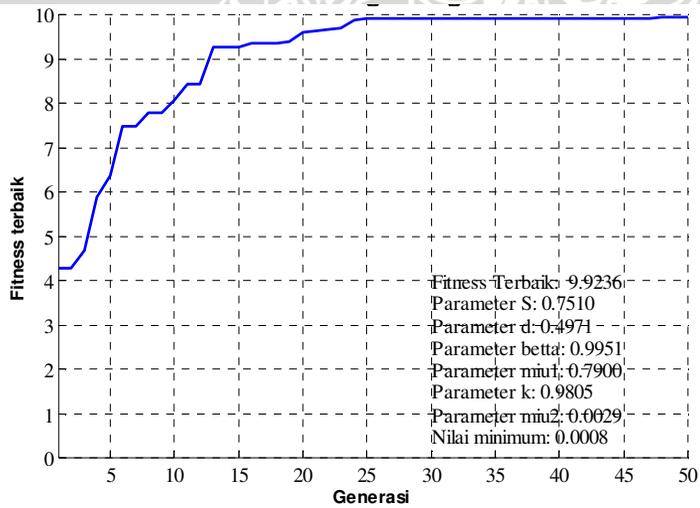
Solusi Algoritma Genetika dengan  $Pop\_size$  50,  $p_c$  0.65



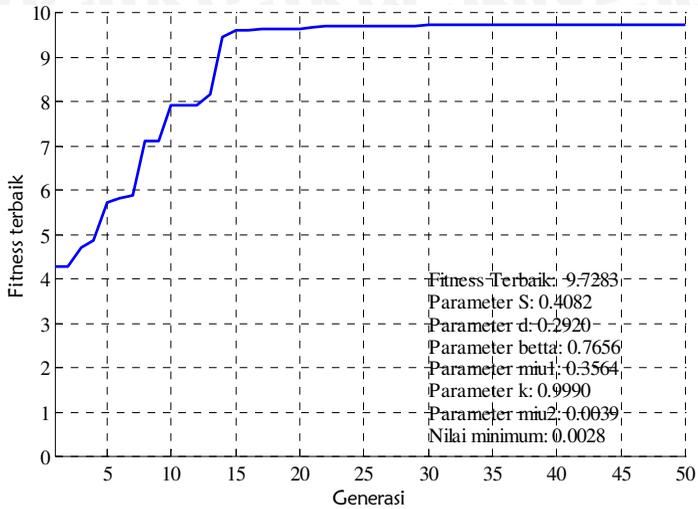
Solusi Algoritma Genetika dengan  $Pop\_size$  50,  $p_c$  0.75



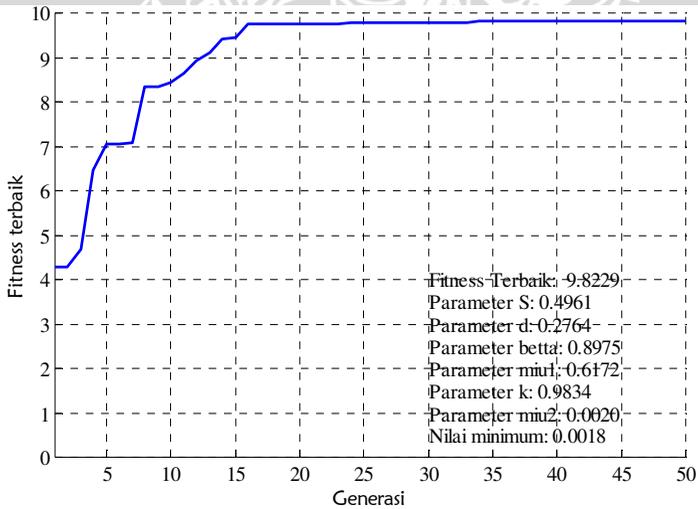
Solusi Algoritma Genetika dengan  $Pop\_size$  50,  $p_c$  0.85



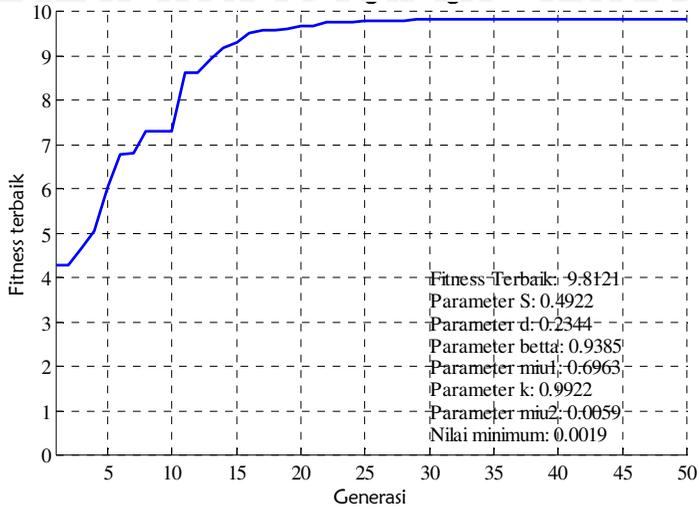
Solusi Algoritma Genetika dengan  $Pop\_size$  50,  $p_c$  0.95



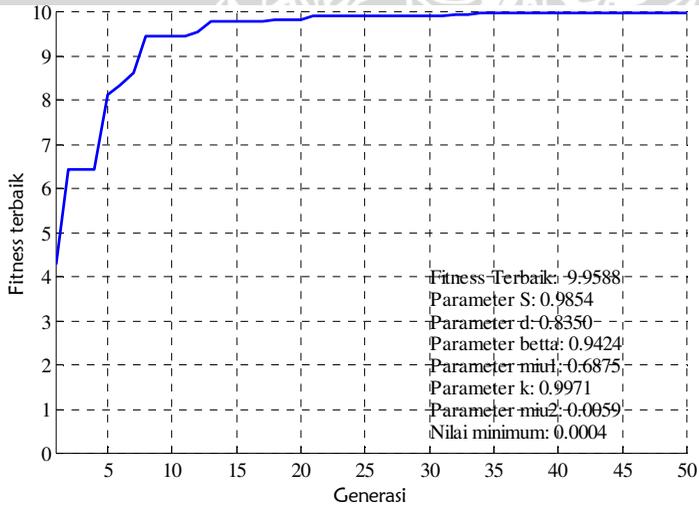
Solusi Algoritma Genetika dengan  $Pop\_size$  100,  $p_c$  0.65



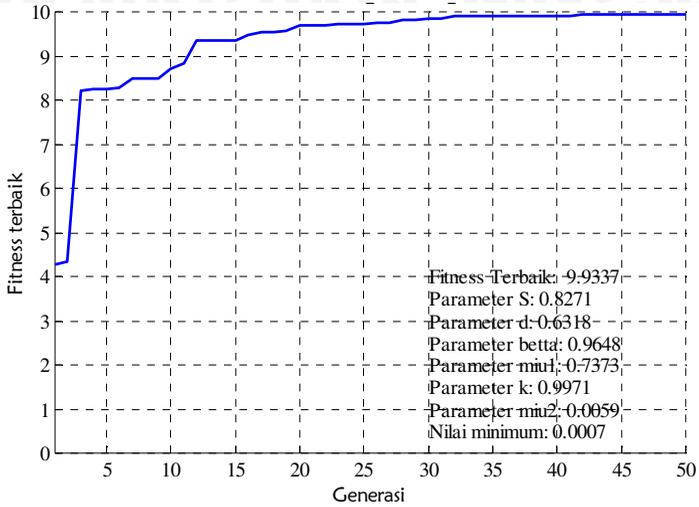
Solusi Algoritma Genetika dengan  $Pop\_size$  100,  $p_c$  0.75



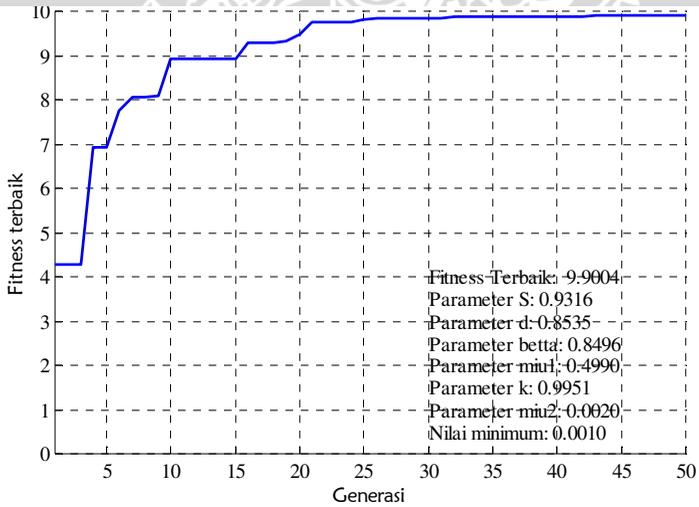
Solusi Algoritma Genetika dengan  $Pop\_size$  100,  $p_c$  0.85



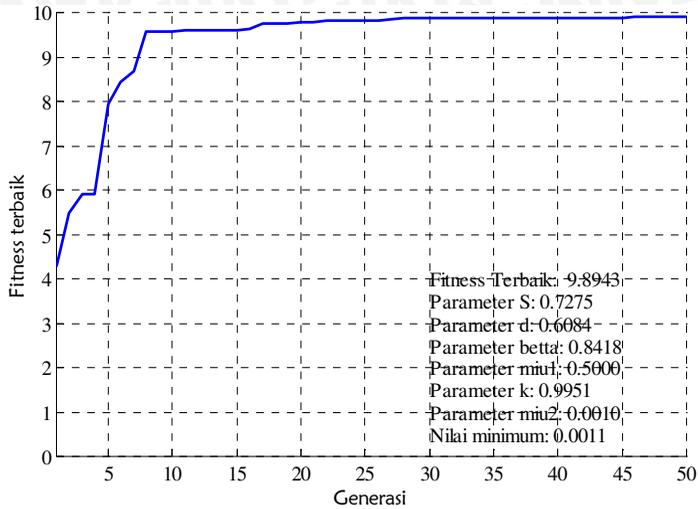
Solusi Algoritma Genetika dengan  $Pop\_size$  100,  $p_c$  0.95



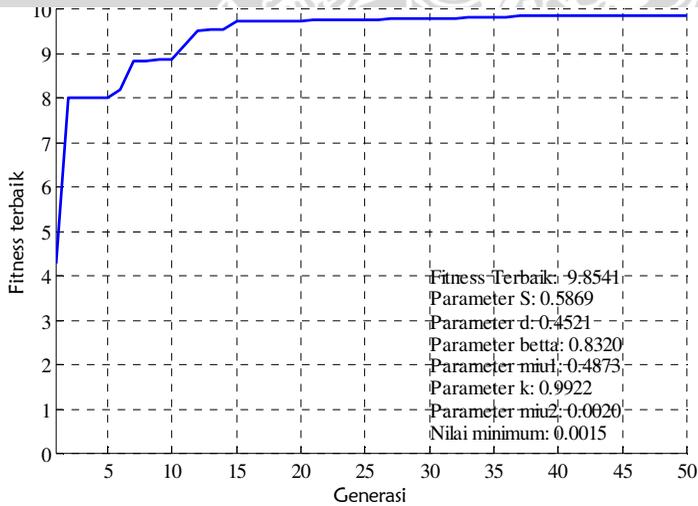
Solusi Algoritma Genetika dengan  $Pop\_size$  200,  $p_c$  0.65



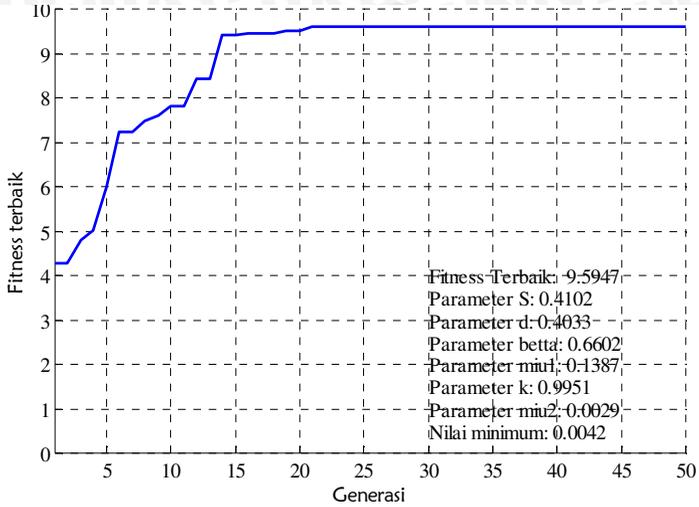
Solusi Algoritma Genetika dengan  $Pop\_size$  200,  $p_c$  0.75



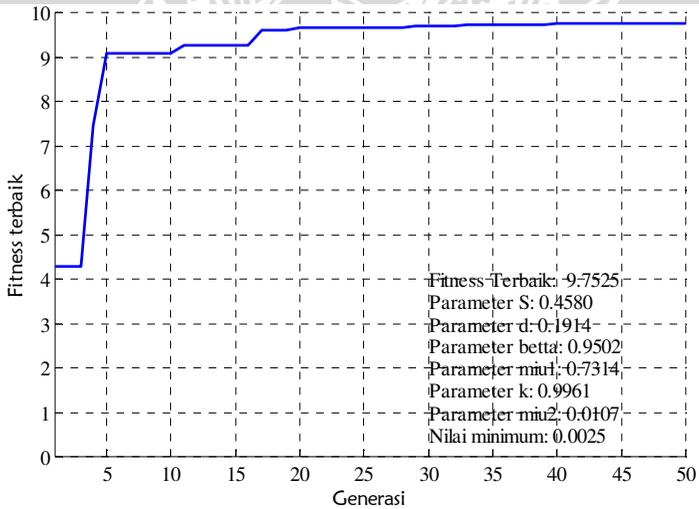
Solusi Algoritma Genetika dengan  $Pop\_size$  200,  $p_c$  0.85



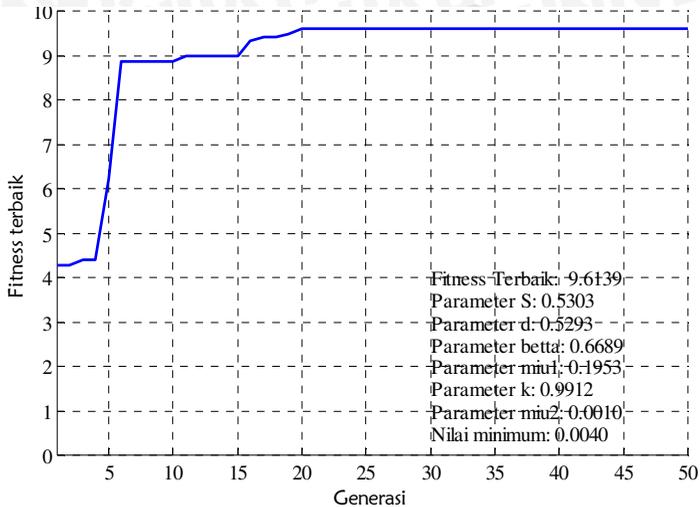
Solusi Algoritma Genetika dengan  $Pop\_size$  200,  $p_c$  0.95



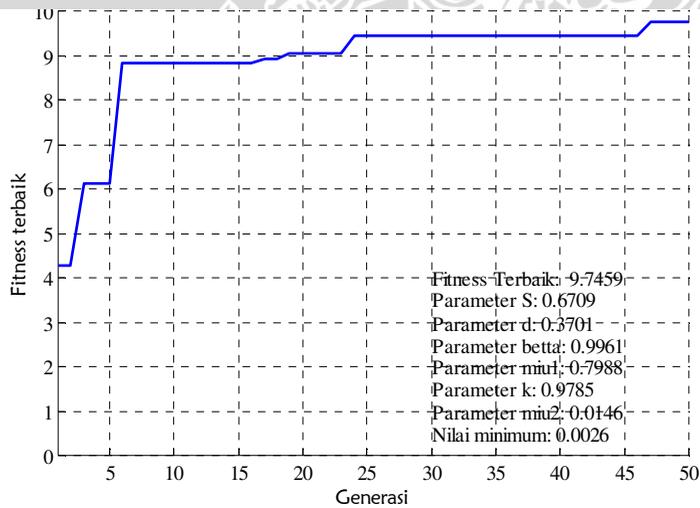
Solusi Algoritma Genetika dengan  $Pop\_size$  50,  $p_m$  0.03



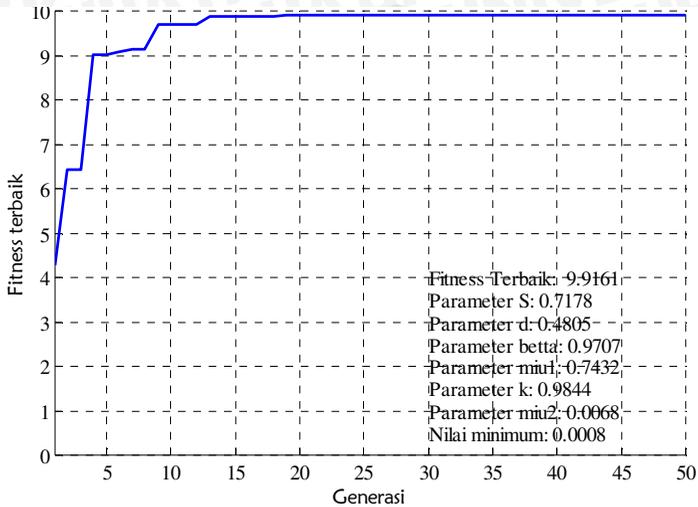
Solusi Algoritma Genetika dengan  $Pop\_size$  50,  $p_m$  0.05



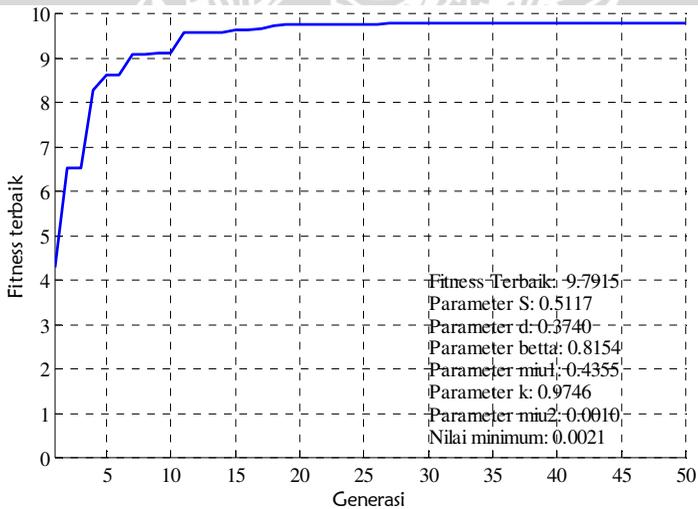
Solusi Algoritma Genetika dengan  $Pop\_size$  50,  $p_m$  0.08



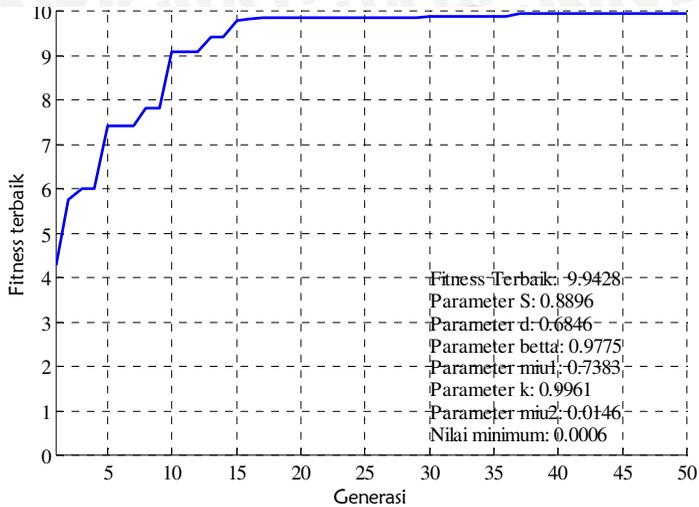
Solusi Algoritma Genetika dengan  $Pop\_size$  50,  $p_m$  0.1



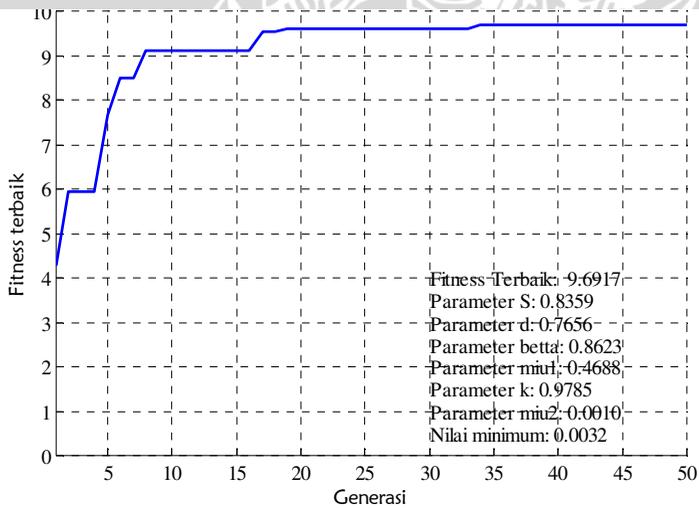
Solusi Algoritma Genetika dengan  $Pop\_size$  100,  $p_m$  0.03



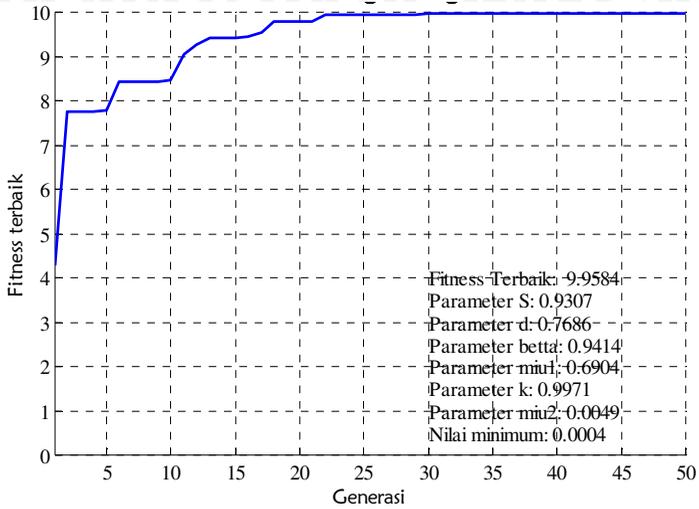
Solusi Algoritma Genetika dengan  $Pop\_size$  100,  $p_m$  0.05



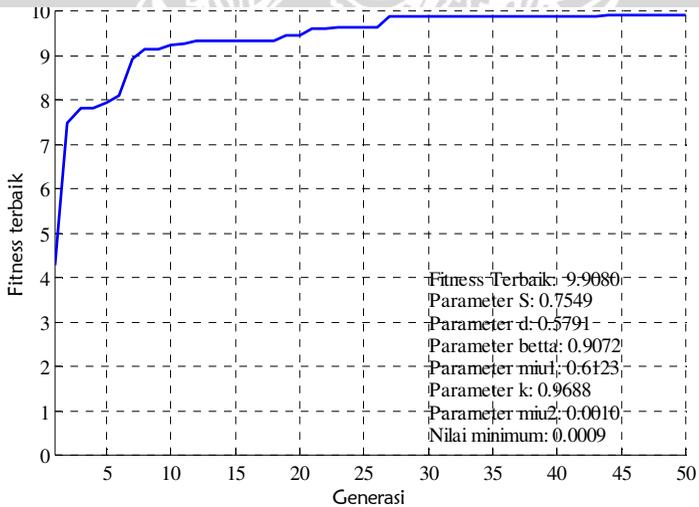
Solusi Algoritma Genetika dengan  $Pop\_size$  100,  $p_m$  0.08



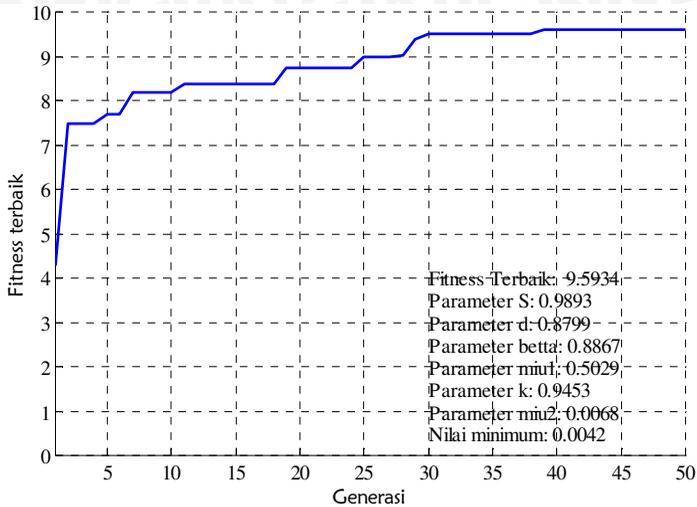
Solusi Algoritma Genetika dengan  $Pop\_size$  100,  $p_m$  0.1



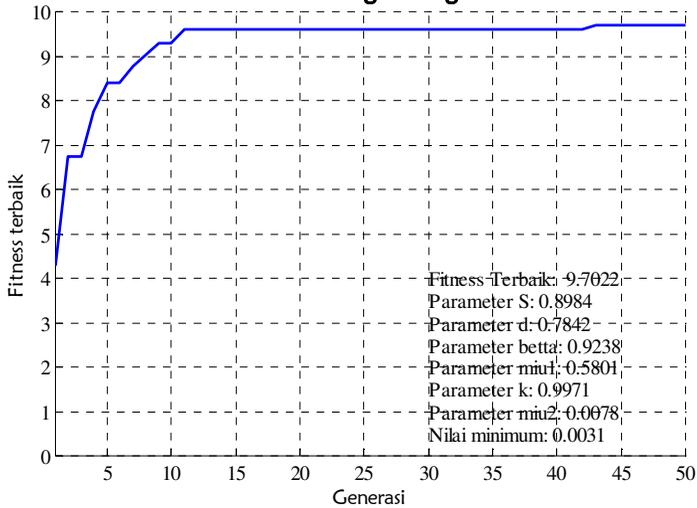
Solusi Algoritma Genetika dengan  $Pop\_size$  200,  $p_m$  0.03



Solusi Algoritma Genetika dengan  $Pop\_size$  200,  $p_m$  0.05

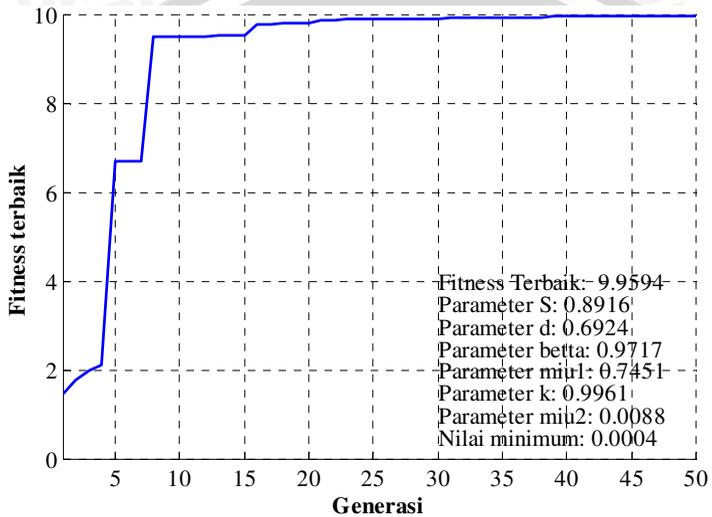


Solusi Algoritma Genetika dengan  $Pop\_size$  200,  $p_m$  0.08

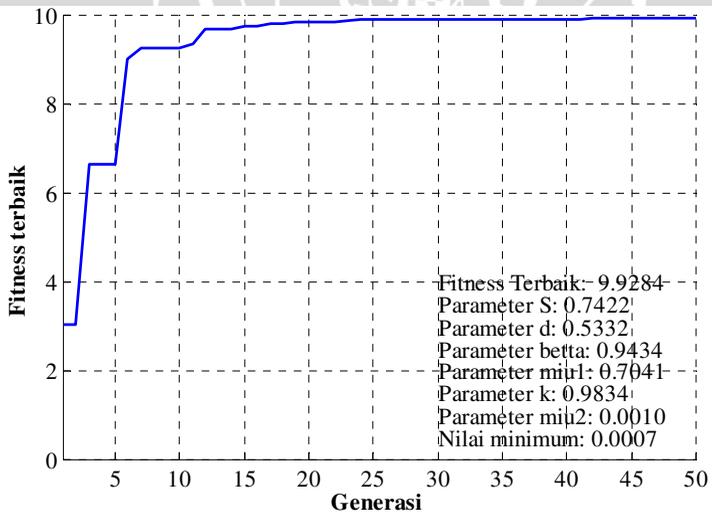


Solusi Algoritma Genetika dengan  $Pop\_size$  200,  $p_m$  0.1

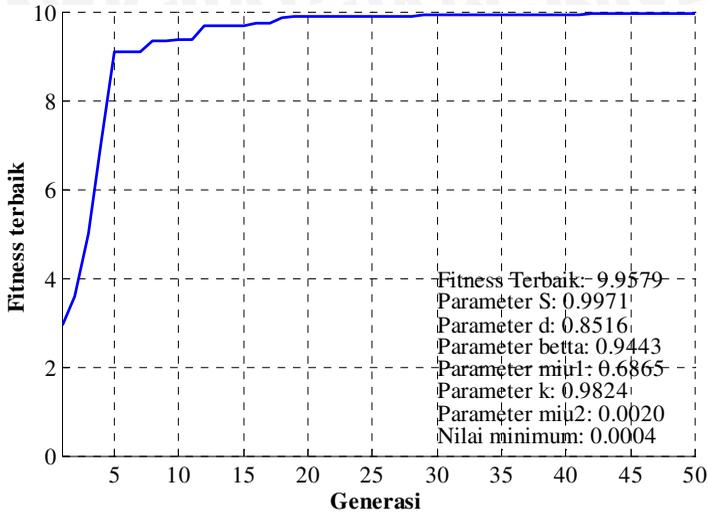
## Lampiran 2. Proses Simulasi Estimasi Parameter dengan 10 Kali Percobaan Setiap Simulasi



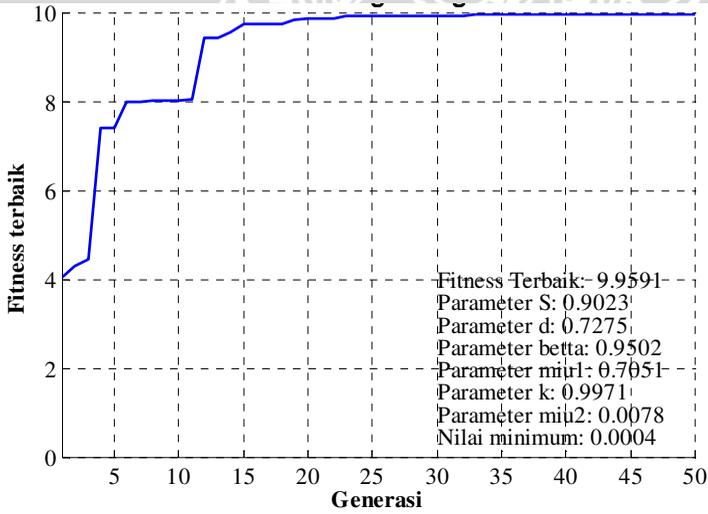
Solusi Algoritma Genetika dengan  $Pop\_size$  50,  $p_c$  0.65



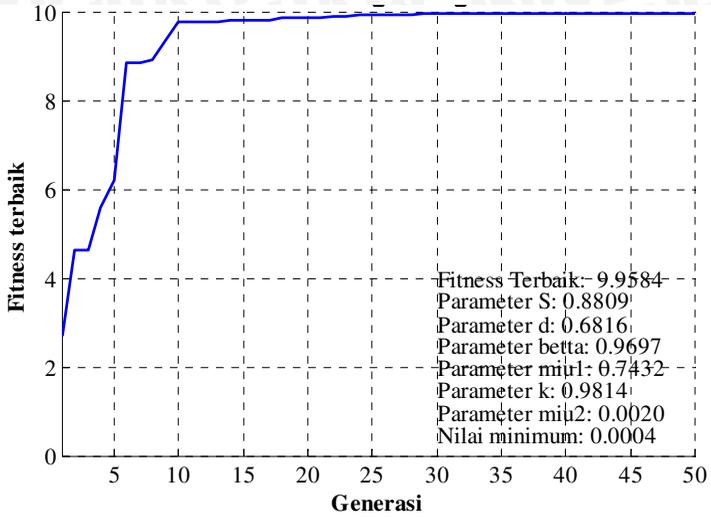
Solusi Algoritma Genetika dengan  $Pop\_size$  50,  $p_c$  0.75



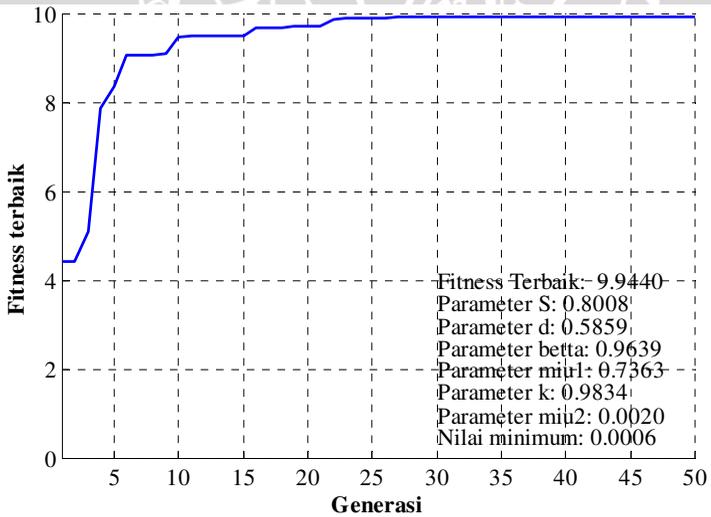
Solusi Algoritma Genetika dengan  $Pop\_size$  50,  $p_c$  0.85



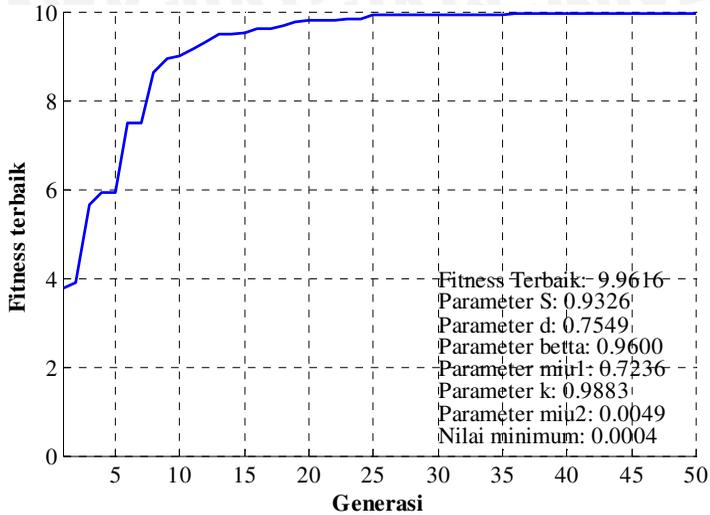
Solusi Algoritma Genetika dengan  $Pop\_size$  50,  $p_c$  0.95



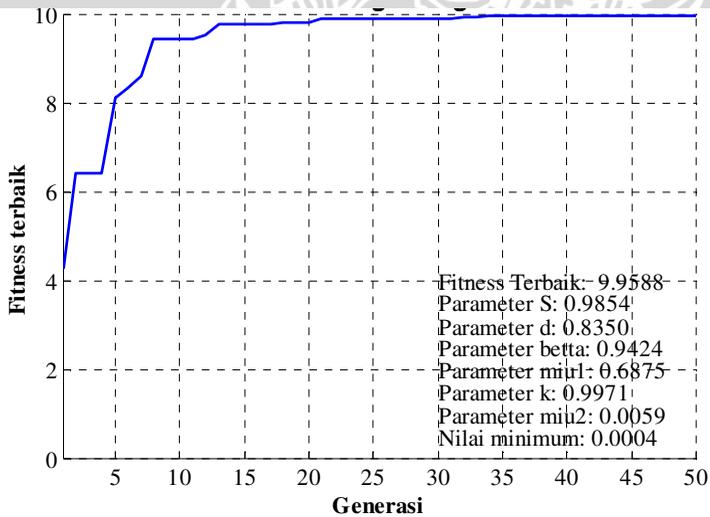
Solusi Algoritma Genetika dengan  $Pop\_size$  100,  $p_c$  0.65



Solusi Algoritma Genetika dengan  $Pop\_size$  100,  $p_c$  0.75



Solusi Algoritma Genetika dengan  $Pop\_size$  100,  $p_c$  0.85



Solusi Algoritma Genetika dengan  $Pop\_size$  100,  $p_c$  0.95