



TUGAS AKHIR - SM141501

**VERIFIKASI KEAMANAN MODEL DINAMIK
BAKTERIOFAG DENGAN BAKTERI RESISTEN
DAN DEBRIS BAKTERI MENGGUNAKAN MODEL
TAYLOR**

ANNISA MEGA PUTRI
NRP 06111440000009

Dosen Pembimbing:
Dr. Dieky Adzkiya, S.Si, M.Si

DEPARTEMEN MATEMATIKA
Fakultas Matematika, Komputasi, dan Sains Data
Institut Teknologi Sepuluh Nopember
Surabaya 2018

Halaman ini sengaja dikosongkan.



FINAL PROJECT - SM141501

**SAFETY VERIFICATION OF BACTERIOPHAGE
DYNAMIC MODEL WITH RESISTANT BACTERIA
AND BACTERIA DEBRIS USING TAYLOR MODEL**

ANNISA MEGA PUTRI
NRP 06111440000009

Supervisors:
Dr. Dieky Adzkiya, S.Si, M.Si

DEPARTMENT OF MATHEMATICS
Faculty of Mathematics, Computing, and Data Science
Institut Teknologi Sepuluh Nopember
Surabaya 2018

Halaman ini sengaja dikosongkan.

LEMBAR PENGESAHAN

VERIFIKASI KEAMANAN MODEL DINAMIK
BAKTERIOFAG DENGAN BAKTERI
RESISTEN DAN DEBRIS BAKTERI
MENGUNAKAN MODEL TAYLOR

*SAFETY VERIFICATION OF
BACTERIOPHAGE DYNAMIC MODEL
WITH RESISTANT BACTERIA AND
BACTERIA DEBRIS USING TAYLOR
MODEL*

Diajukan Untuk Memenuhi Salah Satu Syarat
Untuk Memperoleh Gelar Sarjana Sains
pada

Bidang Studi Matematika Terapan
Program Studi S-1 Departemen Matematika
Fakultas Matematika, Komputasi, dan Sains Data
Institut Teknologi Sepuluh Nopember Surabaya

Oleh:

ANNISA MEGA PUTRI
NRP. 06111440000009

Menyetujui,
Dosen Pembimbing,



Dr. Dieky Adzkiya, S.Si, M.Si
NIP. 19830517 200812 1 003

Mengetahui,
Kepala Departemen Matematika
FMKSD ITS



Dr. Imam Mukhlash, S.Si, MT
NIP. 19700831 199403 1 003

Surabaya, Agustus 2018



Halaman ini sengaja dikosongkan.

VERIFIKASI KEAMANAN MODEL DINAMIK BAKTERIOFAG DENGAN BAKTERI RESISTEN DAN DEBRIS BAKTERI MENGUNAKAN MODEL TAYLOR

Nama Mahasiswa : ANNISA MEGA PUTRI
NRP : 06111440000009
Departemen : Matematika FMKSD-ITS
Pembimbing : Dr. Dieky Adzkiya, S.Si, M.Si

Abstrak

Terapi bakteriofag (atau disebut juga fag) menjadi sebuah pengobatan dengan menggunakan bakteriofag (virus) untuk melawan beberapa strain bakteri yang paling mematikan (superbug) diantara bakteri yang resisten terhadap obat (antibiotik). Dalam Tugas Akhir ini, disajikan sebuah kasus terapi fag terhadap bakteri Escherichia Coli dan virus T4. Digunakan model Vidurupola untuk memodelkan interaksi antara bakteri Escherichia Coli dan virus T4. Analisis reachabilitas pada model non linear ini digunakan untuk memverifikasi bahwa kadar virus yang diberikan kedalam tubuh dapat membunuh superbug dan dapat dikatakan tubuh dalam kondisi aman dari bakteri yang menyerang. Untuk menghitung himpunan state yang terjangkau oleh sistem dalam waktu horizon yang dibatasi, diberikan himpunan kondisi awal dan parameter. Dari himpunan state yang terjangkau, dapat dilakukan verifikasi keamanannya. Hasil analisis reachabilitas dan verifikasi menunjukkan bahwa dari ketiga skenario yang diberikan, hanya skenario ke-3 yang kondisi awalnya dapat menjangkau state target dan menghindari kondisi tidak aman. Verifikasi dilakukan dengan menggunakan perangkat verifikasi Flow.*

Kata-kunci: *Terapi Fag, Analisis Reachabilitas, Verifikasi Sifat Keamanan, Flow*.*

Halaman ini sengaja dikosongkan.

SAFETY VERIFICATION OF BACTERIOPHAGE DYNAMIC MODEL WITH RESISTANT BACTERIA AND BACTERIA DEBRIS USING TAYLOR MODEL

Name : ANNISA MEGA PUTRI
NRP : 0611144000009
Department : Mathematics FMKSD-ITS
Supervisor : Dr. Dieky Adzkiya, S.Si, M.Si

Abstract

Treatment of bacteriophages (also called phages) becomes a treatment by using bacteriophages (viruses) to fight some of the most deadly (superbug) strains of bacteria among drug-resistant bacteria (antibiotics). In this Final Project, present a case of phage therapy against Escherichia coli bacteria and T4 virus. The Vidurupola model is used to model the interaction between Escherichia Coli bacteria and T4 virus. Reachability analysis on non linear models is used to verify that the viral load given to the body can kill the superbug and can be said to be in a safe condition from the invading bacteria. To calculate the set of state that the system reaches within a bounded time horizon, given the set of initial conditions and parameters. From an affordable set of states, safety can be verified. The results of reachability and verification analysis show that of the three scenarios were given, only the 3rd scenario whose initial condition can reach the target state and avoid unsafe conditions. Verification is done by using the Flow verification tool.*

Keywords: *Phage Therapy, Reachability Analysis, Verification of Safety Properties, Flow*.*

Halaman ini sengaja dikosongkan.

KATA PENGANTAR

Alhamdulillahirobbilalamin, segala puji dan syukur bagi Allah SWT yang telah memberikan rahmat, kasih sayang, dan petunjuk-NYA, sehingga penulis dapat menyelesaikan Tugas Akhir yang berjudul:

VERIFIKASI KEAMANAN MODEL DINAMIK BAKTERIOFAG DENGAN BAKTERI RESISTEN DAN BAKTERI DEBRIS MENGGUNAKAN MODEL TAYLOR

sebagai salah satu syarat kelulusan Program Sarjana Departemen Matematika FMKSD Institut Teknologi Sepuluh Nopember (ITS) Surabaya.

Tugas Akhir ini dapat terselesaikan dengan baik berkat bantuan dan dukungan dari berbagai pihak. Oleh karena itu, penulis menyampaikan ucapan terima kasih kepada :

1. Dr. Imam Mukhlash, S.Si, MT, selaku Kepala Departemen Matematika ITS yang telah memberikan dukungan dan motivasi selama perkuliahan hingga terselesaikannya Tugas Akhir ini.
2. Bapak Dr. Didik Khusnul Arif, S.Si, M.Si selaku Kaprodi S1 dan Bapak Drs. Iis Herisman, M.Sc selaku Sekretaris Kaprodi S1 yang telah memberikan banyak bantuan dalam proses pengumpulan Tugas Akhir ini.
3. Bapak Dr. Dieky Adzkiya S.Si, M.Si selaku dosen pembimbing atas segala bimbingan dan motivasinya

kepada penulis dalam mengerjakan Tugas Akhir ini sehingga dapat terselesaikan dengan baik.

4. Bapak Drs. Kamiran, M.Si, ibu Dr. Dwi Ratna Sulistyaningrum, S.Si, MT, dan bapak Drs. Sadjidon, M.Si selaku dosen penguji atas saran yang telah diberikan demi perbaikan Tugas Akhir ini.
5. Bapak Dr. Hariyanto selaku dosen wali yang telah memberikan arahan akademik.
6. Bapak Ibu dosen serta seluruh staf Tenaga Kependidikan Departemen Matematika Institut Teknologi Sepuluh Nopember.
7. Keluarga (Ibu, Bapak, dan Kakak) tercinta terima kasih atas doa dan dukungannya yang selalu dicurahkan kepada penulis.
8. Teman-teman mahasiswa matematika ITS angkatan 2014 yang telah memberikan masa-masa berkesan bagi penulis selama menjadi bagian dari mereka.

Penulis juga menyadari bahwa dalam Tugas Akhir ini masih terdapat kekurangan. Oleh sebab itu, kritik dan saran yang bersifat membangun sangat penulis harapkan demi kesempurnaan Tugas Akhir ini. Akhirnya, penulis berharap semoga Tugas Akhir ini dapat bermanfaat bagi banyak pihak.

Surabaya, Agustus 2018

Penulis

DAFTAR ISI

HALAMAN JUDUL	i	
LEMBAR PENGESAHAN	vi	
ABSTRAK	vii	
ABSTRACT	ix	
DAFTAR ISI	xiii	
DAFTAR GAMBAR	xv	
DAFTAR TABEL	xvii	
DAFTAR SIMBOL	xix	
BAB I	PENDAHULUAN	1
1.1	Latar Belakang	1
1.2	Rumusan Masalah	3
1.3	Batasan Masalah	3
1.4	Tujuan	4
1.5	Manfaat	4
1.6	Sistematika Penulisan	5
BAB II	TINJAUAN PUSTAKA	7
2.1	Penelitian Terdahulu	7
2.2	Model Matematika	8
2.3	Verifikasi Keamanan (<i>Safety Verification</i>)	11
2.4	Analisis Reachabilitas (<i>Reachability Analysis</i>)	13
2.4.1	<i>Forward dan Backward Reachability</i>	14
2.4.2	Model Taylor	15
2.4.3	Metode Integrasi Model Taylor	16
2.5	Flow*	22

BAB III	METODE PENELITIAN	25
3.1	Tahap Penelitian.....	25
3.2	Alur Penelitian	26
BAB IV	ANALISIS DAN PEMBAHASAN	29
4.1	Terapi Fag	29
4.2	Model Deterministik	31
4.3	Menghitung Flowpipe Model Taylor	33
4.4	Pendeskripsian Permasalahan Reachabilitas .	36
4.4.1	Definisi Sistem	36
4.4.2	Kondisi Awal, Target dan Tidak Aman	37
4.4.3	Pengaturan Reachabilitas.....	39
4.5	Pemeriksaan Sifat Keamanan.....	40
BAB V	PENUTUP	53
5.1	Kesimpulan	53
5.2	Saran	54
DAFTAR PUSTAKA		55
LAMPIRAN A	Diagram Kompartemen Terapi Fag (Bakteri dan Bakteriofag)	57
LAMPIRAN B	File <i>.model .flow</i>	59
LAMPIRAN C	Flowpipe dan Simulasi Numerik Skenario 1	71
LAMPIRAN D	Biodata Penulis	75

DAFTAR GAMBAR

Gambar 2.1	Fase Lisis Bakteriofag	8
Gambar 2.2	Permasalahan <i>Reachability</i>	13
Gambar 2.3	Struktur Flow*	22
Gambar 3.1	Diagram Metode Penelitian Tugas Akhir.	27
Gambar 4.1	Perubahan <i>step-size</i> Pada Skenario 1 . .	36
Gambar 4.2	Definisi Sistem pada file <i>.model</i>	37
Gambar 4.3	Himpunan Kondisi Awal pada file <i>.model</i> (Skenario 1)	38
Gambar 4.4	Permasalahan Reachabilitas (Skenario 1)	39
Gambar 4.5	Himpunan Terjangkau (<i>reachable</i>) dari B dan P (Skenario 1)	41
Gambar 4.6	Dinamika Bakteri E. Coli dan Fag T4 (Skenario 1)	42
Gambar 4.7	Hasil Verifikasi Keamanan pada Flow* (Skenario 1)	42
Gambar 4.8	Flowpipe Hasil Verifikasi (Skenario 1) .	43
Gambar 4.9	Himpunan Terjangkau (<i>reachable</i>) dari B dan P (Skenario 2)	44
Gambar 4.10	Dinamika Bakteri E. Coli dan Fag T4 (Skenario 2)	44
Gambar 4.11	Hasil Verifikasi Keamanan pada Flow* (Skenario 2)	45
Gambar 4.12	Himpunan Terjangkau (<i>reachable</i>) dari B dan P (Skenario 3)	46

Gambar 4.13 Dinamika Bakteri E. Coli dan Fag T4 (Skenario 3)	47
Gambar 4.14 Hasil Verifikasi Keamanan pada Flow* (Skenario 3)	47
Gambar 4.15 Dinamika Bakteri Resisten	49
Gambar 4.16 Dinamika Bakteri-Fag Kompleks	49
Gambar 4.17 Dinamika Bakteri Laten	50
Gambar 4.18 Dinamika Bakteri Aktif	50
Gambar 4.19 Dinamika Bakteri Debris	51

DAFTAR TABEL

Tabel 2.1	Parameter Bakteri <i>E.Coli</i> dan Bakteriofag T4 [1].....	10
Tabel 2.2	Algoritma Perhitungan Reachabilitas Maju	15

Halaman ini sengaja dikosongkan.

Daftar Simbol

B	Populasi bakteri rentan/sensitif.
R	Populasi bakteri resisten.
Q	Populasi bakteri-fag kompleks.
L	Populasi bakteri yang terinfeksi secara laten.
I	Populasi bakteri yang terinfeksi secara aktif.
D	Bakteri Debris.
P	Populasi bakteriofag bebas.
X_l	Himpunan kondisi awal.
$\varphi_f(X_0, t)$	Himpunan solusi dari kondisi awal X_0 .
$p(\vec{x}_0, t)$	Polinomial atas himpunan dari variabel \vec{x}_0 .
I	Sisa interval.
Δ	Interval waktu.
δ_i	<i>Time-step</i> ke-i.
k	Orde model Taylor.
f_i	Komponen ke-i dari $f\vec{x}$.
λ_\uparrow	Skalar keatas.
λ_\downarrow	Skalar kebawah.
\mathcal{F}_i	Flowpipe model Taylor ke-i.
α	<i>Time-step</i> minimum.
β	<i>Time-step</i> maksimum.

Halaman ini sengaja dikosongkan.

BAB I

PENDAHULUAN

Pada bab pendahuluan ini akan dijelaskan dan diuraikan mengenai, latar belakang masalah, rumusan masalah, batasan masalah, tujuan, dan manfaat dalam Tugas Akhir ini.

1.1 Latar Belakang

Bakteriofag atau lebih dikenal sebagai fag adalah virus yang membunuh bakteri [1]. Bakteriofag sendiri merupakan virus yang menggandakan dirinya sendiri dengan cara menyerang bakteri. Dibandingkan dengan kebanyakan virus lainnya, virus ini sangat kompleks dan memiliki beberapa bagian berbeda yang diatur secara cermat. Fag ini dapat digunakan untuk mengobati infeksi bakteri, yaitu sebagai terapi melawan bakteri terinfeksi atau disebut juga terapi fag [2]. Terapi ini ada karena penyebab penyakit yang berasal dari bakteri berkembang dan berevolusi menjadi bakteri yang kebal terhadap obat-obatan [3]. Dalam menyerang bakteri, bakteriofag dapat melakukan 2 fase yaitu fase litik dan fase lisogenik. Penggunaan fase litik dalam terapi fag memiliki kelebihan yaitu, fase litik mengalami proses akhir dimana sel inang (bakteri) yang dihinggapi bakteriofag langsung mengalami kematian dan pada fase lisogenik mengalami proses akhir dimana sel inang mengalami pembelahan terlebih dahulu dan kemudian mengalami kematian. Fase lisogenik tidak cocok untuk terapi fag karena interaksinya saling menguntungkan antara fag dan bakteri. Jadi, untuk terapi fag ini digunakan fag dengan fase litik. Beberapa contohnya, yaitu fag- T, T1 sampai T7, yang bekerja pada

fase litik.

Pada beberapa kasus, terapi fag telah dapat menyembuhkan beberapa penyakit. Antara lain, infeksi saluran pernapasan kronis yang disebabkan oleh strain *Pseudomonas Aeruginosa*, kolera yang disebabkan oleh bakteri *Vibrio Cholerae*, disentri yang disebabkan oleh bakteri *Shigella* atau bakteri *Entamoeba Histolytica*, dan beberapa penyakit lainnya. Pada penelitian ini, terapi fag akan diterapkan pada infeksi oleh bakteri *Escherichia Coli*. Walaupun sebagian besar strain E.Coli tidak berbahaya, namun beberapa dapat berperan sebagai bakteri patogen. Beberapa efek dari bakteri patogen E.Coli antara lain, muntaber, diare, disentri, dan penyakit berbahaya lainnya. Sebuah model matematika nantinya akan digunakan untuk mengetahui keberhasilan terapi fag oleh bakteriofag T4 untuk bakteri patogen *Escherichia Coli* yang ada dalam tubuh dengan dilakukan analisis reachabilitas dan verifikasi sifat keamanan (*safety*).

Verifikasi dan reachabilitas diperlukan untuk mengetahui apakah model tersebut memenuhi sifat verifikasi dan menjangkau kondisi target yang ditetapkan. Salah satu sifat verifikasi (*verification property*) adalah keamanan (*safety*). Verifikasi sifat keamanan (*safety verification*) adalah memutuskan apakah ada keadaan tidak aman (*unsafe set*) yang mungkin dicapai dari keadaan awal (*initial set*) [4]. Pada verifikasi, reachabilitas (*reachability*) memiliki peran penting. Misalnya, dalam verifikasi keamanan tersebut. Analisis reachabilitas merupakan proses menghitung himpunan state yang dapat dicapai sebuah sistem [5]. Namun karena masalah reachabilitas ini tidak dapat dipecahkan, diperlukan sebuah metode yang dapat menghitung *over-approximation* untuk himpunan state yang dapat dicapai (*reachable*).

Pada Tugas Akhir ini, proses perhitungan *over-*

approximation dari state yang dapat dicapai, dihitung dengan metode integrasi model Taylor yang dilakukan dengan menggunakan sebuah perangkat lunak. Dan juga akan digunakan konstruksi flowpipe untuk penggunaan model Taylor pada analisis (*reachability*) sistem kontinu non-linier. Metode integrasi diatas merupakan metode yang dapat digunakan untuk memutuskan apakah model tersebut menjangkau kondisi target yang ditetapkan dan apakah terdapat keadaan tidak aman yang dicapai oleh sebuah sistem. Nantinya akan digunakan sebuah perangkat lunak untuk membantu verifikasi sifat keamanan pada sistem yang ada.

1.2 Rumusan Masalah

Dengan melihat pada latar belakang diatas, rumusan masalah yang akan dijabarkan dan diteliti adalah:

1. Bagaimana analisis *reachability* dari model dinamik bakteriofag dengan bakteri resisten dan debris bakteri menggunakan sebuah perangkat lunak berbasis *reachability automata hibrida*?
2. Bagaimana hasil verifikasi keamanan (*safety property*) pada model matematika dinamik bakteriofag dengan bakteri resisten dan bakteri debris?

1.3 Batasan Masalah

Untuk menghindari pembahasan yang meluas, pembahasan dibatasi dengan batasan masalah sebagai berikut:

1. Model matematika yang akan digunakan adalah sebuah model matematika dinamika bakteriofag dengan bakteri resisten dan debris bakteri oleh Sukhitha W. Vidurupola (2017) [1].

2. Nilai parameter yang digunakan merupakan untuk kasus *E. Coli* dan bakteriofag T4 [1].
3. Pada analisis reachabilitas model kontinu non linear, hanya akan digunakan reachabilitas maju (*forward reachability*).
4. Menggunakan perangkat lunak berbasis *reachability automata hibrida*, yaitu Flow* (flowstar) untuk memverifikasi model kontinu non linear.

1.4 Tujuan

Tujuan yang ingin dicapai dalam pengerjaan Tugas Akhir ini adalah sebagai berikut:

1. Mengetahui analisis reachabilitas dari model dinamik bakteriofag dengan bakteri resisten dan debris bakteri dengan menggunakan sebuah perangkat lunak berbasis *reachability automata hibrida*.
2. Mengetahui bagaimana hasil verifikasi terhadap sifat keamanan yang diberikan pada model matematika dinamik bakteriofag dengan bakteri resisten dan debris bakteri terpenuhi.

1.5 Manfaat

Manfaat yang diperoleh pada tugas akhir ini adalah sebagai berikut:

1. Memberikan hasil verifikasi pada model dinamik bakteriofag dengan bakteri resisten dan debris bakteri, sehingga dapat diketahui apakah terdapat keadaan tidak aman yang dicapai oleh sistem.
2. Memperkenalkan kepada pembaca mengenai perangkat lunak untuk verifikasi keamanan pada sistem kontinu non-linier.

1.6 Sistematika Penulisan

Penulisan tugas akhir ini disusun dalam lima bab yang masing-masing membahas permasalahan dalam tugas akhir. Dibuat dengan sistematika penulisan sebagai berikut:

1. BAB I PENDAHULUAN

Pada bab ini berisi tentang latar belakang, rumusan masalah, batasan masalah, tujuan, manfaat, dan sistematika penulisan.

2. BAB II TINJAUAN PUSTAKA

Pada bab ini berisi penelitian terdahulu dan gambaran secara umum mengenai konsep teori yang mendasari perancangan Tugas Akhir ini seperti model matematika, analisis reachabilitas, verifikasi keamanan dan Flow*.

3. BAB III METODE PENELITIAN

Dalam bab ini dijelaskan secara detail tentang metoda penelitian yang dilakukan. Antara lain studi literatur, selanjutnya dilakukan pendefinisian permasalahan *reachability*, analisis reachabilitas, verifikasi sifat keamanan dan yang terakhir analisis terhadap hasil verifikasi.

4. BAB IV ANALISIS DAN PEMBAHASAN

Pada Bab IV dibahas secara detail mengenai model matematika bakteriofag yang digunakan, pengaturan reachabilitas untuk dijalankan pada perangkat lunak Flow*, analisis reachabilitas dan analisis hasil verifikasi kemananannya.

5. BAB V PENUTUP

Pada bab yang terakhir berisi tentang kesimpulan dan saran-saran untuk pengembangan tugas akhir ini.

Halaman ini sengaja dikosongkan.

BAB II TINJAUAN PUSTAKA

Pada Bab II berisi penelitian terdahulu dan landasan teori yang digunakan dalam penyelesaian Tugas Akhir ini meliputi model bakteriofag, Verifikasi Keamanan, Analisis Reachabilitas, dan Flow*.

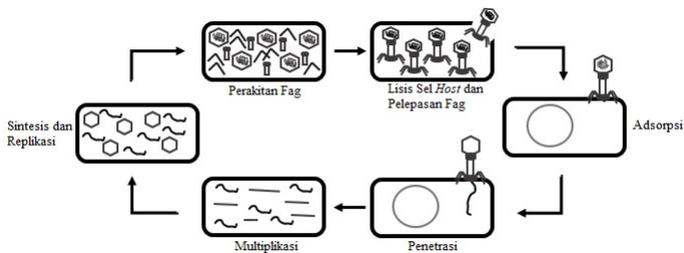
2.1 Penelitian Terdahulu

Dalam Tugas Akhir ini penulis merujuk pada penelitian yang pernah dilakukan, dimana penelitian tersebut berhubungan dengan Tugas Akhir ini. Penelitian yang dirujuk sebagai bahan kajian untuk Tugas Akhir, antara lain oleh Xin Chen, Souradeep Dutta, dan Sriram Sankaranarayanan yang berjudul *Formal Verification of a Multi-Basal Insulin Infusion Control Model* [6]. Pada penelitian tersebut, disajikan studi kasus sistem kontrol multibasal hibrida sederhana yang beralih ke tingkat pengiriman insulin preset yang berbeda pada berbagai rentang kadar glukosa. Penelitian ini menggunakan model Dalla-Man untuk memodelkan fisiologi pasien dan model automaton hibrida dari pengontrol. Tujuan dari penelitian adalah untuk memverifikasi bahwa tingkat glukosa darah tetap dalam kisaran yang aman dalam satu malam dengan menggunakan perangkat lunak Flow*. Mereka menggunakan 2 strategi kontrol berbeda yang didefinisikan oleh himpunan tingkat insulin. Dan hasil dari penelitian tersebut adalah strategi I dan II tidak menimbulkan hypoglycemia dan ketoacidosis, namun glukosa darahnya tidak berada pada rentang euglycemic. Sedangkan glukosa darah harus berada pada rentang euglycemic.

Penelitian selanjutnya adalah oleh Xin Chen, Matthias Althoff, dan Fabian Immler pada tahun 2017 yang berjudul *Continuous Systems with Nonlinear Dynamics* [7]. Pada penelitian tersebut dijelaskan tentang pemecahan masalah analisis reachabilitas pada tiga *benchmark* non linier yang masing-masing memiliki 2,7,12 variabel menggunakan tiga perangkat lunak antara lain adalah Flow*, Cora, dan Isabelle/HOL. Pada masing-masing perangkat lunak tersebut menghasilkan waktu perhitungan (*time cost*) yang berbeda-beda dengan kondisi awal (*initial set*) dan himpunan tidak aman (*unsafe set*) yang sama.

2.2 Model Matematika

Terapi fag yang akan digunakan adalah terapi fag dengan fase lisis, dimana prosesnya seperti pada gambar 2.1.



Gambar 2.1: Fase Lisis Bakteriofag

Untuk merepresentasikan terapi fag, digunakan sebuah model matematika dinamik bakteriofag dengan bakteri resisten dan debris bakteri oleh Sukhitha W. Vidurupola [1]. Model tersebut merupakan model bakteriofag fase litik yang didalamnya terdapat bakteri yang tahan terhadap fag, bakteri kompleks, dan debris bakteri. Model ditunjukkan oleh sistem

persamaan diferensial biasa berdimensi tujuh berikut ini [1]:

$$\begin{aligned}
 \frac{dB}{dt} &= r_1(B + Q + L + I + a_1R)B - f(P)B - kBP - \mu B \\
 \frac{dR}{dt} &= r_2(B + Q + L + I + a_2R)R + f(P)B - \mu B \\
 \frac{dQ}{dt} &= r_3(B + Q + L + I + a_3R)Q + kBP - \delta_1Q - \mu Q \\
 \frac{dL}{dt} &= \delta_1Q - \alpha L - \mu B \\
 \frac{dI}{dt} &= \alpha L - aL - \mu I \\
 \frac{dD}{dt} &= \mu(B + Q + L + I) + aI - \eta D \\
 \frac{dP}{dt} &= NaI + \delta_2Q - kBP - k_1(P)(L + 1) - k_2(P)Q \\
 &\quad - k_3(P)D - \gamma P
 \end{aligned} \tag{2.1}$$

Dengan variabel beserta deskripsinya sebagai berikut [1]:

B : Populasi bakteri tersuspensi / sensitif (Bakteri yg rentan)

R : Populasi bakteri resisten (Bakteri yg tahan)

Q : Populasi bakteri-fag (BP) kompleks

L : Populasi bakteri yang terinfeksi (fag) secara perlahan

I : Populasi bakteri yang terinfeksi (fag) secara aktif

D : Debris bakteri (Puing-puing bakteri)

P : Populasi fag bebas

k : Tingkat adsorpsi (penempelan) fag terhadap bakteri yang rentan

μ : Tingkat kematian alami sel bakteri B, L dan I

α : Tingkat konversi dari bakteri yang terinfeksi secara

- perlahan ke bakteri yang terinfeksi secara aktif
- a : Tingkat kematian sel bakteri yang terinfeksi akibat serangan fag
- N : Jumlah burst (ledakan)
- γ : Tingkat kematian fag yang bebas
- δ_1 : Tingkat konversi dari BP kompleks menjadi bakteri yang terinfeksi secara perlahan
- δ_2 : Tingkat detasemen (pelepasan) fag dari BP kompleks
- η : Tingkat degradasi sisa bakteri
- $f()$: Fungsi yang menunjukkan tingkat mutasi B terhadap R
- $r_i()$: Fungsi yang menunjukkan tingkat pertumbuhan B, R, dan Q
- $k_i()$: Fungsi yang menunjukkan tingkat kelenturan fage ke L, I, Q, dan D
- a_i : Tingkat bakteri resisten yang bersaing untuk sumber daya bersama

Dalam dinamika interaksi bakteriofag, digunakan nilai parameter bakteri patogen *E. Coli* dan bakteriofag T4 yang bekerja dengan fase litik.

Tabel 2.1: Parameter Bakteri *E.Coli* dan Bakteriofag T4 [1]

Parameter	Nilai
r_{11}	$0.3 h^{-1}$
r_{22}	$0.5 h^{-1}$
r_{33}	0
a_1	0.4
a_2	0.5
a_3	0.6

Parameter	Nilai
C	$10^8 v(vml)^{-1}$
c_1	$10^8 / v(vml)$
f_1	$10^{-6} h^{-1}$
k	$9 \times 10^{-7} / v(vmlh)^{-1}$
α	$6.98 h^{-1}$
μ	$10^{-7} h^{-1}$
a	$6.98 h^{-1}$
N	150 virus (sel^{-1})
γ	$1.8 h^{-1}$
δ_1	$6.98 h^{-1}$
δ_2	0
η	$0.025 h^{-1}$

2.3 Verifikasi Keamanan (*Safety Verification*)

Sifat keamanan (*safety*) memiliki spesifikasi “Tidak ada state tidak aman (*unsafe*) yang dapat dijangkau (*reachable*)” . Dan masalah verifikasi keamanan dapat ditulis, Diberikan sistem dinamik n-dimensi, himpunan awal $X_0 \in \mathbb{R}^n$, himpunan tidak aman $unsafe \subseteq \mathbb{R}^n$, dan waktu terbatas $T > 0$, masalah verifikasi keamanan adalah membuktikan bahwa $Reach_u(X_0, [0, T]) \cap unsafe = \emptyset$. Dalam verifikasi keamanan, dapat dihitung superset untuk himpunan yang dapat dicapai dengan tepat. Jika tidak ada state tidak aman yang dapat tercapai, maka sistemnya aman. Sifat keamanan (*safety*) sangat penting bagi sistem hibrid dan verifikasinya dapat dilakukan dengan perhitungan *reachability* di state space hibrid.

Telah dibuktikan bahwa masalah verifikasi keamanan dan analisis *reachability* sistem hibrida pada umumnya tidak dapat dipecahkan (Alur et al., 1995; Lafferriere, Pappas, & Yovine, 1999). Terdapat beberapa teknik yang

dapat digunakan untuk menyelesaikan masalah verifikasi dan reachabilitas ini, salah satunya yaitu dengan teknik aproksimasi atau *over-approximate*. Atau secara lengkapnya, Teknik dan alat yang digunakan untuk verifikasi dalam Tugas Akhir ini adalah teknik *over-approximating flowpipe construction* (konstruksi flowpipe yang di-over-approximasi). Teknik ini diselesaikan dengan sebuah metode, yaitu metode integrasi *Taylor Model* (TM). Setelah himpunan state yang tercapai (*reachable*) didapatkan, selanjutnya memverifikasi apakah sifat keamanan dilanggar dengan memeriksa apakah keadaan tidak aman berada diantara himpunan state yang dapat dicapai. Perangkat lunak yang digunakan untuk verifikasi bernama Flow*, dimana berfokus pada verifikasi keamanan sistem hibrida non linier.

Berikut ini adalah beberapa klasifikasi untuk masalah verifikasi keamanan untuk sistem kontinu, antara lain [8]:

- (a) Jenis dinamika kontinu, misalnya: konstan/linier/non-linier.
- (b) Dimensi sistem, yaitu: jumlah variabel state, $|\vec{x}|$.
- (c) Tipe dari verifikasi keamanan, yaitu: waktu dibatasi atau waktu tak dibatasi.
- (d) Sifat konstrain evolusi, yaitu: state space dibatasi atau state space tak dibatasi.
- (e) Sifat himpunan awal dan himpunan yang tidak diinginkan, yaitu: dibatasi atau tak dibatasi.
- (f) Sifat dari permasalahan verifikasi, yaitu: apakah sistem tersebut aman atau tidak aman.

2.4 Analisis Reachabilitas (*Reachability Analysis*)



Gambar 2.2: Permasalahan *Reachability*

Reachabilitas (*Reachability*) memiliki peran penting dalam verifikasi. Pada penelitian kali ini, akan dilakukan verifikasi terhadap sifat keamanan (*safety*). Jadi dalam verifikasi sifat keamanan, akan diperiksa apakah keadaan tidak aman (*unsafe state*) dapat dijangkau oleh sistem. Teknik pada perhitungan reachability ini adalah teknik *over-approximating flowpipe construction*, yaitu menghasilkan over-aproksimasi untuk himpunan yang dapat dicapai dalam rentang waktu terbatas untuk sistem kontinu. Teknik ini merupakan analisis forward reachability yang berulang (namun dibatasi) berdasarkan beberapa representasi himpunan state simbolik yang over-aproksimatif. Teknik tersebut biasanya digunakan untuk analisis reachabilitas berdasarkan konstruksi flowpipe. Untuk menyatakan himpunan state yang over-aproksimatif, dibutuhkan representasi over-aproksimatif geometrik atau simbolik. Representasi over-aproksimatif yang ada antara lain, polyhedra cembung, zonotopes, ellipsoids, boxes (hyper-rectangles), model Taylor, dan lain-lain [9].

Pada verifikasi keamanan ini, digunakan sebuah model Taylor sebagai representasi over-approximate untuk himpunan segmen yang dapat dijangkau (flowpipe). Himpunan over-aproksimasi dihitung sebagai banyak flowpipe model Taylor. Metode yang akan digunakan untuk menghasilkan over-aproksimasi dengan representasi model Taylor adalah metode integrasi model Taylor. Karena masalah reachabilitas pada sistem kontinu secara umum tidak

dapat dipecahkan, sebuah perangkat lunak akan menghitung over-approximation untuk state set yang dapat dijangkau (reachable). Jika state yang diberikan tidak termasuk, maka state tersebut tidak dapat dijangkau.

Himpunan sistem kontinu yang dapat dijangkau (reachable), ditentukan oleh $\vec{x} = f(\vec{x})$ dari himpunan awal $X_0 \subseteq \mathbb{R}^n$ adalah himpunan flow $\{\varphi_f(\vec{x}_0, t) | \vec{x}_0 \in X_0\}$. Untuk sederhananya, dapat ditulis dengan $\varphi_f(X_0, t)$ jika $\varphi_f(\vec{x}_0, t)$ ada untuk semua $\vec{x}_0 \in X_0$ dalam selang waktu yang ditentukan. Kemudian, Dituliskan \mathbb{I} menunjukkan himpunan semua interval $I = [a, b] \subseteq \mathbb{R}$ dengan $a, b \in \mathbb{R}$ dan $a \leq b$. Diberikan interval waktu $\Delta \in \mathbb{I}$, flow map $\varphi_f(X_0, t)$ dengan $t \in \Delta$, disebut dengan flowpipe.

2.4.1 *Forward dan Backward Reachability*

Dalam analisis reachabilitas untuk sifat keamanan, analisis reachabilitas bertujuan untuk menentukan apakah lintasan H dapat menjangkau I dari T , dengan tuple $S = (H, I, T)$ dimana H adalah model sistem, I adalah himpunan awal, dan T adalah himpunan target yang tidak aman. Terdapat dua jenis analisis pada analisis reachabilitas ini, antara lain reachabilitas maju (*forward reachability*) dan reachabilitas mundur (*backward reachability*). Reachabilitas maju dimulai dengan state pada I dan berjalan mengikuti lintasan (*trajectories*) maju dalam waktu berdasarkan dinamika sistem H . Apabila salah satu lintasan berpotongan dengan T , maka dikatakan sistem dapat menjangkau T . Reachabilitas mundur dimulai dengan state pada T dan berjalan mengikuti lintasan mundur dalam waktu berdasarkan dinamika sistem H . Apabila salah satu lintasan mundur berpotongan dengan I , maka dikatakan sistem dapat menjangkau I [10]. Pada penelitian kali ini, akan digunakan analisis reachabilitas maju (*forward reachability analysis*) untuk analisis reachability. Algoritma perhitungan analisis

Tabel 2.2: Algoritma Perhitungan Reachabilitas Maju

Input:
Himpunan <i>init</i> dari state awal
Algoritma:
$R^{new} := init;$
$R := \emptyset;$
$while(R^{new} \neq \emptyset)\{$
$R := R \cup R^{new};$
$R^{new} := Reach(R^{new}) \setminus R;$
$\}$
Output
Himpunan R dari state <i>reachable</i>

reachabilitas maju ditunjukkan oleh Tabel 2.2 [11].

2.4.2 Model Taylor

Model Taylor awalnya dikembangkan oleh Berz dan Makino untuk memberikan representasi over-aproksimasi untuk fungsi kontinu. Model Taylor (TM) dapat diterapkan untuk memberikan over-aproksimasi untuk flowpipe.

Definisi 2.4.2.1. [12]

Model Taylor (TM) dilambangkan dengan sepasang (p, I) sehingga p adalah polinomial pada himpunan variabel \vec{x} mulai dalam domain interval D , dan I adalah sisa interval. TM mungkin juga bernilai vektor. TM dengan nilai vektor dapat dipandang sebagai vektor TM yang bernilai riil, atau dengan cara keduanya p dan I bernilai vektor dan memiliki dimensi yang sama.

Diberikan TM (p, I) dan fungsi f yang berada pada domain D yang sama, dapat dikatakan bahwa f diover-aproksimasi oleh (p, I) , dilambangkan dengan $f \in (p, I)$ atau $f \in p +$

I , jika $f(\vec{x}) \in p(\vec{x}) + I, \forall \vec{x} \in D$. Salah satu fungsi model Taylor yaitu berfungsi sebagai definisi implisit dari flowpipe yang mengover-aproksimasi flow φ_f dalam himpunan $\vec{x}_0 \in X_0$ dan $t \in \Delta$. Dengan kata lain, sebuah flowpipe $\varphi_f(X_0, t)$ untuk beberapa $X_0 \subseteq \mathbb{R}^n$ dan $t \in \Delta \in \mathbb{I}$ dapat diover-aproksimasi oleh model Taylor $(p(\vec{x}_0, t), I)$ dengan $\vec{x}_0 \in X_0$ dan $t \in \Delta$.

2.4.3 Metode Integrasi Model Taylor

Diberikan sistem kontinu n-dimensional yang didefinisikan oleh $\vec{x} = f(\vec{x})$, dan *time-step* δ , himpunan yang dapat dijangkau untuk horizon waktu terbatas $[0, T]$ dan himpunan awal $X_0 \subseteq \mathbb{R}^n$ yang diover-aproksimasi oleh urutan terbatas dari model Taylor $\mathcal{F}_1, \dots, \mathcal{F}_N$. Akan dihitung flowpipe over-aproksimasi sebagai model Taylor (TM) untuk sistem kontinu non linier. Hal tersebut juga disebut flowpipe TM. Dengan sistem kontinu n-dimensional non-linier $S : \vec{x} = f(\vec{x}, t)$ dan interval atau himpunan awal TM $X_0 \subseteq \mathbb{R}^n$. Asumsikan bahwa besar N *time-step* diberikan oleh $\delta_1, \dots, \delta_N$, lalu untuk $1 \leq i \leq N$, TM flowpipe ke- i adalah bentuk $\mathcal{F}_i(\vec{x}_0, t) = (p_k(\vec{x}_0, t), I)$ dengan $\vec{x}_0 \in X_0$ dan $t \in [0, \delta_i]$. Ini adalah over-aproksimasi dari solusinya $\varphi_f(\vec{x}_0, \sum_{j=1}^{i-1} \delta_j + t)$, yaitu,

$$\varphi_f(\vec{x}_0, \sum_{j=1}^{i-1} \delta_j + t) \in p_k(\vec{x}_0, t) + I$$

untuk semua $\vec{x}_0 \in X_0$, dan $t \in [0, \delta_i]$. Flowpipe TM mungkin berbeda orde. Metode komputasi TM flowpipes disebut integrasi TM. Diberikan uraian sebagai berikut [12]:

Himpunan awal X_l dalam *time-step* ke- i adalah X_0 saat $i=1$, atau diberikan oleh TM $\mathcal{F}_{i-1}(\vec{x}_0, \delta_{i-1})$ untuk $1 \leq i \leq N$. Dilakukan 3 langkah berikut untuk menghitung flowpipe TM \mathcal{F}_i ke- i . Dengan orde TM adalah k .

1. Menghitung polinomial Taylor p_k untuk flow maju (*forward flow*) $\varphi_f(X_l, t)$ hingga orde k di t .
Polinomial p_k dapat berasal dari $\varphi_f(X_l, t)$ sebagai

berikut,

$$p_k(\vec{x}_l, t) = \vec{x}_l + \text{Trunc}_{k-1}(\mathcal{L}_f(\varphi_f(X_l, t)))|_{t=0} \cdot t + \dots \\ + \frac{1}{k!} \text{Trunc}_0(\mathcal{L}_f^k(\varphi_f(X_l, t)))|_{t=0} \cdot t^k$$

dimana, $\vec{x}_l \in X_l$. Dengan operasi $\text{Trunc}_j(p)$ dan perhitungan *Lie Derivative* sebagai berikut,

Definisi 2.4.3.1. [12]

Diberikan ODE $\vec{x} = f(\vec{x})$ dengan n -variabel, *Lie Derivative* dari fungsi diferensial $g(\vec{x}, t)$ yaitu f , didefinisikan dengan,

$$\mathcal{L}_f(g) = \sum_{i=1}^n \left(\frac{\partial g}{\partial x_i} \cdot f_i \right) + \frac{\partial g}{\partial t}$$

Dimana f_i adalah komponen ke- i dari f . Jika g adalah k terdiferensiasi, *Lie Derivative* orde tinggi didefinisikan,

$$\mathcal{L}_f^{m+1}(g) = \mathcal{L}_f(\mathcal{L}_f^m(g))$$

untuk $m=1, 2, \dots, k-1$.

2. Evaluasi interval sisa aman I untuk polinomial Taylor $p_k(\vec{x}_l, t)$ dalam $t \in [0, \delta]$.

Interval I adalah cukup jika operator Picard,

$$\mathbb{P}_f(g(\vec{x}_l, t)) = \vec{x}_l + \int_0^t f(g(\vec{x}_l, s), s) ds$$

Adalah kontraktif dalam $(\overline{p_k}, I)$. Untuk menemukan interval I yang seperti itu, kita dapat terlebih dahulu memberikan perkiraan dan memperbesarnya secara berurutan hingga didapatkan interval kontraktif. Setelah itu, Interval dapat disempurnakan dengan berulang kali menerapkan operator Picard [12].

3. Hitung flowpipe ke- i \mathcal{F}_i dengan mengevaluasi $(p_k(X_l, t), I)$.

Model Taylor \mathcal{F}_i didapatkan dengan mensubstitusi X_l pada \vec{x}_l di (p_k, I) oleh orde k aritmatika model Taylor.

Pengaplikasian tahap-tahap diatas dilakukan pada contoh berikut ini :

Contoh 2.4.1. Diberikan ODE berdimensi 2

$$I = \begin{pmatrix} \dot{x} \\ \dot{y} \end{pmatrix} = \begin{pmatrix} 1 + y \\ -x^2 \end{pmatrix}$$

akan dihitung flowpipe TM \mathcal{F}_i ke-1 dengan orde TM bernilai 4. Untuk tahap pertama, dihitung polinomial model Taylor orde 4 menggunakan operasi *Lie Derivative* dengan *Truncation*. Sebagai berikut:

$$k = 4, i = 1, \dots, 4$$

$$\begin{aligned} Trunc_{k-i} \left(\mathcal{L}_f^i \begin{pmatrix} x \\ y \end{pmatrix} \right) &= Trunc_3 \left(\mathcal{L}_f \begin{pmatrix} x \\ y \end{pmatrix} \right) = \begin{pmatrix} 1 + y \\ -x^2 \end{pmatrix} \\ Trunc_2 \left(\mathcal{L}_f^2 \begin{pmatrix} x \\ y \end{pmatrix} \right) &= Trunc_2 \left(\mathcal{L}_f \begin{pmatrix} 1 + y \\ -x^2 \end{pmatrix} \right) = \begin{pmatrix} -x^2 \\ -2x - 2xy \end{pmatrix} \\ \mathcal{L}_f(1 + y) &= (0)(1 + y) + (1)(-x^2) = -x^2 \\ \mathcal{L}_f(-x^2) &= (-2x)(1 + y) + (0)(-x^2) = -2x - 2xy \\ Trunc_1 \left(\mathcal{L}_f^3 \begin{pmatrix} x \\ y \end{pmatrix} \right) &= Trunc_1 \left(\mathcal{L}_f \begin{pmatrix} -x^2 \\ -2x - 2xy \end{pmatrix} \right) = \begin{pmatrix} -2x \\ -2 - 4y \end{pmatrix} \\ \mathcal{L}_f(-x^2) &= (-2x)(1 + y) + (0)(-x^2) = -2x - 2xy \\ \mathcal{L}_f(-2x - 2xy) &= (-2 - 2y)(1 + y) + (-2)(-x^2) \\ &= -2 - 4y - 2y^2 + 2x^3 \\ Trunc_0 \left(\mathcal{L}_f^4 \begin{pmatrix} x \\ y \end{pmatrix} \right) &= Trunc_0 \left(\mathcal{L}_f \begin{pmatrix} -2x \\ -2 - 4y \end{pmatrix} \right) = \begin{pmatrix} -2 \\ 0 \end{pmatrix} \\ \mathcal{L}_f(-2x) &= (-2)(1 + y) + (0)(-x^2) = -2 - 2y \\ \mathcal{L}_f(-2 - 4y) &= (0)(1 + y) + (-4)(-x^2) = 4x^2 \end{aligned}$$

Kemudian didapatkan ekspansi Taylor orde 4:

$$\begin{aligned}
 p_4(\vec{x}_l, \vec{y}_l, t) &= \begin{pmatrix} x_l + (1 + y_l)t + \frac{1}{2!}(-x_l^2)t^2 + \dots \\ y_l + (-x_l^2)t + \frac{1}{2!}(-2x_l - 2x_ly_l)t^2 + \dots \end{pmatrix} \\
 &= \begin{pmatrix} \dots \frac{1}{3!}(-2x_l)t^3 + \frac{1}{4!}(-2)t^4 \\ \dots \frac{1}{3!}(-2 - 4y_l)t^3 + \frac{1}{4!}(0)t^4 \end{pmatrix} \\
 &= \begin{pmatrix} x_l + t + y_lt - \frac{1}{2}(x_l^2)t^2 - \frac{1}{3}(x_l)t^3 - \frac{1}{12}t^4 \\ y_l - (x_l^2)t - (x_l - x_ly_l)t^2 - \left(\frac{1}{3} + \frac{2}{3}y_l\right)t^3 \end{pmatrix}
 \end{aligned}$$

Diasumsikan himpunan kondisi awal diberikan oleh $\{(x_l, y_l) \mid x_l \in [-1, 1] \wedge y_l \in [-0.5, 0.5]\}$ dan *step size* = 0.02. Selanjutnya akan dihitung ekstensi model Taylor dan diasumsikan estimasi sisanya (*remainder estimation*)

$$J_0 = \begin{pmatrix} [-0.1, 0.1] \\ [-0.1, 0.1] \end{pmatrix}$$

untuk Order 4 Taylor Expansion

$$p_4(\vec{x}_l, \vec{y}_l, t) = \begin{pmatrix} x_l + t + y_lt - \frac{1}{2}x_l^2t^2 - \frac{1}{3}x_lt^3 - \frac{1}{12}t^4 \\ y_l - x_l^2t - (x_l - x_ly_l)t^2 - \left(\frac{1}{3} + \frac{2}{3}y_l\right)t^3 \end{pmatrix}$$

Berikut ekstensi TM orde 4 dari operasi *picard*:

$$\begin{aligned}
\mathcal{P}_f((p_4, J_0)) &= \begin{pmatrix} x_l \\ y_l \end{pmatrix} + \int_0^t f(g(\vec{x}_l, s), s) ds \\
&= \begin{pmatrix} x_l \\ y_l \end{pmatrix} + \int_0^t \begin{pmatrix} 1+y \\ -x^2 \end{pmatrix} ds \\
&= \begin{pmatrix} x_l \\ y_l \end{pmatrix} + \int_0^t \begin{pmatrix} 1 + (y_l - x_l^2 s - (x_l - x_l y_l) s^2 - \dots) \\ -(x_l + s + y_l s - \frac{1}{2} x_l^2 s^2 - \dots) \end{pmatrix} \\
&\quad \left(\dots \begin{pmatrix} \frac{1}{3} + \frac{2}{3} y_l \\ \dots \frac{1}{3} x_l s^3 - \frac{1}{12} s^4, [-0.1, 0.1] \end{pmatrix} s^3, [-0.1, 0.1] \right) ds \\
&= \begin{pmatrix} x_l \\ y_l \end{pmatrix} + \left(\int_0^t \begin{pmatrix} (1 + y_l - x_l^2 s - x_l s^2 - \frac{1}{3} s^3) ds, \dots \\ (-x_l^2 - 2x_l s - s^2 - 2y_l s^2) ds, \dots \end{pmatrix} \right) \\
&\quad \left(\left(\dots (Int(x_l y_l s^2 - \frac{2}{3} y_l s^3) + [-0.1, 0.1]). [0, 0.02] \right) \right) \\
&\quad \left(\left(\dots (Int(x_l^3 s^2 + \frac{2}{3} x_l^2 s^3) + \frac{1}{6} x_l s^4 + x_l^2 s^3 + \frac{2}{3} x_l s^4 + \dots) \right) \right) \\
&\quad \left(\left(\dots \frac{1}{6} s^5 - y_l^2 s^2 + x_l^2 y_l s^3 + \frac{2}{3} x_l y_l s^4 + \frac{1}{6} y_l s^5 - \dots \right) \right) \\
&\quad \left(\left(\dots \frac{1}{4} x_l^4 s^4 - \frac{1}{3} x_l^3 s^5 - \frac{1}{12} x_l^2 s^6 - \frac{1}{9} x_l s^6 - \dots \right) \right) \\
&\quad \left(\left(\dots \frac{1}{18} x_l s^7 - \frac{1}{144} s^4 + [-0.01, 0.01]. [0, 0.02] \right) \right) \\
&= \begin{pmatrix} x_l \\ y_l \end{pmatrix} + \begin{pmatrix} t + y_l t - \frac{x_l t^2}{2} - \frac{x_l t^3}{3} - \frac{t^4}{12}, \dots \\ -x_l^2 t - x_l t^2 - x_l y_l t^2 - \frac{t^3}{3} - \frac{2y_l t^3}{3}, \dots \end{pmatrix} \\
&\quad \left(\dots [-0.0022, 0.0022] \right) \\
&\quad \left(\dots [-0.00021, 0.00021] \right)
\end{aligned}$$

$$= \begin{pmatrix} x_l + t + y_l t - \frac{x_l t^2}{2} - \frac{x_l t^3}{3} - \frac{t^4}{12}, \dots \\ y_l - x_l^2 t - x_l t^2 - x_l y_l t^2 - \frac{t^3}{3} - \frac{2y_l t^3}{3}, \dots \\ \dots[-0.0022, 0.0022] \\ \dots[-0.00021, 0.00021] \end{pmatrix}$$

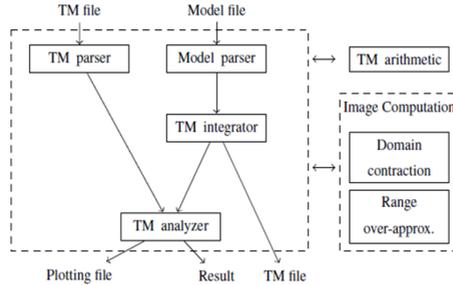
Dari perhitungan diatas, didapatkan hasil operator *picard* pada (p_4, J_0) merupakan terdapat dalam J_0 yaitu,

$$\begin{pmatrix} [-0.0022, 0.0022] \in [-0.1, 0.1] \\ [-0.00021, 0.00021] \in [-0.1, 0.1] \end{pmatrix}$$

maka dapat dikatakan J_0 adalah over-aproksimasi. Untuk mendapatkan interval sisa yang lebih baik, dapat dilakukan *refinement*(penyempurnaan) dengan menggunakan operator *picard* berulang kali. Karena flowpipe model Taylor yang dihitung adalah pada $t \in [0, 0.02]$ yang merupakan *time-step* pada $i = 1$, maka setelah didapatkan $p_4(x_l, t), I$, dilakukan substitusi x_0 pada polinomial model Taylor tersebut. Sehingga menjadi,

$$(p_4(\vec{x}_0, t), I) = \begin{pmatrix} x_0 + t + y_0 t - \frac{x_0 t^2}{2} - \frac{x_0 t^3}{3} - \frac{t^4}{12}, \dots \\ y_0 - x_0^2 t - x_0 t^2 - x_0 y_0 t^2 - \frac{t^3}{3} - \frac{2y_0 t^3}{3}, \dots \\ \dots[-0.0022, 0.0022] \\ \dots[-0.00021, 0.00021] \end{pmatrix}$$

2.5 Flow*



Gambar 2.3: Struktur Flow*

Flow* (flowstar) adalah alat untuk memverifikasi keamanan sistem hybrid, alat untuk analisis *reachability* dari sistem hybrid non linear, dan alat verifikasi untuk sistem cyber-physical. FLOW* mendukung berbagai macam optimasi termasuk ukuran langkah adaptif, pemilihan pilihan aproksimasi adaptif dan pemilihan heuristik dari arah template untuk menggabungkan flowpipes [13]. Flow* berfokus pada verifikasi berbasis *reachability hybrid automaton*. Karena masalah *reachability*, atau bahkan masalah *reachability* yang terbatas pada automata hibrida tidak dapat diputuskan, maka tidak dapat dihitung set yang dapat dicapai secara umum. Oleh karena itu, untuk membuktikan properti keamanan ϕ pada *hybrid automaton*, Flow* mencoba untuk menghitung *over-approximation* dari set *reachable* yang mana tidak menyalahi ϕ .

Perangkat lunak Flow* menggunakan metode integrasi Taylor Model (TM) yang disesuaikan untuk menghitung himpunan overapproximasi yang dapat dicapai untuk sistem kontinyu dan hibrida nonlinier. Serupa dengan metode asli yang diusulkan dalam *Verified Integration of ODEs and Flow Using Differential Algebraic Methods on High-order Taylor Models* oleh M. Berz, solusi ODE, yaitu fungsi diatas

himpunan awal serta variabel waktu, sepanjang interval waktu yang dibatasi telah diover-aproksimasi oleh TM di Flow*, dan oleh karena itu membentuk over-approximasi dari set yang bisa dicapai di sana. Kami juga menyebut TM ini dengan TM flowpipe. Selain itu, untuk mengurangi akumulasi overestimasi selama pekerjaan integrasi, alat tersebut secara simbolis mewakili sisa aliran arus N sebelumnya untuk beberapa $N > 0$ [7].

Alat Flow* merupakan alat untuk menghasilkan flowpipes menggunakan model Taylor (TMs). Flow* juga memeriksa apakah flowpipe itu memotong himpunan yang tidak aman (*unsafe set*) dan mengeluarkan visualisasi dari himpunan state yang dapat dicapai menggunakan polihedral over-aproksimasi dari flowpipes TM yang dihitung [12]. Pada Flow*, memerlukan himpunan keadaan awal yang harus dibatasi [7]. Perangkat lunak Flow* tersedia di *flowstar.org*. Alat ini terdiri dari komponen utama sebagai berikut:

- (a) Sebuah *library* aritmatika dasar untuk interval dan model Taylor,
- (b) Integrator model Taylor untuk menghitung flowpipes di bawah dinamika kontinu,
- (c) *Library* model Taylor untuk menangani invarian mode, *jump guards* dan pembuktian sifat keamanan (*safety properties*).

Halaman ini sengaja dikosongkan.

BAB III METODE PENELITIAN

Pada bab ini dijelaskan mengenai metode yang digunakan dalam Tugas Akhir agar dapat mencapai tujuan yang telah ditetapkan sebelumnya. Langkah-langkah pengerjaan Tugas Akhir ini, yaitu sebagai berikut :

3.1 Tahap Penelitian

Berikut adalah tahap-tahap yang dilakukan dalam pengerjaan Tugas Akhir ini.

1. Studi Literatur

Pada tahap ini dilakukan studi literatur yang berkaitan dengan verifikasi model dinamik bakteriofag dengan bakteri resisten dan bakteri debris terhadap properti *safety*. Selain itu, studi literatur mengenai perangkat lunak Flow* untuk memperdalam pemahaman mengenai topik penelitian. Studi literatur diperoleh dari buku, jurnal, tugas akhir atau tesis yang berkaitan dengan permasalahan, Maupun artikel dari internet.

2. Pendefinisian Permasalahan *Reachability*

Pada tahap ini dilakukan pendefinisian masalah *reachability* sebuah sistem. Definisi yang dilakukan antara lain penulisan model beserta parameternya, himpunan awal, *time-horizon*, himpunan target (*unsafe*), dan lain-lain. Persamaan dinamika kontinu yang sudah ditentukan harus didefinisikan dengan skema integrasi pada perangkat lunak. Pendefinisian permasalahan ini diterapkan dengan file model (.model)

untuk selanjutnya dapat dijalankan oleh sebuah perangkat lunak.

3. Perhitungan Himpunan Terjangkau dan Verifikasi

Dalam tahap ini, digunakan sebuah alat yang bernama Flow* (flowstar) untuk menjalankan perhitungan *reachability*. Dari proses ini akan dihasilkan beberapa file output. Selain itu, juga akan digunakan MATLAB untuk mengetahui plot dari flowpipe yang dihasilkan oleh Flow*.

4. Analisis Hasil

Pada tahap ini dilakukan analisis terhadap hasil dari proses perhitungan *reachability*. Analisis terhadap kondisi tidak aman dan kondisi target. Dan analisa tersebut juga digunakan untuk mengetahui dosis yang dapat direkomendasikan pada terapi fag yang diimplementasikan oleh model matematika dinamik bakteriofag dengan bakteri resisten dan bakteri debris.

5. Penarikan Kesimpulan dan Saran

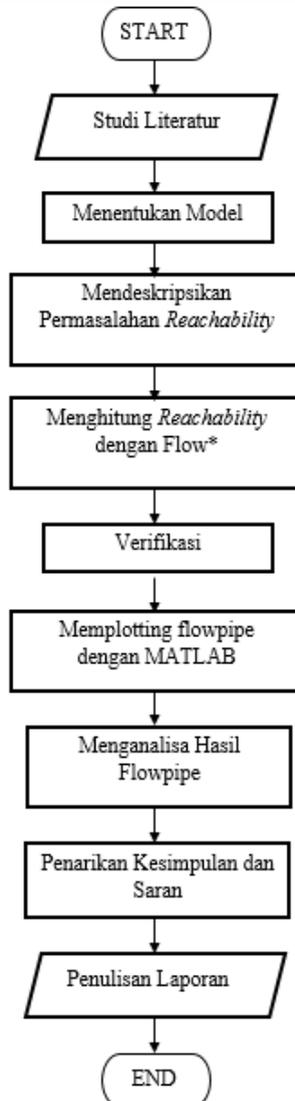
Dalam tahap ini ditarik kesimpulan dari hasil penelitian yang didapatkan sebelumnya. Pemberian saran juga dilakukan atas hasil penelitian ini untuk penelitian selanjutnya.

6. Penyusunan Laporan dan Hasil Penelitian

Pada tahap ini penulis menyusun laporan penelitian berdasarkan proses dan hasil penelitian yang didapatkan dalam menyelesaikan tugas akhir ini

3.2 Alur Penelitian

Berikut adalah alur penelitian yang dilakukan dalam pengerjaan Tugas Akhir ini.



Gambar 3.1: Diagram Metode Penelitian Tugas Akhir.

Halaman ini sengaja dikosongkan.

BAB IV

ANALISIS DAN PEMBAHASAN

Pada bab ini, dibahas analisis reachabilitas dan verifikasi sifat keamanan dari sistem kontinu non-linier yaitu model deterministik bakteriofag. Dua hal tersebut dilakukan dengan menghitung overaproksimasi untuk himpunan yang dapat dicapai oleh sistem. Untuk verifikasi keamanan, apabila himpunan over aproksimasi menjangkau target dan tidak berisi himpunan tidak aman, maka dapat disimpulkan bahwa sistem aman dan terapi berhasil. Dalam menyelesaikan permasalahan tersebut, digunakan perangkat lunak Flow*.

4.1 Terapi Fag

Bakteriofag atau dapat disebut juga “fag” adalah virus yang menyerang bakteri. Tujuan terapi fag adalah untuk mematikan bakteri yang tidak diinginkan. Bakteriofag diklasifikasikan menjadi 2 fase, yaitu litik dan lisogenik. Fag lisogenik tidak digunakan dalam terapi fag karena interaksi yang terjadi antara bakteriofag dan bakteri saling menguntungkan. Dan sedangkan fag litik digunakan dalam terapi fag karena membunuh bakteri *host* (inang). Fase litik dimulai ketika fag dan bakteri inang bersentuhan. Fag berikatan dengan reseptor dinding sel bakteri dalam proses yang dikenal sebagai adsorpsi. Setelah itu, fag menyuntikkan genomnya ke dalam sel bakteri, sehingga menghasilkan bakteri yang terinfeksi. Didalam bakteri terjadi pembentukan komponen fag baru dan siap untuk dilisiskan dari sel inang. Perkembangbiakan bakteriofag ini biasanya membutuhkan waktu 20 menit hingga satu jam. Selanjutnya, fag memecah

(*burst*) sel sehingga 50 sampai 200 bakteriofag yang baru diproduksi menyebar ke seluruh sistem inang. Setelah ledakan terjadi, bakteri mati dan siklus tersebut akan berulang pada bakteri lainnya.

Pada Tugas Akhir ini, disajikan studi kasus infeksi bakteri patogenik *Escherichia Coli* yang dapat diatasi dengan bakteriofag (virus) T-4. *Escherichia Coli* adalah bakteri yang ditemukan di lingkungan, makanan, manusia dan hewan. Kebanyakan E. Coli tidak berbahaya tetapi beberapa bersifat patogen dan diantaranya dapat menyebabkan infeksi. Salah satunya adalah bakteri EHEC (*Enterohemorrhagic Escherichia coli*) yang banyak ditemui di daging yang kurang matang, sayuran dan buah yang terkontaminasi. Jika makanan yang terkontaminasi dicerna, maka tidak menutup kemungkinan akan terserang diare, *colin haemorrhagic* (HC), dan *haemorrhagic uraemis syndrome* (HUS). Karena fag T4 dapat membunuh bakteri *Escherichia coli*, fag T4 telah digunakan untuk pengobatan/ terapi. Dibawah ini, dirancang skenario simulasi perawatan dan kondisi target beserta kondisi tidak amannya.

Skenario Simulasi Perawatan: Diasumsikan skenario berikut untuk pengaturan verifikasi bakteriofag yang dimasukkan dalam tubuh.

1. Dibentuk beberapa rasio perbandingan antara bakteriofag dan bakteri ($100 : 1$, $55 \cdot 10^4 : 1$, $45 \cdot 10^5 : 1$) untuk menentukan dosis awal yang akan direkomendasikan, yaitu:

$$a \ 10^3 PFU/ml - 2 \cdot 10^3 PFU/ml$$

$$b \ 55 \cdot 10^5 PFU/ml - 552 \cdot 10^4 PFU/ml$$

$$c \ 45 \cdot 10^6 PFU/ml - 50 \cdot 10^6 PFU/ml$$

pada waktu $t = 0$.

2. Pengamatan terapi fag pada pasien dilakukan selama 5 jam. Dilakukan selama 5 jam, untuk menghindari terjadinya *colin haemorrhagic* (HC) dan *haemorrhagic uraemis syndrome* (HUS).
3. Dosis infeksi bakteri EHEC (*Enterohemorrhagic E. Coli*) awal pasien berada di kisaran 10 – 100 organisme.

Akan dipertimbangkan target yang harus dicapai dan sifat keamanan yang penting untuk terapi ini.

1. Jumlah bakteri didorong ke tingkat yang cukup rendah, yaitu bernilai habis (0-1 organisme) , untuk mencapai hasil yang baik.
2. Bakteriofag T4 tidak akan pernah lebih sedikit dari bakteri yang ada dalam tubuh selama waktu pengamatan. Karena jika bakteriofag habis, maka bakteri akan berkembang menjadi lebih banyak.

4.2 Model Deterministik

Dalam Tugas Akhir ini, digunakan model matematika untuk mengaplikasikan interaksi bakteriofag dengan bakteri. Pada model matematika persamaan (2.1), diasumsikan [1]:

$$k_1(P) = k_2(P) = \frac{kP}{1 + c_1P} \quad (4.1)$$

$$k_3(P) = 0 \quad (4.2)$$

$$f(P) = \frac{f_1P}{1 + P} \quad (4.3)$$

$$r_i(B + Q + L + I + a_iR) = r_{ii} \left[1 - \frac{B + R + L + I + a_iR}{C} \right] \quad (4.4)$$

Dimana r_{ii} merupakan tingkat pertumbuhan masing-masing sel bakteri yang rentan, sel bakteri resisten, dan

bakteri-fag kompleks. kemudian C adalah daya dukung bakteri. Ketika daya dukung meningkat, maka kepunahan bakteriofag menurun. Tingkat mutasi bakteri rentan ke bakteri tahan saat adanya bakteriofag ditunjukkan dengan f_1 . Selanjutnya, k adalah tingkat adsorpsi bakteriofag dengan bakteri. Dan c_1 menunjukkan rasio tingkat adsorpsi bakteriofag terhadap waktu injeksi fag. Sehingga persamaan (2.1) menjadi:

$$\begin{aligned}
\frac{dB}{dt} &= r_{11} \left[1 - \frac{(B + Q + L + I + a_1 R)}{C} \right] B - \left[\frac{f_1 P}{(1 + P)} \right] B \\
&\quad - kBP - \mu B \\
\frac{dR}{dt} &= r_{22} \left[1 - \frac{(B + Q + L + I + a_2 R)}{C} \right] R + \left[\frac{f_1 P}{(1 + P)} \right] B \\
&\quad - \mu B \\
\frac{dQ}{dt} &= r_{33} \left[1 - \frac{(B + Q + L + I + a_3 R)}{C} \right] Q + kBP - \delta_1 Q \\
&\quad - \mu Q \\
\frac{dL}{dt} &= \delta_1 Q - \alpha L - \mu B \\
\frac{dI}{dt} &= \alpha L - aI - \mu I \\
\frac{dD}{dt} &= \mu(B + Q + L + I) + aI - \eta D \\
\frac{dP}{dt} &= NaI + \delta_2 Q - kBP - \left[\frac{kP}{(1 + c_1 P)} \right] (L + 1) \\
&\quad - \left[\frac{kP}{(1 + c_1 P)} \right] Q - \gamma P
\end{aligned} \tag{4.5}$$

Diagram kompartemen dari model dinamik bakteri dengan bakteriofag diatas disajikan pada Lampiran A. Diagram

kompartemen tersebut menggambarkan hubungan antar bakteri dan bakteri dengan bakteriofag. Pada terapi fag ini, perlu diperhatikan beberapa hal sebagai berikut:

1. Pada terapi fag yang akan dilakukan, bakteriofag hanya tidak akan berinteraksi dengan bakteri resisten. Oleh karena itu bakteri resisten hanya akan diamati untuk mendapatkan jumlahnya agar selanjutnya diatasi dengan terapi koktail fag.
2. Pengamatan model dinamik bakteriofag berfokus pada bakteri resisten dan bakteriofag karena interaksi keduanya yang menjadi tolak ukur keberhasilan terapi fag.

4.3 Menghitung Flowpipe Model Taylor

Pada perhitungan flowpipe TM, akan digunakan teknik langkah waktu adaptif (*adaptive time-step*). Secara adaptif, akan dilakukan perubahan step-size dalam tugas integrasi model Taylor dengan akurasi yang ditentukan akan tetap dipenuhi. Untuk non-polinomial ODE, terlebih dahulu menghitung model Taylor untuk bidang vektor. Berikut ini algoritma dari integrasi model Taylor dengan menggunakan langkah waktu adaptif:

1. Menghitung parameter preconditioning \vec{c}_l dan A_l dengan menggunakan metode preconditioning identitas.
2. Menghitung polinomial Taylor orde-k / *preconditioned Taylor expansion* berdasarkan parameter. Pertama, menentukan $p_{\vec{c}}(t)$ yaitu polinomial Taylor orde-k dari $\varphi_f(\vec{c}_l, t)$ dan $p_{\vec{y}}(\vec{y}_l, t)$ yaitu polinomial Taylor orde-k dari $\varphi_h(\vec{y}_l, t)$.

$$\begin{aligned}
p_{\vec{y}}(\vec{y}_l, t) &= y_l + p_h(h, t) \\
&= y_l + \left(A_l^{-1} \cdot \left(\frac{d\varphi_f(\vec{x}_l, t)}{dt} - \frac{dp_{\vec{c}}}{dt} \right) \right) \\
&= y_l + \left(A_l^{-1} \cdot \left(f(\vec{x}, t) - \frac{dp_{\vec{c}}}{dt} \right) \right) \\
&= y_l + \left(A_l^{-1} \cdot \left(f(p_{\vec{c}} + A_l \cdot \vec{y}, t) - \frac{dp_{\vec{c}}}{dt} \right) \right)
\end{aligned}$$

kemudian didapatkan ekspansi Taylor p_l , sebagai berikut

$$p_k(\vec{y}_l, t) = p_{\vec{c}}(t) + A_l \cdot p_{\vec{y}}(\vec{y}_l, t)$$

3. Mengevaluasi interval sisa yang aman untuk polinomial Taylor orde- k . Untuk input *time-step* pertama, diambil β yang merupakan ukuran langkah (*step-size*) terbesar. Untuk looping selanjutnya, apabila *time-step* $< \beta$ maka dilakukan perbesaran ukuran langkah. Yaitu dengan

$$\delta = \min\{\lambda_{\uparrow}, \delta, \beta\}$$

Dimana *min* adalah elemen paling kecil pada suatu himpunan. Karena diatas merupakan himpunan yang terbatas, maka minimum akan didapatkan dengan mengambil elemen paling kecil antara $\lambda_{\uparrow}, \delta$ dan β . Dan untuk infimum dari himpunan tersebut adalah elemen terbesar yang kurang dari atau sama dengan semua elemen. Jadi untuk kasus diatas, dapat dikatakan infimumnya adalah sama dengan minimum.

Apabila I_{max} bukan merupakan interval sisa aman

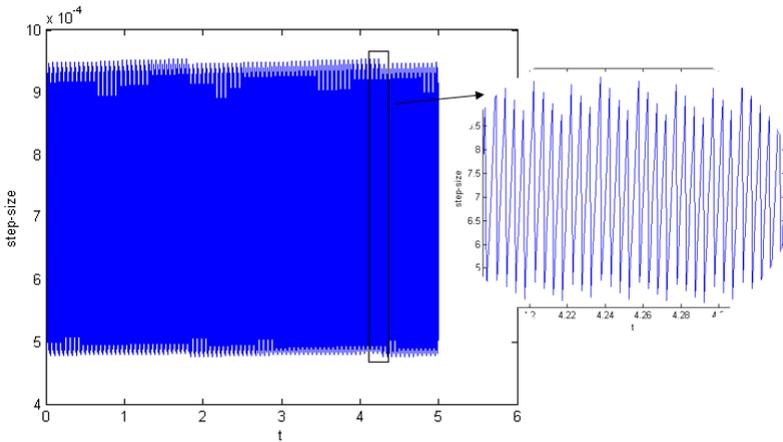
(*safe remainder interval*) untuk p_k pada waktu $[0, \delta]$, dilakukan pengurangan ukuran langkah, sebagai berikut

$$\delta = \lambda_{\downarrow} \cdot \delta$$

Jika $\delta < \alpha$, maka dapat disimpulkan ukuran langkah tersebut diluar rentang $[\alpha, \beta]$ dan perhitungan selanjutnya dihentikan dan dinyatakan gagal. Apabila I_{max} merupakan interval sisa aman, maka didapatkan $I = I_{max}$. Interval sisa dikatakan aman apabila operator *picard* kontraktif pada (p_k, I) dan dikatakan kontraktif apabila sisa dari ekstensi model Taylor (menggunakan operasi *picard*) terdapat pada interval sisa yang telah diestimasi sebelumnya

4. Perbaiki interval sisa I dengan menggunakan iterasi operator *picard* (iterasi *picard*). Perbaikan ini dilakukan sampai tidak ada perbaikan besar pada perbaikan terakhir dari I .
5. Menghitung $\mathcal{F}_i = (p_k(X_l, t), I)$ dengan aritmatika TM orde-k, dengan t merupakan δ .

Algoritma tersebut akan terus berjalan sampai horizon waktu yang ditetapkan. Dan himpunan *reachable* dari time-step satu dengan yang lain digabungkan secara berurutan. Untuk *local initial set* selanjutnya, adalah $(p_k(X_l, t), I)$. Perubahan *time-step* selama horizon waktu $[0, 5]$ dapat dilihat pada Gambar 4.1.



Gambar 4.1: Perubahan *step-size* Pada Skenario 1

4.4 Pendeskripsian Permasalahan Reachabilitas

Flow* dapat berjalan dengan bentuk file yang berupa *.model* file. Oleh karena itu, permasalahan reachability dapat digambarkan dalam bentuk file *.model* dengan konten dan spesifikasi sebagai berikut:

4.4.1 Definisi Sistem

Sistem yang digunakan untuk merepresentasikan interaksi bakteriofag dengan bakteri adalah sistem kontinu non linier. dimana sistem tersebut dapat didefinisikan dengan persamaan diferensial biasa dan dituliskan dalam bentuk file *.model* sebagai berikut:

```

nonpoly ode                                # integration scheme
{
  B' = r11*(1-(B+Q+L+I+a1*R)/(C))*B-(f1*P)/(1+P)*B-k*B*P-miu*B
  R' = r22*(1-(B+Q+L+I+a2*R)/(C))*R+(f1*P)/(1+P)*B-miu*R
  Q' = r33*(1-(B+Q+L+I+a3*R)/(C))*Q+k*B*P-chi1*Q-miu*Q
  L' = chi1*Q-alpha*L-miu*L
  I' = alpha*L-a*I-miu*I
  D' = miu*(B+Q+L+I)+a*I-n*D
  P' = N*a*I+chi2*Q-k*B*P-((k*P)/(1+(c1*P)))*(L+I)-((k*P)/(1
+(c1*P)))*Q-k3*D-gamma*P
  t' = 1
}

```

Gambar 4.2: Definisi Sistem pada file *.model*

Spesifikasi tersebut memberitahu perangkat lunak untuk mengintegrasikan persamaan diferensial biasa model matematika bakteriofag dengan bakteri resisten dan bakteri debris menggunakan skema ODE yang tidak berbentuk polinomial (*nonpoly ODE*).

4.4.2 Kondisi Awal, Target dan Tidak Aman

Pada terapi fag, terdapat target dan sifat keamanan yang akan dilihat keterjangkauannya (*reachable*). Target yang diharapkan akan terjangkau yaitu jumlah bakteri didorong ke tingkat yang cukup rendah. Dan sifat keamanan yang harus diverifikasi adalah Bakteriofag T4 tidak akan pernah habis selama bakteri masih belum punah. Lebih tepatnya, hubungan di antara sistem parameter ditemukan, yang menjamin properti keamanan. Oleh karena itu, diperlukan himpunan kondisi awal yang dapat diberikan dengan interval atau model Taylor (TM). Untuk interval himpunan awal, disini perlu menentukan rentang interval untuk setiap variabel state. Selanjutnya, himpunan kondisi tidak aman juga diperlukan dan dapat didefinisikan oleh sistem ketidaksetaraan polinomial. Berikut himpunan kondisi awal, target dan tidak aman berupa interval untuk model matematika yang merepresentasikan terapi fag.

```

init
{
  B in [10,100]
  R in [0,0]
  Q in [0,0]
  L in [0,0]
  I in [0,0]
  D in [0,0]
  P in [1000,2000]
}

```

Gambar 4.3: Himpunan Kondisi Awal pada file *.model* (Skenario 1)

- (a) Himpunan kondisi awal : Himpunan keadaan awal yang dapat diterima oleh B, R, Q, L, I, D, P .

Skenario 1

$$B(0) \in [10, 100]; P(0) \in [10^3, 2.10^3]$$

Skenario 2

$$B(0) \in [10, 100]; P(0) \in [55.10^5, 552.10^4]$$

Skenario 3

$$B(0) \in [10, 100]; P(0) \in [45.10^6, 50.10^6]$$

Dengan kondisi awal yang lain adalah

$$D(0) \in [0, 0]; Q(0) \in [0, 0]; R(0) \in [0, 0];$$

$$L(0) \in [0, 0]; I(0) \in [0, 0]$$

- (b) Himpunan kondisi target : Himpunan dari kondisi yang seharusnya terjangkau oleh sistem.

$$t = 5, B(t) \in [0, 1]$$

- (c) Himpunan kondisi tidak aman : Himpunan dari kondisi yang seharusnya tidak terjangkau oleh sistem.

$$t \in [0, 5], P(t) - B(t) < 1$$

4.4.3 Pengaturan Reachabilitas

Untuk mendapatkan hasil verifikasi, dibawah ini telah dipertimbangkan pengaturan komputasi pada perangkat lunak Flow*.

```

setting
{
  adaptive steps {min 0.0002 , max 0.01} # time step-size
  time 5 # time horizon
  remainder estimation 1e-5 # remainder estimation
  identity precondition # preconditioning method
  matlab octagon P,B # 2D projection for plotting
  fixed orders 4
  cutoff 1e-8 # cutoff threshold is [-1e-8,1e-8]
  precision 100
  output bakteriTA3 # name of the output files
  print on # print out the computation steps
}

```

Gambar 4.4: Permasalahan Reachabilitas (Skenario 1)

TM dari alur maju (*forward flow*) untuk horizon waktu $[0,5]$ dihitung dengan menetapkan ukuran adaptif *time step* (δ) dalam $[0.0002,0.01]$, yang berarti *Time step-size* telah ditetapkan memiliki nilai minimum yaitu 0.0002 dan nilai maksimum yaitu 0.01 . Untuk order model Taylor (k), digunakan TM order 4 untuk setiap variable statenya. Kinerja Flow* yang baik ditentukan oleh order TM yang besar dan *step-size* yang kecil. Namun semakin besar order TM, maka semakin lama perhitungan yang dilakukan. Selain itu, *remainder estimation* (Ie) pada setiap langkah integrasinya dan dimensinya adalah $[-0.00001,0.00001]$. Pada integrasi TM, digunakan identity preconditioning. Kemudian untuk memvisualisasikan perhitungan flowpipe TM, digunakan proyeksi octagon (segi-delapan) untuk proyeksi 2-dimensinya. *Cutoff threshold* (ε) yang ditetapkan adalah 10^{-8} , yang artinya bagian polinomial yang berada pada rentang $[-10^{-8}, 10^{-8}]$ akan dipindahkan ke bagian sisanya. Dan presisi untuk bilangan floating-point adalah 100. Pada Lampiran B, dilampirkan pengaturan reachabilitas untuk skenario 2 dan 3.

4.5 Pemeriksaan Sifat Keamanan

Tugas komputasi *reachability* dilakukan dengan menggunakan perangkat lunak Flow* dan diplot menggunakan MATLAB R2013a. Flow* menyimpan semua flowpipe model Taylor dengan definisi state space dan perhitungannya pada file flowpipe (.flow). File flowpipe (.flow) telah dilampirkan pada Lampiran B untuk semua skenario. Eksperimen yang dilakukan dibawah menggunakan komputer dengan memori RAM 4GB dan prosesor Intel Core i5.

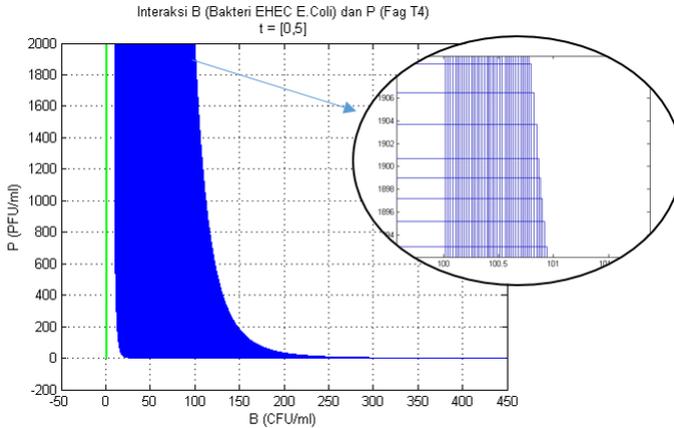
Pada penelitian Tugas Akhir ini, ditampilkan hasil keluaran pada terminal. Selain itu juga akan ditampilkan beberapa plotting yang merupakan visualisasi dari flowpipe model Taylor (TM) dari model bakteriofag yang dihitung. Plot dari himpunan over-aproksimasi nantinya berupa proyeksi 2-dimensi bentuk octagon. Pada analisis reachabilitas dan verifikasi sifat keamanan, digunakan 3 skenario. Skenario tersebut untuk mengetahui apakah dengan dosis tersebut, akan dapat menghindari daerah tidak aman dan mencapai target dalam waktu pengamatan selama 5 jam.

Pemeriksaan pertama yaitu skenario 1, dengan himpunan kondisi awal, himpunan target, dan himpunan tidak aman yaitu:

$$X_0 = \{(B, P) \in \mathbb{R}^7 | B \in [10, 100], P \in [10^3, 2.10^3]\}$$

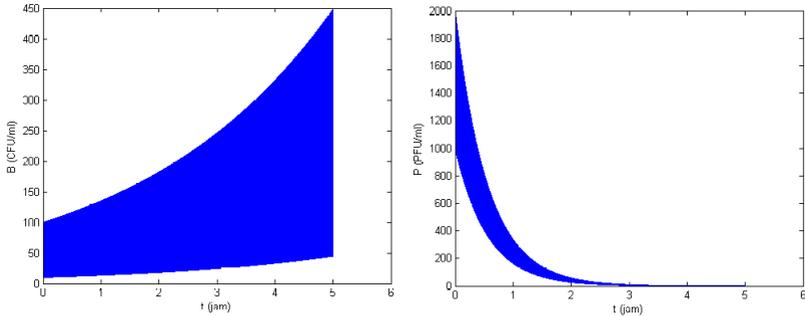
$$B(5) = [0, 1], P(t) - B(t) < 1$$

Setelah dilakukan perhitungan reachabilitas dengan menggunakan Flow* diperoleh hasil pada Gambar 4.5. Dari hasil tersebut tidak ditemui eror dan waktu perhitungan yang dibutuhkan perangkat lunak Flow* adalah 2217 detik dengan jumlah flowpipe over-aproksimasi sebanyak 7270 flowpipe. Dibawah ini plot flowpipe TM dari spesifikasi yang telah disebutkan diatas.



Gambar 4.5: Himpunan Terjangkau (*reachable*) dari B dan P (Skenario 1)

Pada Gambar 4.5, ditunjukkan bahwa semua over-aproksimasi flowpipe dihitung pada rentang waktu $t = [0, 5]$. Dengan nilai awal yang diberikan, laju pertumbuhan bakteri E.Coli rentan mengalami kenaikan dan sebaliknya laju pertumbuhan bakteriofag T4 yang mengalami penurunan seiring waktu. Yang artinya, pertumbuhan bakteri dalam tubuh lebih cepat dari bakteriofagnya. Pada $t = 5$, keterjangkauan (*reachability*) skenario 1 didapatkan bakteri rentan sejumlah $[44.62, 447.9828]$ CFU/ml dan bakteriofag bebas sejumlah $[0.151, 0.3317]$ PFU/ml. Jadi, dapat dikatakan kondisi target (daerah berwarna hijau) tidak terjangkau oleh flowpipe yang telah di-over-aproksimasi sehingga jumlah bakteri belum dapat didorong ke tingkat yang cukup rendah, yaitu bernilai habis $[0, 1]$, untuk mencapai hasil yang baik. Selanjutnya akan diperiksa apakah flowpipe masuk kedalam kondisi yang tidak aman.



Gambar 4.6: Dinamika Bakteri E. Coli dan Fag T4 (Skenario 1)

```
Annisa@DESKTOP-81DQKLS ~/download/flowstar-2.1.0
$ ./flowstar.exe < bakterita3.flow
Safety checking ...
%100
Done.
Time cost of safety verification: 193.187000 seconds.

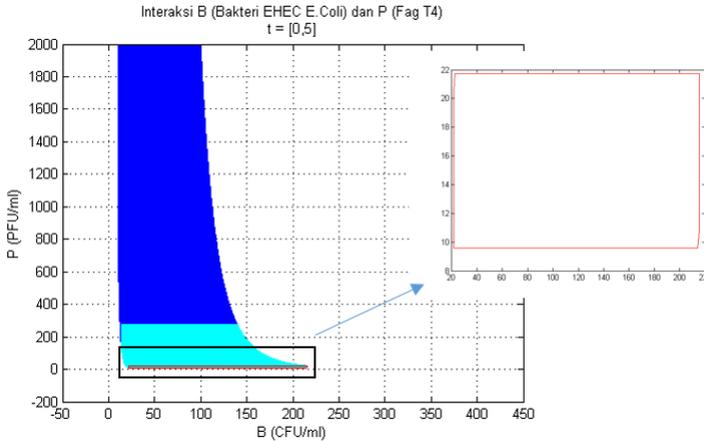
Result of safety verification: UNSAFE

Generating the plot file..
%100
Done.
```

Gambar 4.7: Hasil Verifikasi Keamanan pada Flow* (Skenario 1)

Dengan verifikasi keamanan oleh Flow*, didapatkan hasil UNSAFE karena kondisi tidak aman terjangkau (*reachable*) oleh flowpipe yang telah di-over-approximasi. Pada Gambar 4.8, terlihat flowpipe TM berwarna biru tua, dimana flowpipe TM tersebut tidak memotong daerah tidak aman. Sedangkan yang berwarna biru muda adalah flowpipe TM yang berpotongan dengan daerah yang tidak aman. Dan untuk flowpipe TM pertama yang masuk dalam daerah tidak aman ditunjukkan dengan warna merah. Jadi, Bakteriofag T4 mengalami penurunan dan bakteri mengalami pertumbuhan dalam waktu $t = [0, 5]$ sampai akhirnya bakteriofag T4 dan

bakteri memasuki daerah tidak aman yang seharusnya tidak terjangkau. Skenario 1 tidak direkomendasikan untuk dosis yang nantinya diberikan untuk terapi fag.



Gambar 4.8: Flowpipe Hasil Verifikasi (Skenario 1)

Selanjutnya, Pemeriksaan skenario 2. Yaitu dengan himpunan kondisi awal, himpunan target, dan himpunan tidak aman sebagai berikut:

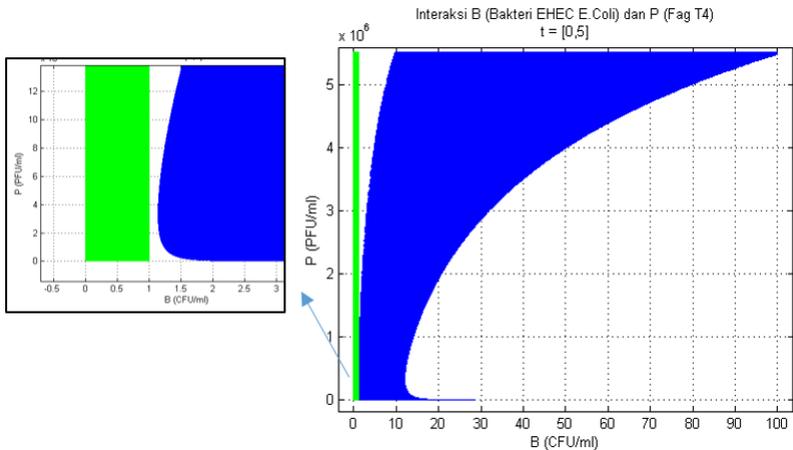
$$X_0 = \{(B, P) \in \mathbb{R}^2 \mid B \in [10, 100], P \in [55 \cdot 10^5, 552 \cdot 10^4]\}$$

$$B(5) = [0, 1], P(t) - B(t) < 1$$

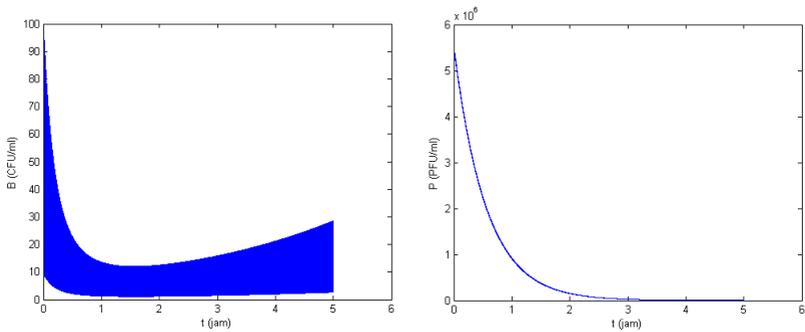
Dibawah ini, Gambar 4.9 merupakan hasil dari perhitungan reachabilitas. Dari hasil tersebut tidak ditemui eror dan waktu perhitungan yang dibutuhkan perangkat lunak Flow* adalah 4632 detik dengan jumlah flowpipe yang telah di-over-approximasi sebanyak 7295 flowpipe. Dibawah ini plot flowpipe dari skenario yang telah disebutkan diatas.

Pada Gambar 4.9, ditunjukkan semua over-approximasi flowpipe dihitung pada rentang waktu $t = [0, 5]$. Dapat dilihat

dari flowpipe TM yang telah diplot bahwa laju pertumbuhan bakteri E.Coli rentan mengalami pergerakan turun naik. Bakteri E.Coli mengalami penurunan, kemudian pada waktu sekitar 1 jam 36 menit mengalami kenaikan.



Gambar 4.9: Himpunan Terjangkau (*reachable*) dari B dan P (Skenario 2)



Gambar 4.10: Dinamika Bakteri E. Coli dan Fag T4 (Skenario

2)

Sedangkan laju pertumbuhan bakteriofag T4 mengalami penurunan terus menerus. Yang artinya, bakteriofag dapat mengurangi keberadaan bakteri namun saat bakteriofag T4 mengalami penurunan jumlah, bakteri mulai berkembangbiak lagi. Pada $t = 5$, keterjangkauan (*reachability*) skenario 2 didapatkan bakteri rentan sejumlah [2.7027, 28.5377] CFU/ml dan bakteriofag bebas sejumlah [680.1361, 701.0155] PFU/ml. Jadi, dapat dikatakan kondisi target masih belum terjangkau oleh flowpipe yang telah diover-aproksimasi sehingga Jumlah bakteri belum dapat didorong ke tingkat yang cukup rendah, yaitu bernilai habis $[0,1]$, untuk mencapai hasil yang baik. Kemudian, seperti skenario 1 akan dilakukan pemeriksaan keamanannya.

```
Annisa@DESKTOP-81DQKLS ~/download/Flowstar-2.1.0
$ ./Flowstar.exe < bakteriTA8.flow
Safety checking ...
%100
Done.
Time cost of safety verification: 243.047000 seconds.

Result of safety verification: SAFE

Generating the plot file...
%100
Done.
```

Gambar 4.11: Hasil Verifikasi Keamanan pada Flow* (Skenario 2)

Dari verifikasi keamanan, didapatkan hasil SAFE karena kondisi tidak aman tidak terjangkau (*not reachable*) oleh flowpipe TM. Jadi selama bakteriofag T4 mengalami penurunan selama $t = [0, 5]$ dan bakteri mengalami turun naik, keduanya tidak memasuki maupun berpotongan dengan kondisi bakteri yang lebih banyak jumlahnya dibanding dengan bakteriofag T4. Skenario 2 tidak direkomendasikan diberikan untuk terapi fag karena tidak mencapai target yang ditentukan dan tidak menutup kemungkinan bakteri akan

terus berkembang dan bakteriofag akan habis.

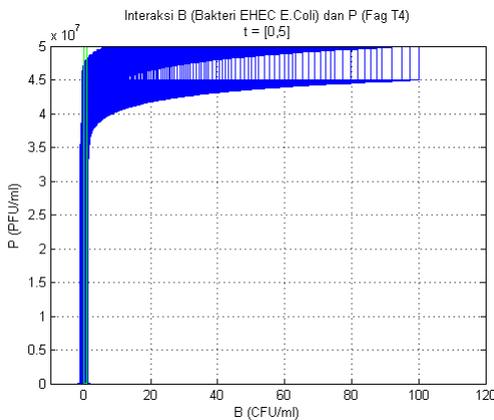
Yang terakhir, Pemeriksaan skenario 3. Yaitu dengan himpunan awal, himpunan target, dan himpunan tidak aman sebagai berikut:

$$X_0 = \{(B, P) \in \mathbb{R}^2 | B \in [10, 100], P \in [45 \cdot 10^6, 50 \cdot 10^6]\}$$

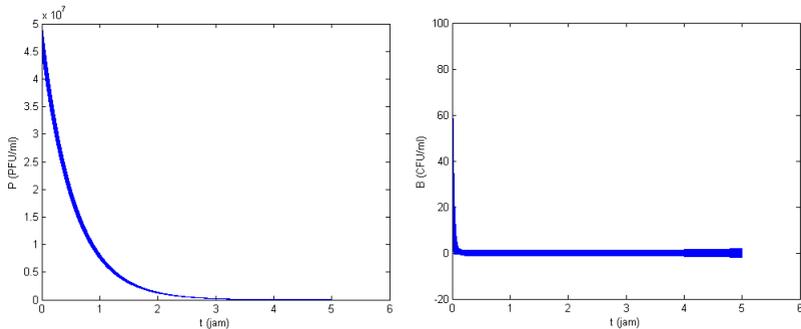
$$B(5) = [0, 1], P(t) - B(t) < 1$$

Dari perhitungan reachabilitas dengan menggunakan Flow* diperoleh hasil pada Gambar 4.12. Pada hasil tersebut tidak ditemui eror dan waktu perhitungan yang dibutuhkan perangkat lunak Flow* adalah 4409 detik dengan jumlah flowpipe over-aproksimasi sebanyak 7595 flowpipe. Dibawah ini adalah plot flowpipe dari skenario yang telah disebutkan diatas.

Pada Gambar 4.12, ditunjukkan semua over-aproksimasi flowpipe dihitung pada rentang waktu $t = [0, 5]$. Dapat dilihat dari flowpipe TM yang telah diplot bahwa bakteri E.Coli rentan mengalami penurunan dan begitu juga dengan bakteriofag yang mengalami penurunan.



Gambar 4.12: Himpunan Terjangkau (*reachable*) dari B dan P (Skenario 3)



Gambar 4.13: Dinamika Bakteri E. Coli dan Fag T4 (Skenario 3)

Pada $t = 5$, keterjangkauan (*reachability*) skenario 3 didapatkan bakteri rentan sejumlah $[-1.867, 1.867]$ CFU/ml dan bakteriofag bebas sejumlah $[5537, 6187]$ PFU/ml. Jadi, dapat dikatakan kondisi target (daerah berwarna hijau) terjangkau oleh flowpipe yang telah di-over-aproksimasi sehingga Jumlah bakteri telah dapat didorong ke tingkat yang cukup rendah, yaitu bernilai habis $[0,1]$ atau mengalami kepunahan. Selanjutnya akan dilakukan verifikasi keamanan untuk mengetahui apakah over-aproksimasi berpotongan dengan kondisi yang tidak aman.

```
Annisa@DESKTOP-81DQKLS ~/download/flowstar-2.1.0
$ ./flowstar.exe < bakteriTA10.flow
Safety checking ...
%100
Done.
Time cost of safety verification: 379.968000 seconds.

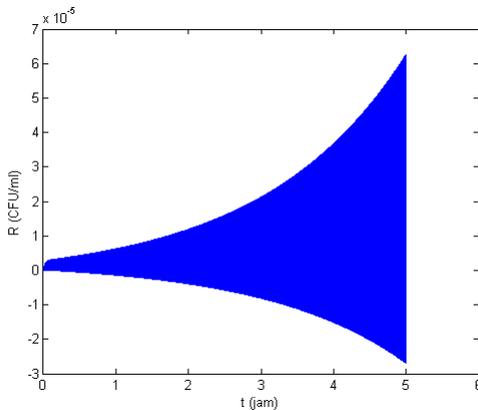
Result of safety verification: SAFE

Generating the plot file...
%100
Done.
```

Gambar 4.14: Hasil Verifikasi Keamanan pada Flow* (Skenario 3)

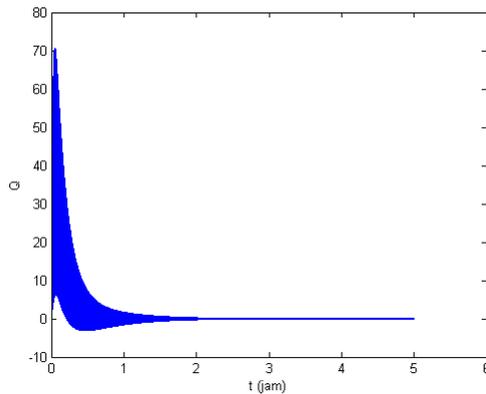
Dengan verifikasi keamanan oleh Flow*, didapatkan hasil SAFE karena kondisi tidak aman tidak terjangkau (*not reachable*) oleh flowpipe TM. Selama bakteri E.Coli mengalami penurunan selama $t = [0, 5]$, bakteriofag T4 juga mengalami penurunan. Keduanya tidak memasuki maupun berpotongan dengan kondisi bakteri yang lebih banyak jumlahnya dibanding dengan bakteriofag T4. Skenario 3 direkomendasikan diberikan untuk keberhasilan terapi fag karena mencapai target yang ditentukan dan tidak menutup kemungkinan bakteri akan terus berkembang dan bakteriofag akan habis.

Dari ketiga hasil verifikasi tersebut, dapat disimpulkan bahwa skenario yang tepat untuk pasien terapi fag adalah skenario 3 dengan dosis fag sebanyak $[4.510^7, 510^7]$ dengan keterjangkauannya (*reachability*) mencapai target $[0, 1]$ setelah 25 menit. Dan dari verifikasi yang dilakukan, permasalahan skenario 3 ini dapat diklasifikasikan sebagai NONLIN-BOUND-TIME-SAFE. Pada Lampiran C, terlampir flowpipe TM beserta integrasi numeriknya.



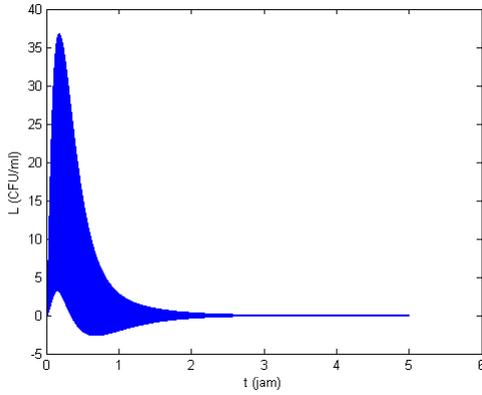
Gambar 4.15: Dinamika Bakteri Resisten

Selanjutnya akan ditampilkan dinamika setiap variabelnya dengan kondisi awal skenario 3. Pada Gambar 4.15 menunjukkan jumlah populasi bakteri resisten setelah 5 jam. Terlihat jumlah populasi yang meningkat dan keterjangkauannya mencapai $6.25 \cdot 10^{-5}$ CFU/ml.

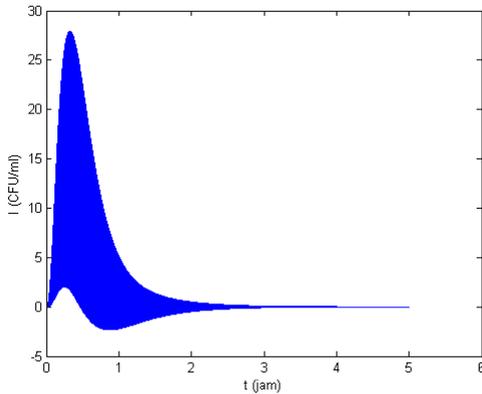


Gambar 4.16: Dinamika Bakteri-Fag Kompleks

Pada Gambar 4.16 menunjukkan jumlah populasi bakteri-fag kompleks setelah 5 jam. Terlihat jumlah populasi yang meningkat kemudian menurun karena seiring menurunnya bakteri rentan dan bakteriofag. Keterjangkauannya mencapai $1.92 \cdot 10^{-3}$ CFU/ml.



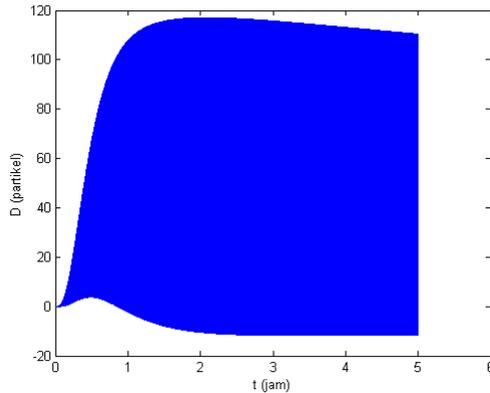
Gambar 4.17: Dinamika Bakteri Laten



Gambar 4.18: Dinamika Bakteri Aktif

Pada Gambar 4.17 menunjukkan jumlah populasi bakteri yang terinfeksi secara perlahan setelah 5 jam. Terlihat jumlah populasi yang meningkat kemudian menurun seperti bakteri-fag kompleks. Hal ini terjadi karena bakteri laten tidak mengalami reproduksi dan hanya bertambah oleh perubahan

dari bakteri-fag kompleks. Keterjangkauannya mencapai $2.44 \cdot 10^{-3}$ CFU/ml. Sama seperti bakteri laten, Gambar 4.18 menunjukkan jumlah populasi bakteri-fag kompleks setelah 5 jam. Bakteri aktif tidak mengalami reproduksi dan hanya bertambah oleh perubahan bakteri laten. Keterjangkauannya mencapai $3.1 \cdot 10^{-3}$ CFU/ml.



Gambar 4.19: Dinamika Bakteri Debris

Pada Gambar 4.19 menunjukkan jumlah bakteri debris (bakteri puing-puing) setelah 5 jam. Bakteri debris merupakan bakteri yang dalam perjalanan infeksiya mengalami kematian dan menjadi puing-puing. Terlihat bahwa mengalami pertumbuhan dan kemudian mengalami penurunan karena mengalami degradasi. Keterjangkauannya mencapai 110.37 debris.

Halaman ini sengaja dikosongkan.

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan hasil analisis dan pembahasan yang telah dilakukan dalam tugas akhir ini dapat diambil kesimpulan sebagai berikut:

1. Dengan model dinamik bakteriofag yang digunakan, didapatkan hasil reachabilitas dengan masing-masing himpunan kondisi awal yang berbeda. Pada skenario 1 dengan bakteriofag berjumlah $1000 - 2000 PFU/ml$, didapatkan sistem menjangkau (*reachable*) kondisi yang tidak aman dan tidak menjangkau kondisi target. Pada skenario 2 dengan bakteriofag berjumlah $55.10^5 - 55,2.10^5 PFU/ml$, didapatkan sistem tidak menjangkau (*reachable*) kondisi yang tidak aman dan tidak menjangkau kondisi target. Pada skenario 3 dengan bakteriofag berjumlah $45.10^6 - 50.10^6 PFU/ml$, didapatkan sistem tidak menjangkau (*not reachable*) kondisi yang tidak aman dan menjangkau kondisi target.
2. Dengan sifat keamanan yang diberikan yaitu "Bakteriofag T4 tidak akan pernah lebih sedikit dari bakteri yang ada dalam tubuh selama waktu pengamatan", dapat disimpulkan bahwa skenario 1 melanggar (tidak memenuhi) sifat keamanan yang diberikan. Namun pada skenario 2 dan 3 memenuhi sifat keamanan yang diberikan.

5.2 Saran

Adapun saran yang dapat dilakukan untuk penelitian selanjutnya, yaitu:

1. Menggunakan jumlah bakteri resisten yang didapat pada tugas akhir ini untuk diterapkan pada model matematika terapi koktail fag.
2. Membandingkan model Taylor dengan representasi himpunan yang lain, yaitu representasi geometri.

DAFTAR PUSTAKA

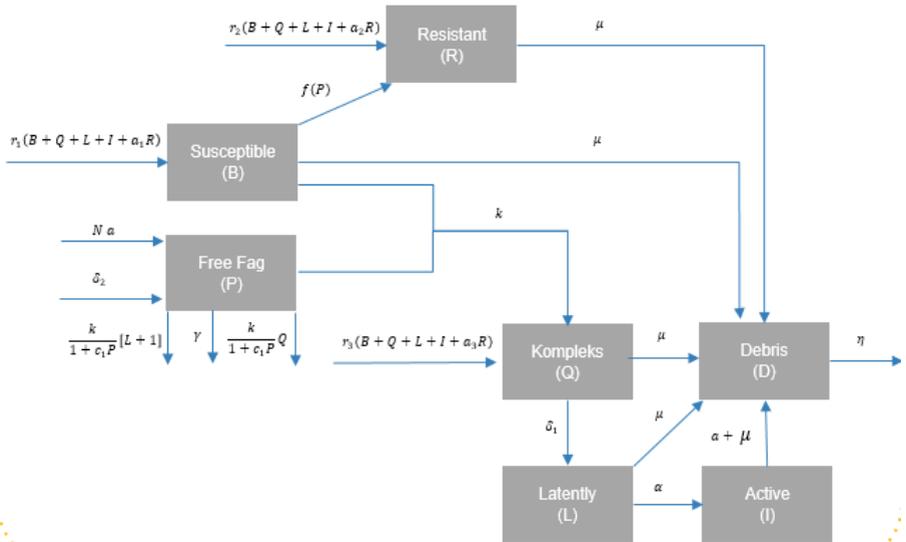
- [1] Vidurupola, S.W., 2017, "*Analysis of Deterministic and Stochastic Mathematical Models with Resistant Bacteria and Bacteria Debris for Bacteriophage Dynamics*", Applied Mathematics and Computation 316, pp. 215-228.
- [2] Vidurupola, S.W., 2014, "*Impact of Variability in Stochastic Models of Bacteria-Phage Dynamics Applicable to Phage Therapy*", Stochastic Analysis and Applications 32, pp. 427-449.
- [3] Hariyanto, Yunus, M., Yuni SL, G., 2016, "*Persistence Analysis on Pre-coalition Models of H1N1-p with H5N1 Virus in L^2 Space*", INTERNATIONAL JOURNAL OF COMPUTING SCIENCE AND APPLIED MATHEMATICS, VOL. 2, NO. 2, pp. 29-32.
- [4] Chen, Xin., Abraham, E., Sankaranarayanan, S., 2012, "*Taylor Model Flowpipe Construction for Non-linear Hybrid Systems*", In Proc. RTSS12, IEEE, pp. 183-192.
- [5] Seshia, S.A., 2015, "*Reachability Analysis and Model Checking*", Introduction to Embedded Systems.
- [6] Chen, Xin., Dutta, S., Sankaranarayanan, S., 2017, "*Formal Verification of a Multi-Basal Insulin Infusion Control Model*", ARCH@CPSWeek.
- [7] Chen, Xin., Althoff, M., Immler, F., Ghorbal, K., 2017, "*ARCH-COMP17 Category Report: Continuous System*

with Nonlinear Dynamics", EPiC Series in Computing, IEEE, Vol. 48, pp. 160-169

- [8] Sogokon, A., Johnson, T.T., Ghorbal, K., 2015, "*Benchmark for Non-linear Continuous System Safety Verification*", ARCH@CPSWeek, Vol. 43, pp. 42-51
- [9] Abraham, E., 2017, *Techniques and Tools for Hybrid Systems Reachability Analysis*, Theory of Hybrid System - Informatik 2.
- [10] Mitchell, I.M., 2007, "*Comparing Forward and Backward Reachability as Tools for Safety Analysis*", HSCC07 Proceedings of the 10th international conference on hybrid systems: computation and control. pp. 428-443.
- [11] Abraham, E., 2012, *Modeling and Analysis of Hybrid Systems: Reachability analysis for hybrid automata*, Theory of Hybrid System - Informatik 2.
- [12] Chen, Xin. , 2015, *Reachability Analysis of Non-Linear Hybrid Systems Using Taylor Models*, Doctoral dissertation, PhD thesis, RWTH Aachen University.
- [13] Chen, X., brahm, E. and Sankaranarayanan, S., "*Flow*: An analyzer for non-linear hybrid systems*", In Proc. of CAV13, Vol. 8044 of LNCS, pp. 258-263

LAMPIRAN A

Diagram Kompartemen Terapi Fag (Bakteri dan Bakteriofag)



Halaman ini sengaja dikosongkan.

LAMPIRAN B

File *.model* dan *.flow*

File *.model* Skenario 2

```
continuous reachability
{
  state var B, R, Q, L, I, D, P, t
  par
  {
    r11 = 0.3      r22 = 0.5      r33 = 0      al = 0.4      a2 = 0.5      a3 = 0.6 C
    = 1e8      cl = 1e-8      fl = 1e-6      k = 9e-7      alpha = 6.98      mdmu = 1e-7      a =
    6.98      N = 150      gamma = 1.8      chil = 6.98      chi2 = 0      n = 0.025      k3 = 0
  }

  setting
  {
    adaptive steps {min 0.0002 , max 0.01}      # time step-size
    time 5      # time horizon
    remainder estimation 1e-1      # remainder estimation
    identity precondition      # preconditioning method
    matlab octagon B,P      # 2D projection for plotting
    fixed orders 5
    cutoff 1e-6      # cutoff threshold is [-1e-6,1e-6]
    precision 100
    output bakteriTA8      # name of the output files
    print on      # print out the computation steps
  }

  nonpoly ode      # integration scheme
  {
    B' = r11*(1-(B+Q+L+I+a1*R)/(C))*B-(fl*P)/(1+P)*B-k*B*P-miu*B
    R' = r22*(1-(B+Q+L+I+a2*R)/(C))*R+(fl*P)/(1+P)*B-miu*R
    Q' = r33*(1-(B+Q+L+I+a3*R)/(C))*Q+k*B*P-chil*Q-miu*Q
    L' = chil*Q-alpha*L-miu*L
    I' = alpha*L-a*I-miu*I
    D' = miu*(B+Q+L+I)+a*I-n*D
    P' = N*a*I+chi2*Q-k*B*P-((k*P)/(1+(cl*P)))*(L+I)-((k*P)/(1+(cl*P)))*Q-k3*D-gamma*P
    t' = 1
  }

  init
  {
    B in {10,100}
    R in {0,0}
    Q in {0,0}
    L in {0,0}
    I in {0,0}
    D in {0,0}
    P in {5500000,5520000}
  }
}
```

File *.model* Skenario 3

```

continuous_reachability
{
state var B, R, Q, L, I, D, P, t
par
{
r11 = 0.3      r22 = 0.5      r33 = 0      a1 = 0.4      a2 = 0.5
a3 = 0.6      C = 1e8      cl = 1e-8      f1 = 1e-6      k = 9e-7
alpha = 6.98  miu = 1e-7      a = 6.98      N = 150      gamma = 1.8
chil = 6.98  chi2 = 0      n = 0.025      k3 = 0
}

setting
{
adaptive steps {min 0.0002 , max 0.01} # time step-size
time 5 # time horizon
remainder estimation 1e-1 # remainder estimation
identity precondition # preconditioning method
matlab octagon B,P # 2D projection for plotting
fixed orders 5
cutoff 1e-6 # cutoff threshold is [-1e-6,1e-6]
precision 100
output bakteriTAL0 # name of the output files
print on # print out the computation steps
}

nonpoly ode # integration scheme
{
B' = r11*(1-(B+Q+L+I+a1*R)/(C))*B-(f1*P)/(1+P)*B-k*B^P-miu*B
R' = r22*(1-(B+Q+L+I+a2*R)/(C))*R+(f1*P)/(1+P)*B-miu*R
Q' = r33*(1-(B+Q+L+I+a3*R)/(C))*Q+k*B^P-chil*Q-miu*Q
L' = chil*Q-alpha*L-miu*L
I' = alpha*L-a*I-miu*I
D' = miu*(B+Q+L+I)+a*I-n*D
P' = N*a*I+chi2*Q-k*B^P-((k*P)/(1+(cl*P)))*(L+I)-((k*P)/(1+(cl*P)))*Q-k3*D-gamma*P
t' = 1
}

init
{
B in [10,100]
R in [0,0]
Q in [0,0]
L in [0,0]
I in [0,0]
D in [0,0]
P in [45000000,50000000]
}
}

```

File *.flow* Skenario 1,2,3
(*Persamaan flowpipe model Taylor pertama*)

```

state var B,R,Q,L,I,D,P,t

matlab octagon B , P

order 4

cutoff 1.000000e-08

output bakteriTA3

continuous flowpipes
{
tm var local_var_1,local_var_2,local_var_3,local_var_4,local_var_
5,local_var_6,local_var_7,local_var_8

{
B = [5.500000000000000e1 , 5.500000000000000e1] +
[4.499999999999999e1 , 4.500000000000001e1] * local_var_1 +
[1.64256804613676e1 , 1.64256804613677e1] * local_t +
[-2.474999999998626e-2 , -2.474999999998625e-2] * local_t *
local_var_7 + [1.34391856797553e1 , 1.34391856797554e1] * local_t
* local_var_1 + [2.51957982752431 , 2.51957982752432] * local_t^2
+ [-2.024999999998876e-2 , -2.024999999998875e-2] * local_t *
local_var_1 * local_var_7 +
[-6.0749999999989875e-6 , -6.0749999999989874e-6] * local_t *
local_var_1^2 + [1.48840583972956e-2 , 1.48840583972957e-2] *
local_t^2 * local_var_7 + [2.06147258170089 , 2.06147258170090] *
local_t^2 * local_var_1 + [2.24030984506284e-1 ,
2.24030984506285e-1] * local_t^3 + [5.568749999993812e-6 ,
5.568749999993813e-6] * local_t^2 * local_var_7^2 +
[1.21783721607986e-2 , 1.21783721607987e-2] * local_t^2 *
local_var_1 * local_var_7 +
[-1.49124760011877e-6 , -1.49124760011876e-6] * local_t^2 *
local_var_1^2 + [-7.84690170862671e-3 , -7.84690170862670e-3] *
local_t^3 * local_var_7 + [1.83296087977844e-1 ,
1.83296087977845e-1] * local_t^3 * local_var_1 +
[2.73201088436932e-2 , 2.73201088436933e-2] * local_t^4 +
[-8.04931609745363e-9 , 8.13757387669118e-10]

R = [5.49633577612175e-5 , 5.49633577612176e-5] * local_t +
[4.49700199864507e-5 , 4.49700199864508e-5] * local_t *
local_var_1 + [2.19482005343844e-5 , 2.19482005343845e-5] *
local_t^2 + [-1.23667554962053e-8 , -1.23667554962052e-8] *
local_t^2 * local_var_7 + [6.71511609578022e-6 ,
6.71511609578023e-6] * local_t^2 * local_var_1 +
[4.49733030788851e-6 , 4.49733030788852e-6] * local_t^3 +
[-1.01182544968952e-8 , -1.01182544968951e-8] * local_t^2 *
local_var_1 * local_var_7 + [6.86699226558324e-7 ,
6.86699226558325e-7] * local_t^3 * local_var_1 +
[6.18135789841818e-7 , 6.18135789841819e-7] * local_t^4 +
[-8.07518358398369e-10 , 8.04398257457964e-9]

```

```

Q = [7.424999999995875e-2 , 7.424999999995876e-2] * local_t +
[2.474999999998625e-2 , 2.474999999998626e-2] * local_t *
local_var_7 + [6.074999999996625e-2 , 6.074999999996626e-2] *
local_t * local_var_1 +
[-3.14872003378041e-1 , -3.14872003378040e-1] * local_t^2 +
[2.024999999998875e-2 , 2.024999999998876e-2] * local_t *
local_var_1 * local_var_7 +
[-1.85965407092077e-2 , -1.85965407092076e-2] * local_t^2 *
local_var_7 + [4.56065567924306e-2 , -4.56065567924305e-2] *
local_t^2 * local_var_1 + [7.60527870382928e-1 ,
7.60527870382929e-1] * local_t^3 +
[-5.568749999993813e-6 , -5.568749999993812e-6] * local_t^2 *
local_var_7^2 + [-1.52158543474767e-2 , -1.52158543474766e-2] *
local_t^2 * local_var_1 * local_var_7 +
[-1.23018749998634e-6 , -1.23018749998633e-6] * local_t^2 *
local_var_1^2 + [9.33530127181173e-3 , 9.33530127181174e-3] *
local_t^3 * local_var_7 + [2.28496321365590e-2 ,
2.28496321365591e-2] * local_t^3 * local_var_1 +
[-1.33763903983912 , -1.33763903983911] * local_t^4 +
[-1.44870940915562e-7 , 1.44870945401893e-7]

L = [2.59132500000416e-1 , 2.59132500000417e-1] * local_t^2 +
[-1.33551715317453 , -1.33551715317452] * local_t^3 +
[3.65759859952188 , 3.65759859952189] * local_t^4 +
[-1.45496166820843e-7 , 1.45496151993864e-7]

I = [6.02914950005288e-1 , 6.02914950005289e-1] * local_t^3 +
[-3.38256403514589 , -3.38256403514588] * local_t^4 +
[-6.37508426453938e-10 , 6.37513028084236e-10]

D = [5.49999999997249e-6 , 5.49999999997250e-6] * local_t +
[4.49999999997749e-6 , 4.49999999997750e-6] * local_t *
local_var_1 + [7.56246523063205e-7 , 7.56246523063206e-7] *
local_t^2 + [6.71959283984407e-7 , 6.71959283984408e-7] *
local_t^2 * local_var_1 + [7.58259564454930e-8 ,
7.58259564454931e-8] * local_t^3 + [6.87157026042696e-8 ,
6.87157026042697e-8] * local_t^3 * local_var_1 +
[1.05208659406568 , 1.05208659406569] * local_t^4 +
[-2.80440690858365e-12 , 2.80217216645989e-12]

P = [1.50000000000000e3 , 1.50000000000000e3] +
[4.99999999999999e2 , 5.00000000000000e2] * local_var_7 +
[-2.70007425007500e3 , -2.70007425007499e3] * local_t +
[-9.00024750025000e2 , -9.00024750024999e2] * local_t *
local_var_7 + [-6.07499999999662e-2 , -6.07499999999662e-2] *
local_t * local_var_1 + [2.43012251452038e3 , 2.43012251452039e3]
* local_t^2 + [-2.02499999999887e-2 , -2.02499999999887e-2] *
local_t * local_var_1 * local_var_7 + [8.10040854879960e2 ,
8.10040854879961e2] * local_t^2 * local_var_7 +
[1.00281556793645e-1 , 1.00281556793646e-1] * local_t^2 *
local_var_1 + [-1.45810134919334e3 , -1.45810134919333e3] *

```

```

local_var_1 + [-1.45810134919324e3 , -1.45810134919322e3] *
local_t^3 + [5.568749999993812e-6 , 5.568749999993813e-6] *
local_t^2 * local_var_7^2 + [3.34408543478816e-2 ,
3.34408543478817e-2] * local_t^2 * local_var_1 * local_var_7 +
[1.230187499998633e-6 , 1.230187499998634e-6] * local_t^2 *
local_var_1^2 + [-4.86033819835226e2 , -4.86033819835225e2] *
local_t^3 * local_var_7 +
[-8.30185648612521e-2 , -8.30185648612520e-2] * local_t^3 *
local_var_1 + [8.13969029325007e2 , 8.13969029325008e2] *
local_t^4 + [-6.78074623732096e-10 , 6.75436983446084e-10]

t = [1.00000000000000 , 1.00000000000000] * local_t + [-0 , 0]

local_t in [0 , 6.250000000000001e-4]
local_var_1 in [-1.000000000000000 , 1.000000000000000]
local_var_2 in [-1.000000000000000 , 1.000000000000000]
local_var_3 in [-1.000000000000000 , 1.000000000000000]
local_var_4 in [-1.000000000000000 , 1.000000000000000]
local_var_5 in [-1.000000000000000 , 1.000000000000000]
local_var_6 in [-1.000000000000000 , 1.000000000000000]
local_var_7 in [-1.000000000000000 , 1.000000000000000]
local_var_8 in [-1.000000000000000 , 1.000000000000000]
}

```

```

state var B,R,Q,L,I,D,P,t

matlab octagon B , P

order 5

cutoff 1.000000e-06

output bakteriTA8

continuous flowpipes
{
tm var local_var_1,local_var_2,local_var_3,local_var_4,local_var_|
5,local_var_6,local_var_7,local_var_8

{
B = [5.500000000000000e1 , 5.500000000000000e1] +
[4.499999999999999e1 , 4.500000000000001e1] * local_var_1 +
[-2.56245005498760e2 , -2.56245005498759e2] * local_t +
[-4.94999999997251e-1 , -4.94999999997250e-1] * local_t *
local_var_7 + [-2.09655004498986e2 , -2.09655004498985e2] *
local_t * local_var_1 + [8.42400003562926e2 , 8.42400003562927e2]
* local_t^2 + [-4.04999999997751e-1 , -4.04999999997750e-1] *
local_t * local_var_1 * local_var_7 + [2.75171730073592 ,
2.75171730073593] * local_t^2 * local_var_7 +
[6.89241889637735e2 , 6.89241889637736e2] * local_t^2 *
local_var_1 + [-2.21798989817355e3 , -2.21798989817354e3] *
local_t^3 + [2.22749999997525e-3 , 2.22749999997526e-3] *
local_t^2 * local_var_7^2 + [2.25141508798848 , 2.25141508798849]
* local_t^2 * local_var_1 * local_var_7 + [4.51888874994979e-3 ,
4.51888874994980e-3] * local_t^2 * local_var_1^2 +
[-9.92457744301346 , -9.92457744301345] * local_t^3 * local_var_7
+ [-1.81475994489483e3 , -1.81475994489482e3] * local_t^3 *
local_var_1 + [5.04426541448175e3 , 5.04426541448176e3] *
local_t^4 + [1.82249999997975e-3 , 1.82249999997976e-3] *
local_t^2 * local_var_1 * local_var_7^2 + [8.201249999990887e-6 ,
8.201249999990888e-6] * local_t^2 * local_var_1^2 * local_var_7 +
[-1.43875697376513e-2 , -1.43875697376512e-2] * local_t^3 *
local_var_7^2 + [-8.12024939041313 , -8.12024939041312] *
local_t^3 * local_var_1 * local_var_7 +
[-3.34942277021543e-2 , -3.34942277021542e-2] * local_t^3 *
local_var_1^2 + [2.81514275082642e1 , 2.81514275082643e1] *
local_t^4 * local_var_7 + [4.12731183576501e3 ,
4.12731183576502e3] * local_t^4 * local_var_1 +
[-1.02902248102018e4 , -1.02902248102017e4] * local_t^5 +
[-2.71086084768772e-6 , 2.62236396744556e-6]

```

```

R = [0,0] + [0 , 6.29007176616143e-8]

Q = [2.72744999998484e2 , 2.72744999998485e2] * local_t +
[4.94999999997250e-1 , 4.94999999997251e-1] * local_t *
local_var_7 + [2.23154999998760e2 , 2.23154999998761e2] * local_t
* local_var_1 + [-1.83271680521364e3 , -1.83271680521363e3] *
local_t^2 + [4.04999999997750e-1 , 4.04999999997751e-1] * local_t
* local_var_1 * local_var_7 +
[-2.82596727598428 , -2.82596727598427] * local_t^2 * local_var_7
+ [-7.20690129829309e2 , -7.20690129829308e2] * local_t^2 *
local_var_1 + [6.56635103169992e3 , 6.56635103169993e3] *
local_t^3 + [-2.22749999997526e-3 , -2.22749999997525e-3] *
local_t^2 * local_var_7^2 +
[-2.31216506773714 , -2.31216506773713] * local_t^2 * local_var_1
* local_var_7 + [-4.51888874994980e-3 , -4.51888874994979e-3] *
local_t^2 * local_var_1^2 + [1.01997490813585e1 ,
1.01997490813586e1] * local_t^3 * local_var_7 +
[1.88368411088271e3 , 1.88368411088272e3] * local_t^3 *
local_var_1 + [-1.66688973159495e4 , -1.66688973159494e4] *
local_t^4 + [-1.82249999997976e-3 , -1.82249999997975e-3] *
local_t^2 * local_var_1 * local_var_7^2 +
[-8.20124999990888e-6 , -8.20124999990887e-6] * local_t^2 *
local_var_1^2 * local_var_7 + [1.46103196633950e-2 ,
1.46103196633951e-2] * local_t^3 * local_var_7^2 +
[8.34539082416105 , 8.34539082416106] * local_t^3 * local_var_1 *
local_var_7 + [3.39461164265120e-2 , 3.39461164265121e-2] *
local_t^3 * local_var_1^2 +
[-2.88957805683634e1 , -2.88957805683633e1] * local_t^4 *
local_var_7 + [-4.26341878626087e3 , -4.26341878626086e3] *
local_t^4 * local_var_1 + [3.38626617538138e4 ,
3.38626617538139e4] * local_t^5 + [-3.08090228010310e-4 ,
3.08090229168739e-4]

L = [9.51880050001530e2 , 9.51880050001531e2] * local_t^2 +
[-6.47882871490969e3 , -6.47882871490968e3] * local_t^3 +
[2.27638388199675e4 , 2.27638388199676e4] * local_t^4 +
[-5.50481001014113e4 , -5.50481001014112e4] * local_t^5 +
[-3.08552904167080e-4 , 3.08552903051032e-4]

I = [2.21470758301942e3 , 2.21470758301943e3] * local_t^3 +
[-1.51702208953627e4 , -1.51702208953626e4] * local_t^4 +
[5.29559476663847e4 , 5.29559476663848e4] * local_t^5 +
[-1.35196003810860e-6 , 1.35195998852211e-6]

```

```

D = [5.49999999997249e-6, 5.49999999997250e-6] * local_t +
[4.49999999997749e-6, 4.49999999997750e-6] * local_t *
local_var_1 + [-1.04827502248969e-5, -1.04827502248968e-5] *
local_t^2 * local_var_1 +
[-1.28122505496603e-6, -1.28122505496602e-6] * local_t^3 +
[2.29745455516013e-5, 2.29745455516014e-5] * local_t^3 *
local_var_1 + [3.86466473450258e3, 3.86466473450259e3] *
local_t^4 + [-4.53679751896867e-5, -4.53679751896866e-5] *
local_t^4 * local_var_1 +
[-2.11969517743712e4, -2.11969517743711e4] * local_t^5 +
[-5.90253405524329e-9, 5.90282444688096e-9]

P = [5.51000000000000e6, 5.51000000000000e6] +
[9.99999999999999e3, 1.00000000000001e4] * local_var_7 +
[-9.91827274527550e6, -9.91827274527549e6] * local_t +
[-1.80004950005000e4, -1.80004950004999e4] * local_t *
local_var_7 + [-2.23154999999876e2, -2.23154999999876e2] *
local_t * local_var_1 + [8.92668535310996e6, 8.92668535310997e6] *
local_t^2 + [-4.04999999999775e-1, -4.04999999999775e-1] *
local_t * local_var_1 * local_var_7 + [1.62021689593737e4,
1.62021689593738e4] * local_t^2 * local_var_7 +
[9.21529629833772e2, 9.21529629833773e2] * local_t^2 *
local_var_1 + [-5.35620445708672e6, -5.35620445708671e6] *
local_t^3 + [2.22749999997525e-3, 2.22749999997526e-3] *
local_t^2 * local_var_7^2 + [2.67666506774523, 2.67666506774524] *
local_t^2 * local_var_1 * local_var_7 + [4.51888874994979e-3,
4.51888874994980e-3] * local_t^2 * local_var_1^2 +
[-9.72794254405324e3, -9.72794254405323e3] * local_t^3 *
local_var_7 + [-2.43658548677490e3, -2.43658548677489e3] *
local_t^3 * local_var_1 + [2.99029076243777e6,
2.99029076243778e6] * local_t^4 + [1.82249999999797e-3,
1.82249999999797e-3] * local_t^2 * local_var_1 * local_var_7^2 +
[8.20124999999088e-6, 8.20124999999088e-6] * local_t^2 *
local_var_1^2 * local_var_7 +
[-1.59468196634174e-2, -1.59468196634173e-2] * local_t^3 *
local_var_7^2 + [-9.951326009711515, -9.951326009711514] *
local_t^3 * local_var_1 * local_var_7 +
[-3.66574496765573e-2, -3.66574496765572e-2] * local_t^3 *
local_var_1^2 + [4.39831897302898e3, 4.39831897302899e3] *
local_t^4 * local_var_7 + [5.35979404031975e3,
5.35979404031976e3] * local_t^4 * local_var_1 +
[-4.25021428393427e6, -4.25021428393426e6] * local_t^5 +
[-2.12199074749413e-4, 2.12197790617629e-4]

t = [1.00000000000000, 1.00000000000000] * local_t + [-0, 0]

```

```

state var E,R,Q,L,I,D,P,t|

matlab octagon B , P

order 5

cutoff 1.000000e-06

output bakteriTA10

continuous flowpipes
{
tm var local_var_1,local_var_2,local_var_3,local_var_4,local_var_
5,local_var_6,local_var_7,local_var_8

{
B = [5.500000000000000e1 , 5.500000000000000e1] +
[4.499999999999999e1 , 4.500000000000000e1] * local_var_1 +
[-2.32475000548722e3 , -2.32475000548722e3] * local_t +
[-1.237499999999313e2 , -1.237499999999313e2] * local_t *
local_var_7 + [-1.91025000448954e2 , -1.91025000448953e2] *
local_t * local_var_1 + [5.16712521764166e4 , 5.16712521764167e4]
* local_t^2 + [-1.012499999999438e2 , -1.012499999999437e2] *
local_t * local_var_1 * local_var_7 + [5.36456557513201e3 ,
5.36456557513202e3] * local_t^2 * local_var_7 +
[4.22765266662443e4 , 4.22765266662444e4] * local_t^2 *
local_var_1 + [-7.92306210574274e5 , -7.92306210574273e5] *
local_t^3 + [1.39218749998453e2 , 1.39218749998454e2] * local_t^2
* local_var_7^2 + [4.38919252195460e3 , 4.38919252195461e3] *
local_t^2 * local_var_1 * local_var_7 + [3.895593749995671e-2 ,
3.895593749995672e-2] * local_t^2 * local_var_1^2 +
[-1.21055211772266e5 , -1.21055211772265e5] * local_t^3 *
local_var_7 + [-6.48253287946546e5 , -6.48253287946545e5] *
local_t^3 * local_var_1 + [9.44341135521632e6 ,
9.44341135521633e6] * local_t^4 + [1.13906249998734e2 ,
1.13906249998735e2] * local_t^2 * local_var_1 * local_var_7^2 +
[2.05031249997721e-3 , 2.05031249997722e-3] * local_t^2 *
local_var_1^2 * local_var_7 +
[-6.16043922978142e3 , -6.16043922978141e3] * local_t^3 *
local_var_7^2 + [-9.90454609499252e4 , -9.90454609499251e4] *
local_t^3 * local_var_1 * local_var_7 +
[-2.25165512097724 , -2.25165512097723] * local_t^3 * local_var_1
^2 + [1.89019723440723e6 , 1.89019723440726e6] * local_t^4 *
local_var_7 + [7.72651163186951e6 , 7.72651163186952e6] *
local_t^4 * local_var_1 +
[-9.31041847786247e7 , -9.31041847786246e7] * local_t^5 +
[-5.12326968863336e-6 , 5.11877423351963e-6]

R = [0,0] + [0 , 7.05048578545601e-8]

```

```

Q = [2.35124999998693e3, 2.35124999998694e3] * local_t +
[1.23749999999312e2, 1.23749999999313e2] * local_t * local_var_7
+ [1.92374999998931e3, 1.92374999998932e3] * local_t^4
local_var_1 + [-6.02273271780722e4, -6.02273271780722e4] *
local_t^2 + [1.01249999999437e2, 1.01249999999438e2] * local_t *
local_var_1 * local_var_7 +
[-5.38312806894411e3, -5.38312806894410e3] * local_t^2 *
local_var_7 + [-4.25630640714005e4, -4.25630640714004e4] *
local_t^2 * local_var_1 + [9.37602250664944e5,
9.37602250664945e5] * local_t^3 +
[-1.39218749998454e2, -1.39218749998453e2] * local_t^2 *
local_var_7^2 + [-4.40438001689177e3, -4.40438001689176e3] *
local_t^2 * local_var_1 * local_var_7 +
[-3.89559374995672e-2, -3.89559374995671e-2] * local_t^2 *
local_var_1^2 + [1.21591668150950e5, 1.21591668150951e5] *
local_t^3 * local_var_7 + [6.52480939219641e5,
6.52480939219642e5] * local_t^3 * local_var_1 +
[-1.11389502528893e7, -1.11389502528892e7] * local_t^4 +
[-1.13906249998735e2, -1.13906249998734e2] * local_t^2 *
local_var_1 * local_var_7^2 +
[-2.05031249997722e-3, -2.05031249997721e-3] * local_t^2 *
local_var_1^2 * local_var_7 + [6.17436110014040e3,
6.17436110014041e3] * local_t^3 * local_var_7^2 +
[9.94843800558068e4, 9.94843800558069e4] * local_t^3 *
local_var_1 * local_var_7 + [2.25555071342859, 2.25555071342860]
* local_t^3 * local_var_1^2 +
[-1.89927637229615e6, -1.89927637229614e6] * local_t^4 *
local_var_7 + [-7.77513061293450e6, -7.77513061293449e6] *
local_t^4 * local_var_1 + [1.09220764068140e8,
1.09220764068141e8] * local_t^5 + [-2.97770574510384e-3,
2.97768927908652e-3]

L = [8.20586250001319e3, 8.20586250001320e3] * local_t^2 +
[-1.59221221592348e5, -1.59221221592347e5] * local_t^3 +
[1.91395696308321e6, 1.91395696308322e6] * local_t^4 +
[-1.82218585119073e7, -1.82218585119072e7] * local_t^5 +
[-2.98569423211236e-3, 2.98562200379407e-3]

I = [1.90923067501674e4, 1.90923067501675e4] * local_t^3 +
[-3.11157107437226e5, -3.11157107437225e5] * local_t^4 +
[3.10625924869193e6, 3.10625924869194e6] * local_t^5 +
[-1.30821714800991e-5, 1.30818489178790e-5]

D = [5.4999999997249e-6, 5.4999999997250e-6] * local_t +
[4.4999999997749e-6, 4.4999999997750e-6] * local_t *
local_var_1 + [-6.18749999993469e-6, -6.18749999993468e-6] *
local_t^2 * local_var_7 +
[-9.55125002239994e-5, -9.55125002239993e-5] * local_t^2 *

```

```

local_var_1 + [-1.16737500546832e-5, -1.16737500546831e-5] *
local_t^3 + [-5.06249999994657e-6, -5.06249999994656e-6] *
local_t^2 * local_var_1 * local_var_7 + [1.78818852503506e-4,
1.78818852503507e-4] * local_t^3 * local_var_7 +
[1.40921596844068e-3, 1.40921596844069e-3] * local_t^3 *
local_var_1 + [3.33160754084590e4, 3.33160754084591e4] *
local_t^4 + [4.64062499992523e-6, 4.64062499992524e-6] *
local_t^3 * local_var_7^2 + [1.46306417397755e-4,
1.46306417397756e-4] * local_t^3 * local_var_1 * local_var_7 +
[-3.02637920316455e-3, -3.02637920316454e-3] * local_t^4 *
local_var_7 + [-1.62062633980292e-2, -1.62062633980291e-2] *
local_t^4 * local_var_1 +
[-4.34541904217306e5, -4.34541904217305e5] * local_t^5 +
[-5.71150461882823e-8, 5.71151994011349e-8]

P = [4.75000000000000e7, 4.75000000000000e7] +
[2.49999999999999e6, 2.50000000000001e6] * local_var_7 +
[-8.55023512523750e7, -8.55023512523749e7] * local_t +
[-4.50012375012500e6, -4.50012375012499e6] * local_t *
local_var_7 + [-1.92374999998932e3, -1.92374999998931e3] *
local_t * local_var_1 + [7.69700643946839e7, 7.69700643946840e7]
* local_t^2 + [-1.01249999999438e2, -1.01249999999437e2] *
local_t * local_var_1 * local_var_7 + [4.05427868887285e6,
4.05427868887286e6] * local_t^2 * local_var_7 +
[4.42944390714389e4, 4.42944390714390e4] * local_t^2 *
local_var_1 + [-4.64492103891011e7, -4.64492103891010e7] *
local_t^3 + [1.59825774085434e2, 1.59825774085435e2] * local_t^2
* local_var_7^2 + [4.49550501689379e3, 4.49550501689380e3] *
local_t^2 * local_var_1 * local_var_7 + [3.89559374995671e-2,
3.89559374995672e-2] * local_t^2 * local_var_1^2 +
[-2.53570628740161e6, -2.53570628740160e6] * local_t^3 *
local_var_7 + [-6.79056978950446e5, -6.79056978950445e5] *
local_t^3 * local_var_1 + [2.91574221542346e7,
2.91574221542347e7] * local_t^4 +
[-3.49271594706440e-1, -3.49271594706439e-1] * local_t^2 *
local_var_7^3 + [1.13906249998734e2, 1.13906249998735e2] *
local_t^2 * local_var_1 * local_var_7^2 + [2.05031249997721e-3,
2.05031249997722e-3] * local_t^2 * local_var_1^2 * local_var_7 +
[-6.59977789508102e3, -6.59977789508101e3] * local_t^3 *
local_var_7^2 + [-1.02181671381836e5, -1.02181671381835e5] *
local_t^3 * local_var_1 * local_var_7 +
[-2.27892427592899, -2.27892427592898] * local_t^3 * local_var_1
^2 + [2.82584743536949e6, 2.82584743536950e6] * local_t^4 *
local_var_7 + [8.08068567528916e6, 8.08068567528917e6] *
local_t^4 * local_var_1 +
[-1.08289845488252e8, -1.08289845488251e8] * local_t^5 +
[-1.28704288349095e-2, 1.28642838611627e-2]

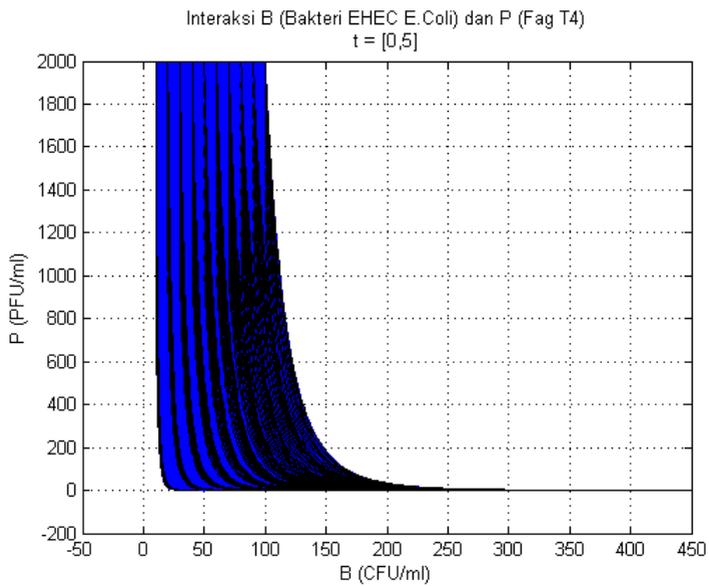
t = [1.00000000000000, 1.00000000000000] * local_t + [-0, 0]

```

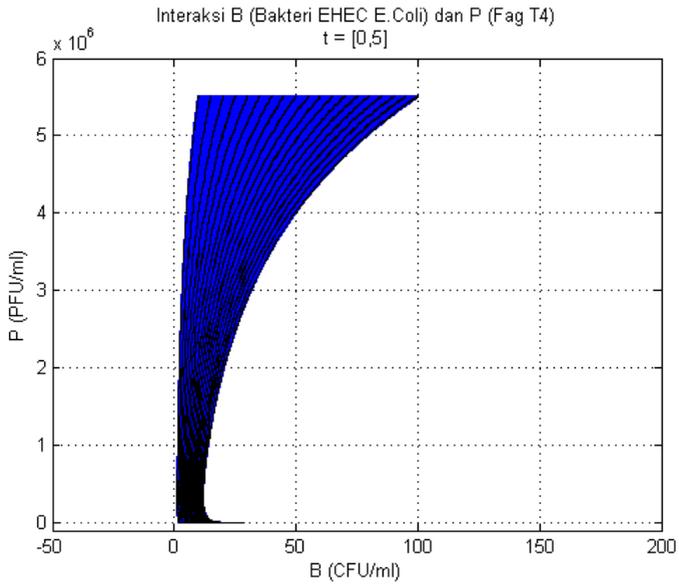
Halaman ini sengaja dikosongkan.

LAMPIRAN C

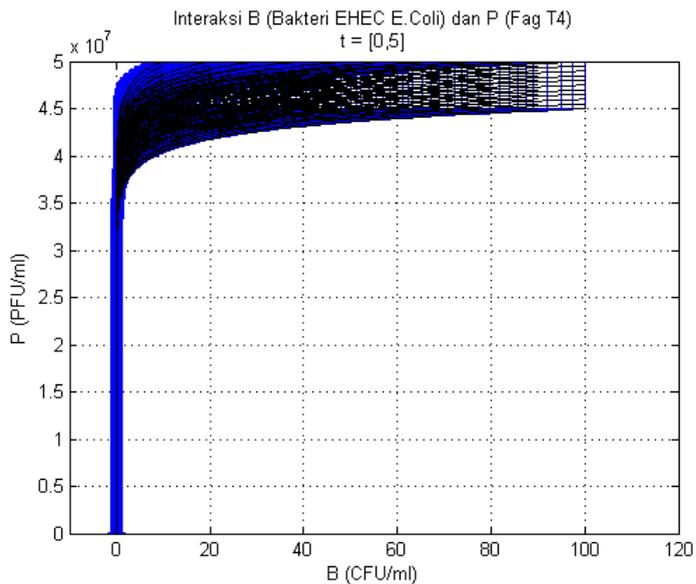
Flowpipe dan Simulasi Numerik Skenario 1



Flowpipe dan Simulasi Numerik Skenario 2



Flowpipe dan Simulasi Numerik Skenario 3



Halaman ini sengaja dikosongkan.

LAMPIRAN D

Biodata Penulis



Penulis bernama Annisa Mega Putri, biasa dipanggil Annisa. Penulis dilahirkan di Surabaya, 29 Juli 1996. Penulis telah menempuh pendidikan formal SD Muhammadiyah 22 Surabaya (2002-2008), SMP Negeri 16 Surabaya (2008-2011), dan SMA Negeri 1 Surabaya (2011-2014). Kemudian penulis melanjutkan studi ke jenjang S1 di Departemen Matematika ITS Surabaya pada tahun 2014. Di Departemen

Matematika, penulis mengambil Bidang Minat Matematika Terapan. Selama kuliah, penulis memiliki pengalaman berkomunitas di luar lingkungan ITS yaitu Young On Top (YOT) Surabaya (2015-2017). Selain itu penulis juga aktif di kegiatan kepanitian yang ada di ITS seperti GERIGI 2015, OMITS 2015, OMITS 2016, YES SUMMIT 2016 dan berbagai kegiatan yang lain. Penulis juga aktif dalam pelatihan kemahasiswaan dan kegiatan seminar yang diadakan oleh ITS.

Informasi lebih lanjut mengenai Tugas Akhir ini dapat ditujukan ke penulis melalui email: *meganisajuly@gmail.com*