



**PENERAPAN METODE RUNGE-KUTTA ORDE SEPULUH PADA
MODEL PENYEBARAN VIRUS AVIAN INFLUENZA**

TESIS

Oleh:

SUHARYONO

NIM: 091820101024

**PROGRAM STUDI MAGISTER MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS JEMBER**

2015



**PENERAPAN METODE RUNGE-KUTTA ORDE SEPULUH PADA
MODEL PENYEBARAN VIRUS AVIAN INFLUENZA**

TESIS

diajukan guna melengkapi tugas akhir dan memenuhi salah satu syarat untuk menyelesaikan Program Pasca Sarjana (S2) dan mencapai gelar Magister Sains

Oleh:

SUHARYONO

NIM: 091820101024

**PROGRAM STUDI MAGISTER MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS JEMBER**

2015

PERSEMBAHAN

Dengan menyebut nama Allah yang maha pengasih lagi maha penyayang, serta sholawat atas Nabi Muhammad S.A.W, kupersembahkan suatu kebahagiaan penggalan bait dalam perjalanan hidupku teriring rasa terima kasih kepada:

1. Ayahanda Suja'i dan Ibunda Lisna serta Bapak Suharto dan Mama Sri Asih yang senantiasa memberikan kasih sayang, dukungan, motivasi dan do'a yang tiada henti, yang membuatku dewasa dalam menjalani hidup;
2. Istriku tercinta Vita Mila Sari S, S.Pd yang memberiku semangat dan doa didalam mewarnai kehidupan ini;
3. Anak-anakku Ahmad Mu'tasim Billah dan Muhammad Ibnu Haikal Mansyuri dan Bintang Fitriyah Bilkis yang selalu memberiku keceriaan;
4. Adik-adikku H. Nurhalifah, Ernawati, Sri Wahyuni, S.S yang selalu memberiku motivasi untuk menyelesaikan pendidikanku. Tak lupa untuk seluruh keluarga besarku yang selalu mendoakanku;
5. Bapak dan Ibu Dosen Program Studi Magister Matematika FMIPA yang telah dengan sabar dan penuh semangat memberikan ilmunya kepadaku;
6. Teman-teman Program Studi Magister Matematika FMIPA Angkatan 2010 atas kekompakannya dan kenangan yang mengesankan;
7. Almamater Fakultas MIPA Program Studi Magister Matematika Universitas Jember.

MOTTO

"Ceroboh dan tidak bisa menahan emosi adalah sikap yang bisa berakibat fatal"

"hindarilah hukum dan penyakit"

"Jadilah kutu buku dan janganlah menjadi kutu orang"

PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Suharyono

NIM : 091820101024

Menyatakan dengan sesungguhnya bahwa tesis yang berjudul: *Penerapan Metode Runge-Kutta Orde Sepuluh pada Model Penyebaran Virus Avian Influenza* adalah benar-benar hasil karya sendiri, kecuali jika dalam pengutipan substansi disebutkan sumbernya, dan belum diajukan pada instansi manapun, serta bukan karya jiplakan. Saya bertanggung jawab atas keabsahan dan kebenaran isinya sesuai dengan sikap ilmiah yang harus dijunjung tinggi.

Demikian pernyataan ini saya buat dengan sebenarnya, tanpa adanya tekanan dan paksaan dari pihak manapun serta bersedia mendapat sanksi akademik jika ternyata di kemudian hari pernyataan ini tidak benar.

Jember, Juni 2015

Yang menyatakan,

SUHARYONO

NIM. 091820101024

HALAMAN PENGANTAR

PENERAPAN METODE RUNGE-KUTTA ORDE SEPULUH PADA
MODEL PENYEBARAN VIRUS AVIAN INFLUENZA

TESIS

Diajukan untuk dipertahankan di depan Tim Penguji sebagai salah satu persyaratan untuk menyelesaikan Program Studi magister Matematika pada Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember

Oleh:

Nama : Suharyono
NIM : 091820101024
Tempat dan Tanggal Lahir : Situbondo, 03 Juli 1977
Jurusan / Program : MIPA / Magister Matematika

Disetujui oleh:

Pembimbing I,

Pembimbing II,

Prof. Drs. Dafik, M.Sc, Ph.D.
NIP. 19680802 199303 1 004

Kosala Dwidja Purnomo, S.Si, M.Si
NIP. 19690828 199802 1 001

HALAMAN PENGESAHAN

Tesis berjudul Penerapan Metode Runge-Kutta Orde Sepuluh pada Model Penyebaran Virus Avian Influenza telah diuji dan disahkan oleh Fakultas Matematika Dan Ilmu Pengetahuan Alam pada:

Hari : Senin

Tanggal : 08 Juni 2015

Tempat : Gedung Pasca FMIPA Matematika UNEJ

Tim Penguji :

Ketua,

Sekretaris,

Prof. Drs. Dafik, M.Sc, Ph.D

NIP. 19680802 199303 1 004

Kosala Dwidja Purnomo, S.Si, M.Si

NIP. 19690828 199802 1 001

Anggota:

1. Prof. Drs. Slamir, M.Comp.Sc, Ph.D (.....)

NIP.19670420 199201 1 001

2. Drs.Rusli Hidayat, M.Sc (.....)

NIP. 19661012 199303 1 001

Mengetahui,

Dekan Fakultas Matematika Dan Ilmu Pengetahuan Alam

Universitas Jember

Prof. Drs. Kusno, DEA, Ph.D.

NIP. 196101081986021001

RINGKASAN

PENERAPAN METODE RUNGE-KUTTA ORDE SEPULUH PADA MODEL PENYEBARAN VIRUS AVIAN INFLUENZA; Suharyono, 091-820101024; 2015: 140 halaman; Program Studi Magister Matematika Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Jember.

Avian influenza atau flu burung adalah penyakit menular yang disebabkan virus Avian Influenza A dengan subtype H1 sampai H16 dan N1 sampai N9. Virus ini menyerang berbagai jenis unggas, meliputi ayam, kalkun, unggas air, burung peliharaan dan burung liar. Terakhir virus ini dapat menginfeksi babi, harimau, kucing dan macan tutul. Pada umumnya virus avian influenza A tidak menyerang manusia, tetapi subtype tertentu seperti H5N1, H7N7 yang bersifat sangat patogen dapat menyerang manusia dan mengakibatkan kematian. Terdapat beberapa cara penularan virus Avian Influenza A dari spesies unggas ke manusia antara lain melalui kontak langsung maupun tidak langsung dengan unggas yang sakit termasuk air liur dan tinja, udara dan alat alat peternakan yang terkontaminasi dengan virus Avian Influenza. Model matematika penyebaran virus *Avian Influenza* merupakan sistem persamaan diferensial biasa (PDB) orde satu non linier yang dikemukakan oleh Kermack Mckendric dan telah ditulis oleh Okosun pada tahun 2007. Penelitian ini menggunakan metode one step yaitu metode Runge-Kutta orde sepuluh untuk menganalisis sistem persamaan diferensial orde satu non linier model penyebaran virus *Avian Influenza*.

Penelitian ini bertujuan untuk mengetahui sifat metode Runge-Kutta orde sepuluh, mengetahui formulasi metode Runge-Kutta orde sepuluh, mengetahui apakah metode Runge-Kutta orde sepuluh merupakan metode yang konvergen, mengetahui efektivitas dan efisiensi metode Runge-Kutta orde sepuluh dalam menyelesaikan model penyebaran virus Avian Influenza.

Dalam penelitian ini, peneliti menggunakan metode dokumentasi dan metode eksperimen dalam proses pengumpulan data. Metode dokumentasi diartikan sebagai teknik pengumpulan data dengan menghimpun dan menganalisa

dokumen-dokumen, baik dokumen tertulis, gambar, hasil karya, maupun elektronik. Dalam penelitian ini peneliti menggunakan jurnal yang ditulis oleh Okosun yaitu tentang model penyebaran virus Avian Influenza dan berbagai sumber lainnya yang relevan dengan penelitian ini.

Adapun hasil penelitian disimpulkan sebagai berikut. Pertama Metode Runge-Kutta sepuluh tahap yang sekaligus berorde sepuluh mempunyai sifat seperti pada Lemma 4.1.1. Kedua, Hasil penurunan formula metode Runge-Kutta orde sepuluh adalah pada Corollary 4.1 dan Corollary 4.2. Ketiga, Metode Runge-Kutta orde sepuluh merupakan metode yang konvergen, dibuktikan pada teorema 4.1. Keempat, Berdasarkan hasil eksekusi *programming* metode Runge-Kutta orde sepuluh RK10A lebih efektif dibandingkan metode Runge-Kutta orde sepuluh RK10B. Akan tetapi metode Runge-Kutta orde sepuluh RK10B lebih efisien dibandingkan metode Runge-Kutta orde sepuluh RK10A dalam menyelesaikan model.

KATA PENGANTAR

Puji syukur ke hadirat Allah Swt atas segala rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan tesis yang berjudul Penerapan Metode Runge-Kutta Orde Sepuluh Pada Model Penyebaran Virus Avian Influenza. Tesis ini disusun sebagai salah satu syarat untuk menyelesaikan pendidikan strata dua (S2) pada Program Studi Pasca Sarjana Jurusan Matematika Fakultas Matematika Dan Ilmu Pengetahuan Alam Universitas Jember.

Pada kesempatan ini penulis mengucapkan terima kasih dan penghargaan yang sebesar-besarnya atas bantuan dan bimbingan dalam penyusunan tesis ini, terutama kepada yang terhormat:

1. Dekan Fakultas Matematika Dan Ilmu Pengetahuan Alam Universitas Jember;
2. Ketua Jurusan MIPA Fakultas Matematika Dan Ilmu Pengetahuan Alam Universitas Jember;
3. Ketua Program Studi Pasca Sarjana Jurusan Matematika Fakultas Matematika Dan Ilmu Pengetahuan Alam Universitas Jember;
4. Bapak Prof. Drs. Dafik, M.Sc., Ph.D., selaku Dosen Pembimbing utama sekaligus Bapak Kosala Dwidja Purnomo, S.Si, M.Si, selaku Dosen Pembimbing Anggota yang telah meluangkan waktu, pikiran, dan perhatian dalam penulisan tesis ini;
5. Segenap dosen dan Karyawan Fakultas Matematika Dan Ilmu Pengetahuan Alam Universitas Jember;
6. Semua pihak yang telah membantu terselesaikannya tesis ini.

Penulis juga menerima segala kritik dan saran dari semua pihak demi kesempurnaan tesis ini.

Jember,
Penulis

DAFTAR ISI

HALAMAN JUDUL	
Halaman Persembahan	ii
HALAMAN MOTTO	iii
Halaman Pernyataan	iv
Halaman Pengajuan	v
Halaman Pengesahan	vi
Ringkasan	vii
Kata Pengantar	ix
DAFTAR ISI	xii
DAFTAR GAMBAR	xiv
DAFTAR TABEL	xv
1 PENDAHULUAN	1
1.1 Latar Belakang Masalah	1
1.2 Rumusan Masalah	3
1.3 Tujuan Penelitian	3
1.4 Manfaat Penelitian	4
2 TINJAUAN PUSTAKA	5
2.1 Avian Influenza (Flu Burung)	6
2.1.1 Pengertian Avian Influenza (Flu Burung)	6
2.1.2 Pengaruh Pada Manusia	7
2.1.3 Model Penyebaran Avian Influenza	8
2.2 Nilai Awal	10
2.3 Persamaan Diferensial Biasa	11
2.3.1 Sistem Persamaan Diferensial Biasa (PDB)	12
2.3.2 Metode Penyelesaian Sistem PDB	13
2.4 Konsep Dasar Metode Numerik	15
2.5 Aturan Matematika yang Digunakan dalam Penelitian	19

2.6	Metode Runge-Kutta	20
2.6.1	Konsep Konvergensi Metode Runge-Kutta	23
2.6.2	Penurunan Formula Metode Runge-Kutta	24
2.6.3	Metode Runge-Kutta Orde Sepuluh	28
2.7	Efektivitas dan Efisiensi	32
2.8	Jumlah Iterasi	33
2.9	Algoritma dan Pemrograman	34
2.10	MATLAB Programming	36
3	METODE PENELITIAN	38
3.1	Rancangan Penelitian	38
3.2	Definisi Operasional	39
3.3	Tempat Penelitian	41
3.4	Metode Pengumpulan Data	41
3.5	Analisis Data	42
4	HASIL DAN PEMBAHASAN	43
4.1	Metode Runge-Kutta Orde Sepuluh	43
4.2	Konvergensi Metode Runge-Kutta Orde Sepuluh	61
4.3	Pemrograman MATLAB	68
4.3.1	Tahap Pemodelan	69
4.3.2	Tahap Formulasi Numerik	69
4.3.3	Pola Algoritma Metode Runge-Kutta Orde Sepuluh	70
4.3.4	Format Pemrograman Metode Runge-Kutta Orde Sepuluh	73
4.4	Efektivitas dan Efisiensi Metode Runge-Kutta Orde Sepuluh	74
4.4.1	Simulasi Pemodelan	74
4.4.2	Hasil Komputasi Metode Runge-Kutta Orde Sepuluh dengan MATLAB	76
4.4.3	Analisis Efektivitas Metode Runge-Kutta Orde Sepuluh	99
4.4.4	Analisis Efisiensi Metode Runge-Kutta Orde Sepuluh	104
5	KESIMPULAN DAN SARAN	107
5.1	Kesimpulan	107

5.2 Saran	109
DAFTAR PUSTAKA	110
LAMPIRAN-LAMPIRAN	113



DAFTAR GAMBAR

2.1	Diagram Aproksimasi	18
2.2	Tampilan Awal dan Lembar Kerja MATLAB	36
3.1	Prosedur Penelitian	40
4.1	Grafik eksekusi RK10A dengan iterasi 100 pada populasi manusia dan unggas terhadap waktu (t) dengan $h=0.01$	77
4.2	Grafik eksekusi RK10B dengan iterasi 100 pada populasi manusia dan unggas terhadap waktu (t) dengan $h=0.01$	78
4.3	Grafik eksekusi RK10A dengan iterasi 500 pada populasi manusia dan unggas terhadap waktu (t) dengan $h=0.01$	80
4.4	Grafik eksekusi RK10B dengan iterasi 500 pada populasi manusia dan unggas terhadap waktu (t) dengan $h=0.01$	81
4.5	Grafik eksekusi RK10A dengan iterasi 2500 pada populasi manusia dan unggas terhadap waktu (t) dengan $h=0.01$	82
4.6	Grafik eksekusi RK10B dengan iterasi 2500 pada populasi manusia dan unggas terhadap waktu (t) dengan $h=0.01$	83
4.7	Grafik eksekusi RK10A dengan iterasi 25000 pada populasi manusia dan unggas terhadap waktu (t) dengan $h=0.01$	85
4.8	Grafik eksekusi RK10B dengan iterasi 25000 pada populasi manusia dan unggas terhadap waktu (t) dengan $h=0.01$	86
4.9	Grafik eksekusi RK10A dengan iterasi 50000 pada populasi manusia dan unggas terhadap waktu (t) dengan $h=0.01$	87
4.10	Grafik eksekusi RK10B dengan iterasi 50000 pada populasi manusia dan unggas terhadap waktu (t) dengan $h=0.01$	88
4.11	Grafik eksekusi RK10A dan RK10B dengan tol 10^{-3} pada populasi manusia dan unggas terhadap waktu (t) dengan $h=0.01$	90
4.12	Grafik eksekusi RK10A dan RK10B dengan tol 10^{-4} pada populasi manusia dan unggas terhadap waktu (t) dengan $h=0.01$	92

4.13	Grafik konvergensi metode RK10A dan RK10B dengan iterasi 100 dan $h=0.01$ pada populasi manusia dan unggas	94
4.14	Grafik konvergensi metode RK10A dan RK10B dengan iterasi 500 dan $h=0.01$ pada populasi manusia dan unggas	95
4.15	Grafik konvergensi metode RK10A dan RK10B dengan iterasi 2.500 dan $h=0.01$ pada populasi manusia dan unggas	96
4.16	Grafik konvergensi metode RK10A dan RK10B dengan iterasi 25.000 dan $h=0.01$ pada populasi manusia dan unggas	97
4.17	Grafik konvergensi metode RK10A dan RK10B dengan iterasi 50.000 dan $h=0.01$ pada populasi manusia dan unggas	98
4.18	Grafik konvergensi metode RK10A dan RK10B dengan tol 10^{-3} dan $h=0.01$ pada populasi manusia dan unggas	100
4.19	Grafik konvergensi metode RK10A dan RK10B dengan tol 10^{-4} dan $h=0.01$ pada populasi manusia dan unggas	101

DAFTAR TABEL

2.1	Interpretasi Parameter	10
2.2	Koefisien-koefisien Metode Runge-Kutta	28
4.1	Matriks Koefisien Runge-Kutta Order Sepuluh (RK10A)	57
4.2	Matriks koefisien Runge-Kutta Orde Sepuluh (RK10B)	68
4.3	Interpretasi Parameter	70
4.4	Data Efektivitas Metode Runge-Kutta Orde Sepuluh	102
4.5	Data Efisiensi Runge-Kutta Orde Sepuluh	105

BAB 1. PENDAHULUAN

1.1 Latar Belakang Masalah

Banyak masalah di dalam ilmu pengetahuan dan teknologi menyangkut pengkajian suatu sistem selama periode waktu tertentu dapat dimodelkan dengan menggunakan suatu sistem persamaan diferensial, dengan waktu sebagai variabel bebas yang diterjemahkan dalam sistem persamaan diferensial biasa dengan kondisi-kondisi nilai batas yang diketahui. Bidang kajian persamaan diferensial tidak hanya merupakan salah satu bagian dalam matematika, namun persamaan diferensial juga merupakan alat yang penting di dalam memodelkan berbagai fenomena dan masalah dalam bidang ilmu-ilmu fisika, kimia, biologi, kesehatan, ekonomi dan teknik. Sebagai contoh, pemanfaatan matematika dalam bidang fisika dapat diterapkan pada mekanika dan persamaan gelombang. Dalam bidang biologi, model matematika dapat digunakan untuk membantu merepresentasikan model penyebaran sebuah penyakit. Dalam bidang ekonomi, matematika dapat ditemukan pada perhitungan rugi/ laba atau bunga dan masih banyak contoh aplikasi matematika dalam bidang kimia, sosial dan lain-lain. Dalam penelitian ini akan dibahas mengenai salah satu aplikasi matematika dalam bidang kesehatan, yakni dalam menganalisis model penyebaran virus Avian Influenza.

Penyelesaian persamaan diferensial ini dapat diselesaikan dengan menggunakan metode numerik. Adapun teori-teori yang mendasari metode numerik sudah demikian banyaknya. Salah satu metode numerik yang mendasari persamaan diferensial biasa dengan kondisi-kondisi nilai awal yang sudah diketahui yaitu metode Runge Kutta yang merupakan metode *one-step*. Metode Runge Kutta adalah salah satu metode yang dipakai dalam memecahkan sistem persamaan diferensial seperti model penyebaran virus Avian Influenza.

Baru-baru ini, model matematika sering digunakan untuk mempelajari dinamika penyebaran virus Avian Influenza di berbagai daerah. Sebagai contoh

model penyebaran virus Avian Influenza (Okosun, 2007) digunakan untuk menganalisis penyebaran virus Avian Influenza. Efektivitas metode Adams Bashforth-Moulton orde delapan terhadap metode Runge-Kutta orde enam Pada model penyebaran Virus Avian Influenza di teliti oleh (Bukaryo; 2012). Peneliti ingin mengembangkan penelitian tersebut dengan menggunakan orde yang lebih tinggi.

Beberapa penelitian tentang solusi persamaan diferensial secara numerik telah banyak dilakukan dalam berbagai bidang kesehatan. Diantaranya adalah Susanti (2010) yang telah menggunakan metode *one step* dalam menyelesaikan model penyebaran bakteri *Leptospira*. Susanti menggunakan metode *one step* yaitu metode Runge-Kutta orde lima terhadap model penyebaran bakteri *Leptospira*.

Penelitian lain yang dilakukan Yustica (2010) yang menggunakan metode satu langkah (*one-step*) dalam menyelesaikan model penyebaran virus flu burung, yakni dengan menggunakan metode Runge-Kutta orde lima. Yustica menggunakan metode Runge-Kutta orde lima dibandingkan dengan metode Runge-Kutta orde empat dalam penelitiannya untuk mengetahui efektivitas metode Runge-Kutta orde lima dalam menyelesaikan model penyebaran virus flu burung. Dari hasil penelitiannya, Yustica menyimpulkan bahwa metode Runge-Kutta orde lima lebih efektif jika dibandingkan dengan metode Runge-Kutta orde empat dalam menyelesaikan model penyebaran virus flu burung, dan ia juga mengajukan saran agar dilakukan penelitian lagi dengan orde lebih tinggi untuk melihat keefektivan metode tersebut.

Penelitian-penelitian sebelumnya yang berkaitan dengan metode Runge-Kutta yaitu: metode Runge-Kutta orde tiga (Asih, 2001); metode Runge-Kutta orde empat (Faisol, 2001); metode Runge-Kutta orde lima (Yustica, 2010); metode Runge-Kutta orde enam (Bukaryo, 2012); metode Runge-Kutta orde tujuh (Shodiq, 2012); metode Runge-Kutta orde delapan (Ardhilia, 2013); metode Runge-Kutta orde sembilan (Anggraeni, 2013) dilakukan dengan membandingkan metode Runge-Kutta dengan metode numerik berbeda dengan orde yang sama. Berdasarkan penelitian Yustica (2010) diperoleh kesimpulan bahwa metode

Runge-Kutta orde lima lebih efektif bila dibandingkan dengan metode Runge-Kutta orde empat. Di sisi lain, menurut Shodiq (2012) metode Runge-Kutta orde tujuh lebih efektif bila dibandingkan dengan metode Adams Bashforth-Moulton orde enam.

Dengan memperhatikan solusi-solusi numerik hasil penelitian yang telah dilakukan oleh peneliti lain dan juga model persamaan diferensial biasa dengan kondisi nilai awal yang diketahui dalam berbagai bidang ilmu maka peneliti ingin melakukan penelitian yang serupa dengan orde yang lebih tinggi yaitu metode Runge-Kutta order sepuluh dimana model yang digunakan peneliti merupakan model penyebaran virus Avian Influenza. Sehingga Peneliti melakukan penelitian yang berjudul " Penerapan Metode Runge-Kutta Orde Sepuluh pada Model Penyebaran Virus Avian Influenza ".

1.2 Rumusan Masalah

Berdasarkan uraian yang dikemukakan pada latar belakang, dirumuskan beberapa masalah sebagai berikut:

1. bagaimanakah sifat metode Runge-Kutta orde sepuluh?
2. bagaimanakah formulasi metode Runge-Kutta orde sepuluh?
3. apakah metode Runge-Kutta orde sepuluh merupakan metode yang konvergen?
4. bagaimanakah efektivitas dan efisiensi metode Runge-Kutta orde sepuluh dalam menyelesaikan model penyebaran virus Avian Influenza?

1.3 Tujuan Penelitian

Sebagaimana yang tertulis dalam rumusan masalah, maka tujuan yang ingin dicapai dalam penelitian ini adalah:

1. mengetahui sifat metode Runge-Kutta orde sepuluh;
2. mengetahui formulasi metode Runge-Kutta orde sepuluh;
3. mengetahui apakah metode Runge-Kutta orde sepuluh merupakan metode yang konvergen;
4. mengetahui efektivitas dan efisiensi metode Runge-Kutta orde sepuluh dalam menyelesaikan model penyebaran virus Avian Influenza.

1.4 Manfaat Penelitian

Adapun manfaat dari penelitian ini adalah:

1. dalam bidang ilmu eksak khususnya Matematika, dapat memberikan kontribusi berupa formulasi dan hasil programming metode Runge Kutta order sepuluh;
2. bagi masyarakat umum, dapat memberikan deskripsi tentang efektivitas dan efisiensi metode Runge-Kutta orde sepuluh dalam menyelesaikan model penyebaran virus Avian Influenza;
3. dalam bidang kesehatan, dapat dijadikan masukan dalam merencanakan strategi pencegahan (preventif) dan penyembuhan (kuratif) terhadap penyebaran virus Avian Influenza;
4. bagi peneliti lain, dapat digunakan sebagai referensi dan bahan pertimbangan apabila ingin melakukan penelitian sejenis.

BAB 2. TINJAUAN PUSTAKA

Wabah flu burung (*Avian Influenza : AI*) saat ini telah menjadi isu global. Penanganan yang serius perlu segera diambil agar wabah flu burung tidak bermutasi menjadi flu yang menular dari manusia ke manusia dan menjadi wabah pandemi influenza (*Pandemic Influenza : PI*). Kerugian yang terjadi seandainya virus flu burung menjadi flu yang menular dari manusia ke manusia akan sangat besar berupa kerugian ekonomi akibat banyaknya unggas yang harus dimusnahkan, kerugian berupa biaya sosial karena banyaknya orang yang sakit dan bahkan meninggal di Indonesia. Sebagai bagian dari komunitas internasional maka Indonesia juga berkewajiban untuk membuat suatu rencana strategis nasional dalam menangani flu burung. Rencana strategis nasional disusun secara terpadu, baik dari aspek penanganan kesehatan hewan/ ternak maupun penanganan kesehatan manusia. Untuk itu, disusun rencana strategis nasional pengendalian flu burung dan kesiapsiagaan menghadapi *Pandemi Influenza* (PI). Rencana strategis nasional pengendalian flu burung dan kesiapsiagaan menghadapi pandemi influenza ini merupakan panduan nasional penanganan flu burung di Indonesia bagi setiap *stakeholders* sebagaimana diidentifikasi dalam rencana strategis nasional ini. Selanjutnya, di tingkat regional dan global, rencana strategis ini merupakan bagian dari strategi regional dan global. Dengan demikian, rencana strategis Indonesia juga menjadi bagian dari penyelesaian masalah flu burung secara global (Suzetta, 2005).

Penyebaran flu burung pertama kali di Indonesia pada pertengahan tahun 2003, diawali dengan kematian sejumlah besar unggas di Kabupaten Pekalongan, Jawa Tengah dan Kabupaten Tangerang Banten. Penyebaran virus flu burung di Indonesia terus berlanjut. Bahkan penyakit ini menjadi wabah yang terdapat di sebagian besar wilayah provinsi di Indonesia. Daerah penyebarannya meluas dari 9 provinsi dengan 53 kabupaten/kota, tahun 2003 menjadi 26 provinsi dengan

172 kabupaten/kota, tahun 2006 dengan diagnosis lebih lanjut dipastikan bahwa wabah disebabkan oleh virus flu burung tipe A, sub tipe H5N1. Bahkan hingga Mei 2007 wilayah yang tertular flu burung pada unggas telah mencapai 31 propinsi di Indonesia (Berri dalam Suzetta, 2005).

Fenomena yang sering terjadi hampir disetiap daerah baik di Indonesia maupun di negara-negara lain tentang menyebarnya virus *Avian Influenza* (flu burung) menjadi wabah internasional yang meresahkan warga di berbagai negara. Bahkan Organisasi Kesehatan Dunia (WHO) menaruh perhatian serius terhadap ancaman virus *Avian Influenza* (flu burung). Pemodelan penyebaran virus avian influenza ke dalam formula matematika menjadi diperlukan sebab dapat meramalkan jangkauan waktu populasi unggas dan manusia yang rentan ataupun yang terinfeksi virus ini.

Pemodelan matematika tersebut melibatkan suatu persamaan diferensial biasa orde satu non linier, selanjutnya digunakan metode numerik yaitu metode Runge-Kutta sepuluh untuk menganalisa model penyebaran virus Avian Influenza tersebut dengan menggunakan MATLAB sebagai software pembantu. oleh karena itu, bahasan bab ini mencakup teori-teori mengenai permasalahan yang dibutuhkan.

2.1 Avian Influenza (Flu Burung)

2.1.1 Pengertian Avian Influenza (Flu Burung)

Akhir akhir ini Avian Influenza atau lebih dikenal sebagai flu burung, khususnya virus H5N1 yang sangat patogen, telah menyita perhatian dunia. Penyebaran penyakit ini begitu cepat di antara unggas serta dapat menular ke manusia dengan dampak mortalitas yang tinggi, membuat masyarakat dunia menjadi gelisah.

Avian influenza atau flu burung adalah penyakit menular yang disebabkan virus Avian Influenza A dengan subtipe H1 sampai H16 dan N1 sampai N9. Virus ini menyerang berbagai jenis unggas, meliputi ayam, kalkun, unggas air, burung peliharaan dan burung liar. Terakhir virus ini dapat menginfeksi babi, harimau, kucing dan macan tutul. Pada umumnya virus avian influenza A tidak menye-

rang manusia, tetapi subtipe tertentu seperti H5N1, H7N7 yang bersifat sangat patogen dapat menyerang manusia dan mengakibatkan kematian. Terdapat beberapa cara penularan virus Avian Influenza A dari spesies unggas ke manusia antara lain melalui kontak langsung maupun tidak langsung dengan unggas yang sakit termasuk air liur dan tinja, udara dan alat-alat peternakan yang terkontaminasi dengan virus Avian Influenza. Saat ini pengobatan dengan *oseltamivir* dan *zanamivir* masih memberikan hasil yang baik terhadap virus avian influenza A H5N1. Meskipun berbagai aspek penyakit ini telah diketahui, tetapi masih terdapat bagian yang belum terungkap seperti pengembangan vaksin. Hingga kini belum ada vaksin yang dapat mencegah penyakit flu burung pada manusia.

Virus Avian Influenza dapat menimbulkan gejala penyakit pernafasan pada unggas, dari yang kurang ganas (*Low Pathogenic Avian Influenza/LPAI*) sampai yang bersifat sangat ganas (*Highly Pathogenic Avian Influenza/HPAI*). Masa inkubasi penyakit ini adalah 3 hari pada unggas diluar kandang, sedangkan untuk unggas di dalam kandang mencapai 14-21 hari. Hal ini tergantung pada jumlah virus, cara penularan, spesies atau jenis yang terinfeksi, dan kemampuan peternak untuk mendeteksi gejala klinis (DEPKES RI, 2010: 8).

Virus Avian Influenza diluar tubuh inangnya merupakan virus yang lemah, tidak tahan panas dan lemah terhadap zat *desinfektan/* pencuci hama. Dalam daging, virus ini mati pada pemasakan standar suhu 80C selama satu menit atau 70C selama 30 menit dan pada telur ayam, virus ini mati pada suhu 64C selama 4,5 menit. Dikandang ayam, virus Avian Influenza bertahan selama 2 minggu setelah *depopulasi* ayam, namun virus Avian Influenza dapat mati juga dengan *desinfektan*.

2.1.2 Pengaruh Pada Manusia

Pertama kali virus Avian Influenza ditemukan pada tahun 1878 di Itali, menyebabkan epidemi penyakit *Fowl Plague* pada ternak ayam dengan angka kematian 100 persen. Virus Avian Influenza secara normal tidak menginfeksi diluar spesies unggas dan babi. Baru pertama kali pada tahun 1997 di Hong Kong ter-

jadi wabah flu burung yang disebabkan virus Avian Influenza H5N1 yang patogen. Ketika itu telah terjadi penularan virus H5N1 dari spesies unggas ke manusia. Wabah flu burung tersebut menyebabkan enam penderita meninggal dari 18 kasus flu burung. Kini virus H5N1 terbukti dapat menginfeksi babi, harimau, macan tutul dan kucing. Pada Februari 2003 virus Avian Influenza A subtype H7N7 mulai menyerang daratan Eropa terutama Belanda. Wabah flu burung ini mengakibatkan seorang meninggal dunia dari 89 penderita. Pada akhir tahun 2003 sampai awal tahun 2004, wabah flu burung yang disebabkan virus H5N1 kembali merebak di berbagai negara Asia meliputi Korea Selatan, Jepang, China, Vietnam, Thailand, Kamboja dan Laos. Sedikitnya 100 juta ternak ayam telah dimusnahkan untuk menghentikan penularan. Wabah ini telah menginfeksi 35 orang dan mengakibatkan 24 penderita meninggal dunia. Kemudian wabah flu burung dengan cepat menjalar ke beberapa negara Asia Tenggara lainnya termasuk Indonesia. Sejak kasus flu burung pertama di Indonesia yang mengakibatkan meninggalnya seorang ayah beserta kedua anaknya pada pertengahan bulan Juli 2005 lalu, tingkat kewaspadaan sistem surveilan ditingkatkan baik di masyarakat maupun di institusi kesehatan seperti rumah sakit, puskesmas dan yang lainnya. Menurut laporan terakhir WHO, awal November 2005, data kumulatif kasus Avian Influenza A (H5N1) yaitu 122 kasus, dengan 62 penderita meninggal. Prevalensi tertinggi flu burung terjadi di Vietnam, terdapat 91 kasus, meninggal 41 penderita; disusul Thailand 20 kasus, meninggal 13 penderita; Indonesia 7 kasus, meninggal 4 penderita dan Kamboja terdapat 4 kasus yang keseluruhannya meninggal dunia (Kumala, 2005).

2.1.3 Model Penyebaran Avian Influenza

Model sebagai representasi dari bentuk penyebaran virus Avian Influenza (flu burung), akan diadopsi model SIRS, yaitu model yang mengkatagorikan suatu kasus menjadi kategori *susceptible*-rentan (S), *infectious*-terinfeksi (I), dan *recovered but susceptible*-sembuh namun rentan (RS). Beberapa dari kasus hanya membagi model menjadi dua katagori saja yaitu *susceptible* - rentan (S), dan

infectious - terinfeksi (I).

Pada kasus Avian Influenza (flu burung), kondisi *recovered but susceptible* - sembuh namun rentan (RS) untuk burung (unggas) memiliki kemungkinan yang kecil sekali bahkan tidak ada karena virus ini sangat mematikan untuk burung (unggas). Sedangkan untuk katagori manusia yang sembuh (*recovered*) adalah faktor atau unsur yang kembali lagi menjadi katagori rentan (*susceptible*) disebabkan kemungkinan untuk terinfeksi kembali setelah sembuh dari Avian Influenza, walaupun tentu saja terdapat kemungkinan timbul imun alami dalam tubuh manusia, namun imun yang terbentuk kemungkinan besar hanya bersifat sementara (*temporary immunity*).

Dengan demikian model ini memonitor dinamika sementara dari populasi burung (unggas) yang rentan - susceptible birds ($S_B(t)$), burung (unggas) yang terinfeksi - infectious birds ($I_B(t)$), manusia yang rentan - susceptible humans ($S_H(t)$), dan manusia yang terinfeksi - infectious humans ($I_H(t)$), sebagai fokus dari persamaan model. Adalah penting untuk menjelaskan bahwa $S_B = S_B(t) + I_B(t)$ menggambarkan total populasi dari burung (unggas) yang ada di lokasi tertentu, dan $S_H = S_H(t) + I_H(t)$ menggambarkan total pupulasi manusia di lokasi yang sama (Okosun, 2007: 9).

Model penyebaran virus Avian Influenza dalam bentuk sistem Persamaan Diferensial Biasa (PDB) orde satu yang dikemukakan oleh Kermac Mckendric dan telah ditulis oleh Okosun pada tahun 2007 (dalam Yustica, 2010) sebagai berikut :

$$\frac{dS_B}{dt} = N_B + (1 - \lambda_B)M_B - \alpha_B S_B \frac{I_B}{N_B} - \delta_B S_B, \quad (2.1)$$

$$\frac{dI_B}{dt} = \alpha_B S_B \frac{I_B}{N_B} - (\delta_B + d_B)I_B + \lambda_B M_B, \quad (2.2)$$

$$\frac{dS_H}{dt} = N_H \beta_H - \frac{\alpha_H S_H I_B}{N_B} - \delta_H S_H + v I_H, \quad (2.3)$$

$$\frac{dI_H}{dt} = \frac{\alpha_H S_H I_B}{N_B} - (\delta_H + d_H + v)I_H, \quad (2.4)$$

Interpretasi parameter-parameter dari persamaan tersebut diasumsikan berdasarkan penelitian dalam Jurnal Internasional Medwell (2007: 11) yang membahas simulasi numerik pada epidemi *avian influenza* (flu burung). Tetapan tersebut diberikan dengan nilai - nilai pada tabel 2.1 berikut:

Tabel 2.1: Interpretasi Parameter

Parameter	Deskripsi	N. Estimasi	Satuan
N_B	Jumlah total dari unggas di lokasi	1.000	ekor
N_H	Jumlah total dari manusia di lokasi	variabel	orang
β_B	Rata-rata tingkat kelahiran unggas	0,03	$hari^{-1}$
β_H	Rata-rata tingkat kelahiran manusia	0,001	$hari^{-1}$
λ_B	Peluang infeksi pada unggas migrasi	0,01	-
M_B	Jumlah total unggas migrasi	10	$hari^{-1}$
δ_B	Tingkat kematian alami unggas	$1/(365 \times 2)$	$hari^{-1}$
δ_H	Tingkat kematian alami manusia	$1/(365 \times 75)$	$hari^{-1}$
α_B	Tingkat transmisi infeksi dari unggas ke unggas	0,9	$hari^{-1}$
α_H	Tingkat transmisi infeksi dari unggas ke manusia	0,1	$hari^{-1}$
d_B	Tingkat kematian unggas karena infeksi flu burung	0,99	$hari^{-1}$
d_H	Tingkat kematian manusia karena infeksi flu burung	0,009	$hari^{-1}$
v	Tingkat kesembuhan manusia	$1/7$	$hari^{-1}$

Nilai parameter tersebut diperoleh dari jurnal okosun 2007, dimana penelitian tersebut diadakan di negara Nigeria. Model PDB yang menggambarkan penyebaran virus *Avian Influenza* (flu burung) cukup rumit sehingga tidak bisa diselesaikan dengan metode analitik. Dengan demikian, digunakan metode numerik untuk mencari solusi dari model PDB tersebut.

2.2 Nilai Awal

Data simulasi diambil berdasarkan keterangan dalam situs resmi yang dipublikasikan oleh Antara Jatim (awal Januari 2015). Model penyebaran virus Avian Influenza mempunyai nilai awal, dimana nilai-nilai awal tersebut antara lain : $S_B(0) = 300$; $I_B(0) = 165$; $S_H(0) = 60$; $I_H(0) = 7$.

2.3 Persamaan Diferensial Biasa

Persamaan penyebaran virus Avian Influenza berbentuk persamaan diferensial. Persamaan diferensial adalah persamaan yang memuat turunan satu (atau beberapa) fungsi yang tak diketahui (Finizio, 1988:1). Secara lebih sederhana persamaan diferensial merupakan persamaan yang mengandung beberapa turunan dari suatu fungsi. Ada dua jenis persamaan diferensial yaitu Persamaan Diferensial Biasa (PDB) dan Persamaan Diferensial Parsial (PDP). Untuk lebih jelas dapat dilihat definisi berikut:

Definisi 2.3.1. (Persamaan Diferensial) Suatu persamaan yang meliputi turunan fungsi dari satu atau lebih variabel bebas disebut persamaan diferensial. Selanjutnya jika turunan fungsi tersebut hanya bergantung pada satu variabel bebas maka disebut Persamaan Diferensial Biasa (PDB) dan bila bergantung pada lebih dari satu variabel bebas disebut Persamaan Diferensial Parsial (PDP) (Dafik, 1999a:1).

Berdasarkan definisi tersebut model penyebaran virus Avian Influenza tergolong Persamaan Diferensial Biasa (PDB). Selanjutnya pemaparan teori akan lebih ditekankan pada Persamaan Diferensial Biasa (PDB). Bentuk umum dari Persamaan Diferensial Biasa (PDB) adalah:

$$F(x, y, y', y'', \dots, y^n) = 0 \quad (2.5)$$

dengan n adalah orde dari PDB tersebut. Sehubungan dengan itu, Dafik (2008a:2) memberikan beberapa definisi sebagai berikut:

Definisi 2.3.2. (Orde) Orde dari suatu PDB adalah orde tertinggi dari turunan dalam persamaan $F(x, y, y', y'', \dots, y^n) = 0$

Definisi 2.3.3. (Linieritas dan Homogenitas) PDB orde n dikatakan linier bila dapat dinyatakan dalam bentuk $a_0(x)y^n + a_1(x)y^{n-1} + \dots + a_n(x)y = F(x)$, $a_0(x) \neq 0$, Selanjutnya :

1. bila tidak dapat dinyatakan dengan bentuk di atas dikatakan tak linier;

2. bila koefisien $a_0(x), a_1(x), \dots, a_n(x)$ konstan dikatakan mempunyai koefisien konstan. Bila tidak, dikatakan mempunyai koefisien variabel;
3. bila $F(x) = 0$ maka PDB tersebut dikatakan homogen. Bila tidak, disebut non homogen.

Model penyebaran virus Avian Influenza berbentuk sistem Persamaan Diferensial Biasa (PDB). Dengan demikian, akan dibahas mengenai sistem Persamaan Diferensial Biasa (PDB).

2.3.1 Sistem Persamaan Diferensial Biasa (PDB)

Suatu fenomena biasa dimodelkan dalam suatu sistem PDB. Sistem PDB adalah suatu sistem yang melibatkan beberapa persamaan diferensial yang memuat beberapa variabel terikat, variabel bebas dan parameter yang saling berkaitan satu sama lain. Variabel yang berada dalam sistem PDB berkaitan erat dengan faktor-faktor dalam suatu fenomena. Secara umum, solusi dari sistem PDB dikategorikan ke dalam tiga jenis solusi yaitu:

1. solusi umum, yaitu solusi yang mengandung konstanta esensial;
2. solusi khusus, yaitu solusi yang tidak mengandung konstanta esensial yang disebabkan oleh tambahan syarat awal pada suatu PDB;
3. solusi singular, yaitu solusi yang tidak didapat dari hasil mensubstitusikan suatu nilai pada konstanta solusi umumnya.

Persamaan diferensial biasa orde pertama dapat disajikan dalam bentuk berikut:

$$\frac{dy}{dx} = f(x, y) \text{ atau } y' = f(x, y) \quad (2.6)$$

Solusi dari persamaan ini adalah $y(x)$ yang memenuhi persamaan $y'(x) = f(x, y(x))$ di semua titik pada interval domain $[a, b]$. Selanjutnya persamaan (2.6) dikatakan merupakan masalah nilai awal bila solusi itu memenuhi nilai awal $y(a) = y_0$, sehingga persamaan tersebut dapat digambarkan sebagai:

$$y' = f(x, y), a \leq x \leq b$$

$$y(a) = y_0$$

Kemudian bila persamaan ini terdiri dari lebih dari satu persamaan yang saling terkait maka dikategorikan sebagai sistem persamaan diferensial. Sistem persamaan diferensial orde pertama dituliskan sebagai berikut:

$$y'_1 = f_1(t, y_1, y_2, \dots, y_n)$$

$$y'_2 = f_2(t, y_1, y_2, \dots, y_n)$$

$$\vdots$$

$$y'_n = f_n(t, y_1, y_2, \dots, y_n)$$

Atau dalam bentuk umum,

$$y'_i = f_i(t, y_1, y_2, \dots, y_n), \quad i = 1, 2, \dots, n; a \leq t \leq b \quad (2.7)$$

dan dengan nilai awal $y_1(a) = \alpha_1, y_2(a) = \alpha_2, \dots, y_n(a) = \alpha_n$ (Dafik, 2008b:1-2)

2.3.2 Metode Penyelesaian Sistem PDB

Metode yang dapat digunakan untuk menyelesaikan PDB pada dasarnya ada tiga yaitu analitik, kualitatif, dan numerik. Penjelasan masing-masing metode tersebut ialah sebagai berikut:

1. Metode Analitik

Representasi secara analitik dari suatu solusi bisa berbentuk salah satu dari bentuk berikut:

- bentuk eksplisit $y = F(x)$, dalam hal ini variabel terikat terisolasi secara penuh dan hanya nampak sebagai pangkat satu pada sisi suatu persamaan. Di sisi lain dari persamaan tersebut hanya mengandung ekspresi dalam variabel x atau konstanta;

- bentuk implisit adalah persamaan $F(x, y) = 0$ yang mengandung variabel bebas maupun variabel terikat tetapi tidak mengandung turunannya.

Kelebihan dari solusi secara analitik adalah formula yang diinginkan bisa diperoleh baik secara eksplisit maupun implisit. Namun tidak semua persamaan diferensial biasa dapat diselesaikan secara analitik, tergantung pada karakteristiknya (kelinieran dan homogenitas), artinya persamaan diferensial biasa pada tingkat orde tertentu sulit diselesaikan secara analitik. Di samping itu juga penyelesaian persamaan diferensial biasa secara analitik juga mempunyai keterbatasan pada alat hitungnya.

2. Metode Kualitatif

Representasi secara kualitatif dari solusi PDB memberikan informasi secara jelas tentang kelakuan solusi tanpa harus mendapatkan formula untuk solusi tersebut. Metode kualitatif digunakan untuk mencari kelakuan solusi persamaan diferensial biasa tak linier yang sulit dicari solusinya secara analitik.

3. Metode Numerik

Metode numerik sebagai alternatif untuk menyelesaikan sistem PDB, terutama untuk kasus PDB yang tidak bisa diselesaikan secara analitik maupun kualitatif. Solusi numerik pada dasarnya adalah aproksimasi untuk nilai variabel terikat pada nilai-nilai tertentu variabel bebas dengan tingkat ketelitian tertentu, sehingga solusi yang diperoleh mengandung kesalahan (*error*). Dalam hal ini biasanya solusi PDB berupa tabel nilai variabel terikat dan variabel bebas yang bersesuaian.

Pada prakteknya, mencari solusi PDB secara numerik adalah mencari barisan (x_i, y_i) . Metode numerik untuk menyelesaikan PDB bisa dibedakan menjadi dua yaitu metode satu langkah (*one step method*) dan metode banyak langkah (*multi step method*). Dikatakan satu langkah karena untuk menentukan y_{n+1} hanya memerlukan nilai y_n , sedangkan jika nilai y_{n+1} diperoleh menggunakan lebih

dari satu nilai y sebelumnya, misalkan $y_{n-2}, y_{n-1}, y_n, \dots$ maka metode tersebut dikatakan metode banyak langkah.

Menyelesaikan PDB secara numerik menjadi populer karena pada kenyataannya PDB yang muncul pada masalah sehari-hari tidaklah sederhana dan umumnya tidak dapat diselesaikan secara analitik bahkan mungkin tidak bisa diselesaikan secara kualitatif, tetapi bisa diselesaikan secara numerik (Dafik,1999a:4).

2.4 Konsep Dasar Metode Numerik

Menurut Sahid (2005:1) metode numerik adalah suatu metode untuk menyelesaikan masalah-masalah matematika dengan menggunakan sekumpulan operasi aritmatika sederhana dan operasi logika pada sekumpulan bilangan atau data numerik yang diberikan. Sedangkan menurut Sartono (2006:2) metode numerik secara harfiah adalah cara berhitung dengan menggunakan angka-angka, sedangkan secara istilah metode numerik adalah teknik yang digunakan untuk memformulasikan persoalan matematik sehingga dapat diselesaikan dengan operasi aritmatika biasa. Jadi metode numerik merupakan alat bantu pemecahan masalah matematika yang ampuh karena dengan menggunakan metode numerik masalah yang kompleks dan tidak dapat diselesaikan secara analitik dan kualitatif mampu diselesaikan.

Enam tahapan yang harus dilakukan dalam menyelesaikan persoalan dengan metode numerik adalah:

1. Pemodelan. Semua parameter dalam persoalan dimodelkan dalam bentuk persamaan matematika. Model matematika yang diperoleh pada tahap pertama bisa saja masih kompleks. Untuk memudahkan dan mempercepat kinerja komputer, model tersebut disederhanakan dengan membuang parameter yang dapat diabaikan;
2. Formulasi numerik. Setelah model matematika yang sederhana diperoleh, tahap selanjutnya adalah memformulasikannya secara numerik yaitu dengan menentukan metode numerik yang akan digunakan. Pemilihan metode didasarkan pada:

- apakah metode tersebut teliti?;
 - apakah metode tersebut mudah diprogram dan waktu eksekusinya cepat?
3. Algoritma. Menyusun algoritma dari metode numerik yang dipilih;
 4. Pemrograman. Algoritma yang telah disusun diterjemahkan dalam program komputer, dengan terlebih dahulu membuat *flowchart*-nya kemudian ditulis dalam bentuk program (dengan menggunakan salah satu *software* yang dapat mendukung untuk mempermudah pembuatannya, dalam penelitian ini penulis akan menggunakan MATLAB);
 5. Operasional. Program komputer dijalankan dengan data uji coba sebelum menggunakan data sebenarnya;
 6. Evaluasi. Bila program telah selesai dijalankan dengan menggunakan data sesungguhnya, hasil yang diperoleh diinterpretasi. Interpretasi meliputi analisis hasil perhitungan dan membandingkannya dengan prinsip dasar dan hasil-hasil empirik untuk menentukan kualitas solusi numerik.

Solusi suatu masalah dengan metode numerik tidak terlepas dari kegiatan menentukan fungsi aproksimasi yang paling dekat dengan fenomena yang diselesaikan. Walaupun terlihat sederhana, namun dalam prakteknya hal itu tidak mudah dilakukan mengingat betapa sulitnya menentukan fungsi-fungsi aproksimasi. Penentuan fungsi aproksimasi biasanya diperoleh dari pengembangan fungsi-fungsi polinomial, fungsi rasional ataupun fungsi eksponensial, yang kesemuanya harus dianalisis dengan intensif kedekatannya untuk menentukan fungsi mana yang dianggap paling sesuai.

Untuk itu, konsep seperti pengertian vektor dan matrik sangat diperlukan dalam memberikan gambaran lebih lengkap bagaimana menetapkan fungsi-fungsi aproksimasi. Khusus dalam kegiatan mengukur kedekatan fungsi aproksimasi dengan fungsi yang diaproksimasi diperlukan konsep teoritis tambahan yaitu konsep Norm dan Ruang Linier.

Norm terdiri dari dua jenis yaitu norm vektor dan norm matrik, sebagaimana digambarkan oleh Dafik (2010:20-21) bahwa norm dapat didefinisikan sebagai berikut:

Definisi 2.4.1. (Norm Vektor) Norm Vektor adalah pemetaan dari suatu fungsi terhadap setiap $x \in R^N$ yang disimbolkan dengan $\|x\|$ sedemikian hingga memenuhi sifat-sifat di bawah ini:

1. $\|x\| > 0$ untuk $x \neq 0$, atau $\|x\| = 0$ untuk $x = 0$
2. $\|\alpha x\| = \alpha \|x\|$
3. $\|x + y\| \leq \|x\| + \|y\|$

Definisi 2.4.2. (Norm Matrik) Norm matrik adalah pemetaan dari suatu fungsi terhadap setiap $A \in R^{N \times N}$ yang disimbolkan dengan $\|A\|$ sedemikian hingga memenuhi sifat-sifat dibawah ini :

1. $\|A\| > 0$ untuk $A \neq 0$, atau $\|A\| = 0$ untuk $A = 0$
2. $\|aA\| = a \|A\|$
3. $\|A + B\| \leq \|A\| + \|B\|$
4. $\|AB\| \leq \|A\| \|B\|$

dimana A, B adalah matrik dengan ordo $(n \times n)$ dan a adalah suatu konstanta.

Sementara itu Dafik (2010:21-22) memberikan definisi tentang Ruang Linier dan Ruang Linier Norm. Definisi tersebut ialah:

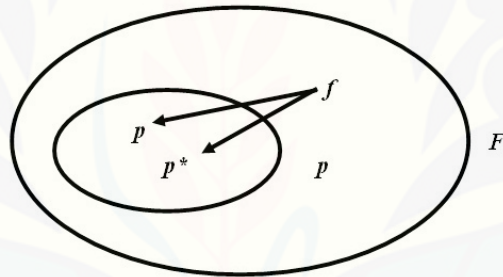
Definisi 2.4.3. (Ruang Linier (RL)) Himpunan F dikatakan suatu ruang linier bila operasi penjumlahan dan perkalian terdefinisi di dalamnya sehingga $f, g \in F$ dan $\alpha f + \beta g \in F$ untuk $\forall f, g \in F$.

Definisi 2.4.4. (Ruang Linier Norm (RLN)) F dikatakan suatu ruang linier norm bila F adalah merupakan RL dan terdapat fungsi norm sehingga

1. $\|f\| > 0$ untuk $f \neq 0$, atau $\|f\| = 0$, untuk $f = 0$
2. $\|\alpha f\| = \alpha\|f\|$
3. $\|f + g\| \leq \|f\| + \|g\|$

untuk semua $f, g \in F$

Dari beberapa gambaran definisi norm dan ruang linier di atas, maka fungsi aproksimasi yang paling dekat (p^*) dengan fungsi yang diaproksimasi (f) dapat ditunjukkan dengan membandingkan norm dari ($f - p^*$) dan norm dari ($f - p$), dimana p adalah fungsi-fungsi aproksimasi lainnya. Dengan demikian misal $f \in F$ dan $f \in P$ maka teknik aproksimasi sebenarnya adalah masalah untuk menentukan $p^* \in P$ sedemikian hingga $\|f - p^*\| \leq \|f - p\|, \forall p \in P$. Selanjutnya p^* disebut aproksimasi terbaik terhadap f , dan dapat digambarkan dalam diagram Venn pada gambar (2.1).



Gambar 2.1: Diagram Aproksimasi

Untuk meminimalisasi kesalahan dalam teknik numerik diatasi dengan teknik penghitungan berulang dan setiap satu kali perulangan iterasi dievaluasi dan dikonfirmasi dengan toleransi e yang telah ditetapkan. Artinya perulangan (*looping*) itu dilakukan agar $\|f - p^*\| \leq e$. Apabila hal ini dipenuhi, proses iterasi itu akan otomatis dihentikan karena target akurasi sudah dicapai oleh komputer, dengan kata lain $\|f - p^*\|$ konvergen atau $\lim_{n \rightarrow \infty} \|f - p^*\| = 0$. Solusi numerik pada dasarnya adalah aproksimasi untuk nilai variabel terikat pada nilai-nilai tertentu, sehingga solusi yang diperoleh mengandung kesalahan (*error*).

2.5 Aturan Matematika yang Digunakan dalam Penelitian

Proses menurunkan formula metode Runge-Kutta orde sepuluh memerlukan beberapa aturan matematis. Berikut akan dijelaskan tentang aturan-aturan tersebut.

1. **Aturan Taylor.** Deret Taylor memegang peranan yang sangat penting dalam analisis numerik. Jika $f(x)$ mempunyai sejumlah $(n + 1)$ turunan yang kontinu pada $[a, b]$ dan c merupakan suatu titik pada $[a, b]$, maka untuk semua $x \in [a, b]$ terdapat,

$$f(x) = f(c) + f'(c)(x - c) + \frac{f''(c)(x - c)^2}{2!} + \dots + \frac{f^{(n)}(c)(x - c)^n}{n!} + R_{n+1}(x) \quad (2.8)$$

dimana,

$$R_{n+1}(x) = \frac{1}{n!} \int_c^x (x - s)^n f^{(n+1)}(s) ds \quad (2.9)$$

2. Aturan Rantai untuk Turunan Parsial

$$y^{(1)} = f(x, y) \quad (2.10)$$

$$y^{(2)} = \frac{\partial f}{\partial x} \frac{\partial x}{\partial x} + \frac{\partial f}{\partial y} \frac{\partial y}{\partial x} \quad (2.11)$$

$$= f_x + f_y y' = f_x + f_y f \quad (2.12)$$

$$y^{(3)} = f_{xx} + 2f_{xy}f + f_{yy}f^2 + f_y(f_x + f_y f) \quad (2.13)$$

$$f(x + m, y + n) = f(x, y) + (m \frac{\partial}{\partial x} + n \frac{\partial}{\partial y})f + \frac{1}{2!} (m \frac{\partial}{\partial x} + n \frac{\partial}{\partial y})^2 f + \dots \quad (2.14)$$

(Dafik, 2009:77)

3. **Aturan Rantai.** Jika fungsi $f(x, y, \dots, z)$ mempunyai turunan-turunan parsial pertama yang kontinu terhadap masing-masing variabel, dan $x =$

$x(t), y = y(t), \dots, z = z(t)$ merupakan fungsi-fungsi yang dapat dideferensiasi dan kontinu, maka

$$g'(t) = \frac{\partial f}{\partial x}x'(t) + \frac{\partial f}{\partial y}y'(t) + \dots + \frac{\partial f}{\partial z}z'(t) \quad (2.15)$$

2.6 Metode Runge-Kutta

Dalam analisis numerik, metode Runge-Kutta merupakan metode yang penting untuk solusi hampiran dari Persamaan Diferensial Biasa (PDB). Metode tersebut dikembangkan oleh ahli matematika bernama Carl Runge dan Martin Wilhelm Kutta sekitar tahun 1900. Dasar pemikiran dari metode Runge-Kutta ialah untuk mempertahankan hampiran Taylor (dalam hubungan dengan kesalahan yang terbawa) sementara menghilangkan kepentingan untuk menghitung berbagai turunan parsial dari f yang tersertakan (Finizio, 1988:271). Metode Runge-Kutta berusaha mencapai derajat ketelitian yang lebih tinggi, dan sekaligus menghindarkan keperluan mencari turunan tingkat tinggi, dengan jalan mengevaluasi fungsi $f(x, y)$ pada titik terpilih dalam setiap subselang (Conte, 1993:329). Metode Runge-Kutta mempunyai tiga sifat khas yaitu:

1. metode Runge-Kutta adalah metode satu langkah: untuk memperoleh $y_{(m+1)}$, kita hanya perlu informasi yang tersedia pada titik sebelumnya, x_m, y_m ;
2. metode Runge-Kutta sesuai dengan deret Taylor sampai dengan suku h^p , dimana p berbeda untuk metoda berbeda dan disebut *orde* metode tersebut;
3. metode Runge-Kutta tidak memerlukan evaluasi setiap turunan $f(x, y)$, tetapi hanya fungsi f itu sendiri (Dorn, 1986:381).

Secara umum metode Runge-Kutta didefinisikan:

$$y_{n+1} = y_n + h \sum_{i=1}^m b_i k_i \quad (2.16)$$

dimana,

$$k_i = f(x_n + c_i h, y_n + h \sum_{j=1}^{i-1} a_{ij} k_j), \quad i = 1, 2, \dots, m \quad (2.17)$$

dengan asumsi bahwa:

$$c_i = \sum_{j=1}^m a_{ij}, \text{ dan } \sum_{i=1}^m b_i = 1 \quad (2.18)$$

Keterangan:

- y_{n+1} = solusi numeris atau solusi aproksimasi pada $n + 1$
- y_n = solusi numeris atau solusi aproksimasi pada n
- h = ukuran langkah yang telah ditetapkan ($x - x_0$)
- a, b, c = tetapan unik
- k = titik terpilih dalam tiap subselang

Tetapan unik a, b , dan c merupakan tetapan yang harus ditentukan sedemikian rupa sehingga persamaan (2.16) akan sama dengan algoritma Taylor dari tingkatan setinggi mungkin, dengan demikian solusi aproksimasi akan mendekati solusi eksak. Untuk mempermudah pencarian nilai dari tetapan, dapat dibuat suatu gambaran nilai tetapan yang disebut *Butcher array*. Gambaran tersebut disesuaikan dengan persamaan (2.18). Gambaran tetapan secara lengkap dapat ditulis:

0				
c_2	a_{21}			
c_3	a_{31}	a_{32}		
\vdots	\vdots	\vdots	\ddots	
c_m	a_{m1}	a_{m2}	\dots	a_{mm-1}
	b_1	b_2	\dots	$b_{m-1} \quad b_m$

Keterangan gambar:

Sumbu vertikal yang memisahkan tetapan c dan tetapan a , menyatakan per-

samaan $c_i = \sum_{j=1}^m a_{ij}$. Tetapan b di bawah sumbu horisontal menyatakan persamaan $\sum_{i=1}^m b_i = 1$. Dari *Butcher array* di atas, didefinisikan vektor c dan b berdimensi m dan matriks $\mathbf{A}_{m \times m}$ oleh:

$$c = [c_1, c_2, \dots, c_m]^T, b = [b_1, b_2, \dots, b_m]^T, \mathbf{A} = [a_{ij}]. \quad (2.19)$$

Perhatikan bahwa berdasarkan persamaan (2.18), tetapan c adalah jumlah baris dari \mathbf{A} . Metode Runge-Kutta secara lengkap dirinci oleh *Butcher array* berikut:

$$\begin{array}{c|c} c & \mathbf{A} \\ \hline & b^T \end{array}$$

Menurut Lambert (1997:150-151) ada tiga jenis metode Runge-Kutta:

1. metode eksplisit:

$$a_{ij} = 0, j \geq i, j = 1, 2, \dots, m \Leftrightarrow \mathbf{A} \text{ berbentuk segitiga bawah yang tegas.}$$

2. metode semi-implisit:

$$a_{ij} = 0, j > i, j = 1, 2, \dots, m \Leftrightarrow \mathbf{A} \text{ berbentuk segitiga bawah.}$$

3. metode implisit:

$$a_{ij} \neq 0 \text{ untuk beberapa } j > i \Leftrightarrow \mathbf{A} \text{ tidak berbentuk segitiga bawah.}$$

Dalam penelitian ini metode Runge-Kutta yang digunakan adalah metode eksplisit karena untuk $j \geq i, j = 1, 2, \dots, m$ berlaku $a_{ij} = 0$.

Berikut adalah beberapa definisi mengenai metode Runge-Kutta (Dafik, 2009:70):

Definisi 2.6.1. (*Kesalahan Global*) *Kesalahan global didefinisikan sebagai:*

$$e_n = y(x_n) - y_n$$

Keterangan :

$$e_n = \text{kesalahan global}$$

$$y(x_n) = \text{solusi analitik atau solusi eksak}$$

$$y_n = \text{solusi numeris atau solusi aproksimasi}$$

Definisi 2.6.2. (Kesalahan Pemenggalan Lokal) Kesalahan Pemenggalan Lokal adalah kesalahan yang ditimbulkan oleh perumusan suatu metode dalam bentuk:

$$I_n = y(x_{n+1}) - y_{n+1}$$

Keterangan :

I_n = kesalahan pemenggalan lokal

$y(x_{n+1})$ = solusi analitik atau solusi eksak

y_{n+1} = solusi numeris atau solusi aproksimasi

Definisi 2.6.3. (Orde) Suatu metode dikatakan berorde p bila $I_n = \phi(h^{p+1})$

Definisi 2.6.4. (Konsisten) Suatu metode dikatakan konsisten bila ordernya minimal satu.

Penelitian ini akan menggunakan definisi di atas. Berdasarkan definisi 2.6.2 akan diperoleh kesalahan pemenggalan lokal (I_n) dari formula yang ditemukan sehingga peneliti dapat menggunakan hasilnya untuk membuktikan bahwa formula tersebut merupakan formula metode Runge-Kutta orde sepuluh yang memenuhi definisi 2.6.3 dan merupakan metode yang konsisten karena memenuhi definisi 2.6.4.

2.6.1 Konsep Konvergensi Metode Runge-Kutta

Golub (dalam Dafik, 2009:66-70) memberikan beberapa konsep konvergensi dalam metode Runge-Kutta dengan definisi sebagai berikut:

Definisi 2.6.5. (Syarat Lipschitz) Suatu fungsi $f(t, y)$ dikatakan memenuhi Syarat Lipschitz dalam variabel y di suatu domain $D \in \mathbb{R}^2$ jika ada konstanta $L > 0$ sedemikian hingga

$$\|f(t, y_1) - f(t, y_2)\| \leq L\|y_1 - y_2\|$$

untuk sembarang $(t, y_1), (t, y_2) \in D$. Selanjutnya konstanta L disebut sebagai konstanta Lipschitz.

Syarat Lipschitz digunakan dalam pembuktian konvergensi dari metode Runge-Kutta pada penelitian ini. Khususnya, konstanta L akan dipakai untuk menyederhanakan fungsi-fungsi yang berkaitan dengan pembuktian konvergensi.

Definisi 2.6.6. Suatu metode dikatakan konvergen bila

$$\max_{0 \leq i \leq n} \|y(x_i) - y_i\| \rightarrow 0, \text{ untuk } h \rightarrow 0$$

dimana h adalah besarnya grid.

$h \rightarrow 0$ artinya kesalahan global dari metode Runge-Kutta harus mendekati nol. Pembuktian konvergensi metode Runge-Kutta secara teoritis pada penelitian ini didasarkan pada definisi 2.6.1, definisi 2.6.5, dan definisi 2.6.6.

2.6.2 Penurunan Formula Metode Runge-Kutta

Berdasarkan persamaan (2.16), (2.17) dan (2.18) kita dapat menentukan proses penurunan metode Runge-Kutta Orde m . Penurunan metode Runge-Kutta eksplisit 3-tahap (orde tiga) dapat ditulis sebagai berikut (Lambert, 1997:153-155):

$$\begin{aligned} y_{n+1} &= y_n + h(b_1k_1 + b_2k_2 + b_3k_3) \\ k_1 &= f(x_n, y_n) \\ k_2 &= f(x_n + hc_2, y_n + hc_2k_1) \\ k_3 &= f(x_n + hc_3, y_n + h(c_3 - a_{32})k_1 + ha_{32}k_2) \end{aligned} \quad (2.20)$$

Kita mengasumsikan bahwa $f(x, y)$ cukup halus, dan digunakan notasi singkat

yaitu:

$$f := f(x, y), \quad f_x := \frac{\partial f(x, y)}{\partial x}, \quad f_{xx} := \frac{\partial^2 f(x, y)}{\partial x^2}, \quad f_{xy} (\equiv f_{yx}) := \frac{\partial^2 f(x, y)}{\partial x \partial y}, \quad \text{dst}$$

Semua dievaluasi pada titik $(x_n, y(x_n))$. Kemudian, dengan mengekspan $y(x_{n+1})$ ke dalam x_n sebagai deret Taylor, kita mempunyai:

$$y(x_{n+1}) = y(x_n) + hy^{(1)}(x_n) + \frac{1}{2}h^2y^{(2)}(x_n) + \frac{1}{6}h^3y^{(3)}(x_n) + 0(h^4).$$

Sekarang,

$$\begin{aligned} y^{(1)}(x_n) &= f, \\ y^{(2)}(x_n) &= f_x + f_y y' = f_x + f f_y \\ y^{(3)}(x_n) &= f_{xx} + f_{xy} f + f(f_{yx} + f_{yy} f) + f_y(f_x + f f_y) \\ &= f_{xx} + 2f f_{xy} + f^2 f_{yy} + f_y(f_x + f f_y). \end{aligned} \quad (2.21)$$

Setelah itu, kita mempersingkat lagi notasi dengan mendefinisikan:

$$F := f_x + f f_y, \quad G := f_{xx} + 2f f_{xy} + f^2 f_{yy}, \quad (2.22)$$

Sehingga kita bisa menuliskan ekspansi untuk $y(x_{n+1})$ sebagai:

$$y(x_{n+1}) = y(x_n) + hf + \frac{1}{2}h^2F + \frac{1}{6}h^3(Ff_y + G) + 0(h^4). \quad (2.23)$$

Supaya bisa menggunakan definisi (2.6.2), kita memerlukan ekspansi yang sama untuk y_{n+1} . Mengekspan k_i pada persamaan (2.20) dapat dilakukan dengan menggunakan aturan rantai dan mensubstitusikan notasi F dan G . Akibatnya

diperoleh:

$$\begin{aligned}
 k_1 &= f \\
 k_2 &= f(x_n + hc_2, y_n + hc_2k_1) \\
 &= f + (hc_2f_x + hc_2k_1f_y) + \frac{1}{2}(hc_2f_x + hc_2k_1f_y)^2 + 0(h^3) \\
 &= f + hc_2(f_x + ff_y) + \frac{1}{2}h^2c_2^2(f_{xx} + 2ff_{xy} + f^2f_{yy}) + 0(h^3) \\
 &= f + hc_2F + \frac{1}{2}h^2c_2^2G + 0(h^3) \\
 k_3 &= f(x_n + hc_3, y_n + h(c_3 - a_{32})k_1 + ha_{32}k_2) \\
 &= f + h\{c_3f_x + [(c_3 - a_{32})k_1 + a_{32}k_2]f_y\} \\
 &\quad + \frac{1}{2}h^2\{c_3^2f_{xx} + 2c_3[(c_3 - a_{32})k_1 + a_{32}k_2]f_{xy} \\
 &\quad + [(c_3 - a_{32})k_1 + a_{32}k_2]^2f_{yy}\} + 0(h^3) \\
 &= f + hc_3F + h^2(c_2a_{32}Ff_y + \frac{1}{2}c_3^2G) + 0(h^3). \tag{2.24}
 \end{aligned}$$

Setelah itu kita mensubstitusikan nilai k_1 , k_2 , dan k_3 ke ekspansi y_{n+1} pada persamaan (2.20) sehingga diperoleh:

$$\begin{aligned}
 y_{n+1} &= y(x_n) + h(b_1 + b_2 + b_3)f + h^2(b_2c_2 + b_3c_3)F \\
 &\quad + \frac{1}{2}h^3[2b_3c_2a_{32}Ff_y + (b_2c_2^2 + b_3c_3^2)G] + 0(h^4). \tag{2.25}
 \end{aligned}$$

Dengan membandingkan persamaan (2.25) dengan ekspansi deret Taylor pada persamaan (2.23) maka diperoleh sistem persamaan pada lemma (2.6.1).

◇ **Lema 2.6.1.** *Metode Runge-Kutta 3-tahap yang sekaligus berorde 3 mempunyai sifat sebagai berikut:*

$$\begin{aligned}
 b_1 + b_2 + b_3 &= 1 \\
 b_2 c_2 + b_3 c_3 &= \frac{1}{2} \\
 b_2 c_2^2 + b_3 c_3^2 &= \frac{1}{3} \\
 b_3 c_2 a_{32} &= \frac{1}{6}
 \end{aligned}$$

Dengan menyelesaikan lemma (2.6.1) diperoleh akibat berikut:

Akibat 2.6.1. *Formula Orde Tiga Heun dengan Butcher array*

$$\begin{array}{c|ccc}
 0 & & & \\
 \frac{1}{3} & \frac{1}{3} & & \\
 \frac{2}{3} & 0 & \frac{2}{3} & \\
 \hline
 & \frac{1}{4} & 0 & \frac{3}{4}
 \end{array}$$

Untuk metode Runge-Kutta 2-tahap diperoleh lemma berikut:

◇ **Lema 2.6.2.** *Metode Runge-Kutta dua tahap yang sekaligus berorde 2 mempunyai sifat sebagai berikut:*

$$\begin{aligned}
 a_{21} &= c_2 \\
 b_1 + b_2 &= 1 \\
 b_2 c_2 &= \frac{1}{2}
 \end{aligned}$$

(Dafik, 2000:14)

Untuk metode Runge-Kutta 4-tahap diperoleh lemma berikut:

◇ **Lema 2.6.3.** *Metode Runge-Kutta empat tahap yang sekaligus berorde 4 mempunyai sifat sebagai berikut:*

$$\begin{aligned}
 b_1 + b_2 + b_3 + b_4 &= 1 \\
 b_2c_2 + b_3c_3 + b_4c_4 &= \frac{1}{2} \\
 b_2c_2^2 + b_3c_3^2 + b_4c_4^2 &= \frac{1}{3} \\
 b_3c_2a_{32} + b_4(c_2a_{42} + c_3a_{43}) &= \frac{1}{6} \\
 b_2c_2^3 + b_3c_3^3 + b_4c_4^3 &= \frac{1}{4} \\
 b_3c_2^2a_{32} + b_4(c_2^2a_{42} + c_3^2a_{43}) &= \frac{1}{12} \\
 c_2c_3a_{32}b_3 + c_4(c_2a_{42} + c_3a_{43})b_4 &= \frac{1}{8} \\
 c_2a_{32}a_{43}b_4 &= \frac{1}{24}
 \end{aligned}$$

(Fausett, 2008:483)

2.6.3 Metode Runge-Kutta Orde Sepuluh

Metode Runge-Kutta merupakan metode satu langkah yang berusaha mencapai derajat ketelitian yang lebih tinggi. Sampai saat ini perkembangan penurunan formula metode tersebut masih mencapai orde sembilan yang sebagian hasilnya dapat dilihat pada tabel (2.2). Sehingga untuk penelitian selanjutnya peneliti tertarik untuk menganalisis metode Runge-Kutta orde sepuluh. Selain itu, peneliti juga tertarik untuk mengetahui efektivitas dan efisiensi metode Runge-Kutta orde sepuluh.

Tabel 2.2: Koefisien-koefisien Metode Runge-Kutta

No	Metode	Hasil	sumber
1	Metode	$y_{n+1} = y_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4)$	Faisol

No	Metode	Hasil	sumber
	Runge Kutta orde 4 untuk model penyebaran virus dengue	$k_1 = f(x_n, y_n)$ $k_2 = f(x_n + \frac{h}{2}, y_n + \frac{h}{2}k_1)$ $k_3 = f(x_n + \frac{h}{2}, y_n + \frac{h}{2}k_2)$ $k_4 = f(x_n + h, y_n + hk_3)$	
2	Metode Runge Kutta orde 5 untuk model penyebaran virus avian influenza	$y_{n+1} = y_n + \frac{h}{6}(k_1 + 4k_4 + 2k_5)$ $k_1 = f(x_n, y_n)$ $k_2 = f(x_n + \frac{h}{3}, y_n + \frac{h}{3}k_1)$ $k_3 = f(x_n + \frac{h}{3}, y_n + \frac{h}{6}(k_1 + k_2))$ $k_4 = f(x_n + \frac{h}{2}, y_n + \frac{h}{8}(k_1 + 3k_3))$ $k_5 = f(x_n + h, y_n + h(\frac{1}{2}k_1 - \frac{3}{2}k_3 + 2k_4))$	Yustica
3	Metode Runge Kutta orde 6 untuk model penyebaran virus avian influenza	$y_{n+1} = y_n + \frac{h}{120}(11k_1 + 81k_2 - 64k_3 + 81k_5 + 11k_6)$ $k_1 = f(x_n, y_n)$ $k_2 = f(x_n + \frac{h}{3}, y_n + \frac{h}{3}k_1)$ $k_3 = f(x_n + \frac{h}{2}, y_n + \frac{h}{4}(k_1 + k_2))$ $k_4 = f(x_n + \frac{h}{2}, y_n + \frac{h}{2}k_1)$ $k_5 = f(x_n + \frac{2h}{3}, y_n + \frac{h}{81}(58k_1 + 64k_2 - 36k_3 - 32k_4))$ $k_6 = f(x_n + h, y_n + \frac{h}{11}(-25k_1 + 6k_2 - k_3 + 4k_4 + 27k_5))$	Bukaryo
4	Metode Runge Kutta orde 7 untuk model penyebaran penyakit tuberculosis	$y_{n+1} = y_n + \frac{h}{3360}(290k_1 - 2008k_2 + 2403k_4 + 2376k_6 + 99k_7)$ $k_1 = f(x_n, y_n)$ $k_2 = f(x_n + \frac{h}{2}, y_n + \frac{h}{2}k_1)$ $k_3 = f(x_n + \frac{h}{2}, y_n + \frac{h}{4}(k_1 + k_2))$ $k_4 = f(x_n + \frac{h}{2}, y_n + \frac{h}{27}(7k_1 + k_2 + k_3))$ $k_5 = f(x_n + \frac{2h}{3}, y_n + \frac{h}{15}(k_1 + 2k_2 + k_3 + k_4))$ $k_6 = f(x_n + \frac{2h}{3}, y_n + \frac{h}{216}(46k_1 + k_2 + k_3 + 24k_4 + 72k_5))$	Jakfar

No	Metode	Hasil	sumber
		$k_7 = f(x_n + h, y_n + \frac{h}{299}(183k_1 - 100k_2 + 996k_3 - 264k_4 + 720k_5 + 756k_6))$	
5	Metode Runge Kutta orde 8 untuk transimi penyakit malaria	$y_{n+1} = y_n + \frac{h}{120960}(5257k_1 + 25039k_2 + 9261k_3 + 20923k_4 + 20923k_5 + 9261k_6 + 25039k_7 + 5257k_8)$ $k_1 = f(x_n, y_n)$ $k_2 = f(x_n + \frac{h}{7}, y_n + \frac{h}{7}k_1)$ $k_3 = f(x_n + \frac{2h}{7}, y_n + \frac{h}{1323}(7538k_1 - 7160k_2))$ $k_4 = f(x_n + \frac{3h}{7}, y_n + \frac{h}{5978}(459k_1 + 4882k_2 - 2869k_3))$ $k_5 = f(x_n + \frac{4h}{7}, y_n + \frac{h}{427}(-693k_1 + 682k_2 - 211k_3 + 466k_4))$ $k_6 = f(x_n + \frac{5h}{7}, y_n + \frac{h}{378}(-79k_1 + 322k_2 + 224k_3 + 126k_4 - 323k_5))$ $k_7 = f(x_n + \frac{6h}{7}, y_n + \frac{h}{3577}(-2537k_1 + 2568k_2 + 1021k_3 + 511k_4 + 511k_5 + 992k_6))$ $k_8 = f(x_n + h, y_n + \frac{h}{1502}(-61k_1 + 102k_2 + 428k_3 - 112k_4 + 126k_5 + 242k_6 + 777k_7))$	Ardhilia
6	Metode Runge Kutta orde 9 untuk sistem kekebalan tubuh terhadap infeksi <i>mycobacterium tuberculosis</i>	$y_{n+1} = y_n + \frac{h}{105000}(5062k_1 + 27357k_2 + 1260k_3 + 1275k_4 + 35050k_5 - 16800k_6 + 19440k_7 + 27273k_8 + 5083k_9)$ $k_1 = f(x_n, y_n)$ $k_2 = f(x_n + \frac{h}{6}, y_n + \frac{h}{6}k_1)$ $k_3 = f(x_n + \frac{h}{3}, y_n + \frac{h}{6}(k_1 + k_2))$ $k_4 = f(x_n + \frac{h}{3}, y_n + \frac{h}{3}(-k_1 + k_2 + k_3))$ $k_5 = f(x_n + \frac{h}{2}, y_n + \frac{h}{10}(k_1 + 2k_2 + k_3 + k_4))$ $k_6 = f(x_n + \frac{2h}{3}, y_n + \frac{h}{48}(-10k_1 + 2k_2 + 24k_3 + 9k_4 + 6k_5))$ $k_7 = f(x_n + \frac{2h}{3}, y_n + \frac{h}{240}(65k_1 + 27k_2$	Anggraeni

No	Metode	Hasil	sumber
		$+27k_3 - 8k_4 + 93k_5 - 46k_6))$ $k_8 = f(x_n + \frac{5h}{6}, y_n + \frac{h}{5.4546}(-13291k_1$ $+24768k_2 + 942k_3 + 3336k_4 + 22542k_5$ $+6612k_6 + 546k_7))$ $k_9 = f(x_n + h, y_n + \frac{h}{5083}(-4844k_1 + 984k_2$ $+2912k_3 + 125k_4 + 296k_5 + 1008k_6$ $+102k_7 + 4500k_8))$	

Berdasarkan persamaan (2.16), metode Runge-Kutta Orde sepuluh didefinisikan dengan:

$$y_{n+1} = y_n + h \sum_{i=1}^m b_i k_i \quad \text{dimana } m = 10$$

$$= y_n + h(b_1 k_1 + b_2 k_2 + b_3 k_3 + b_4 k_4 + b_5 k_5 + b_6 k_6 + b_7 k_7 + b_8 k_8 + b_9 k_9 + b_{10} k_{10})$$

dimana,

$$k_1 = f(x_n, y_n)$$

$$k_2 = f(x_n + c_2 h, y_n + h a_{21} k_1)$$

$$k_3 = f(x_n + c_3 h, y_n + h(a_{31} k_1 + a_{32} k_2))$$

$$k_4 = f(x_n + c_4 h, y_n + h(a_{41} k_1 + a_{42} k_2 + a_{43} k_3))$$

$$k_5 = f(x_n + c_5 h, y_n + h(a_{51} k_1 + a_{52} k_2 + a_{53} k_3 + a_{54} k_4))$$

$$k_6 = f(x_n + c_6 h, y_n + h(a_{61} k_1 + a_{62} k_2 + a_{63} k_3 + a_{64} k_4 + a_{65} k_5))$$

$$k_7 = f(x_n + c_7 h, y_n + h(a_{71} k_1 + a_{72} k_2 + a_{73} k_3 + a_{74} k_4 + a_{75} k_5 + a_{76} k_6))$$

$$k_8 = f(x_n + c_8 h, y_n + h(a_{81} k_1 + a_{82} k_2 + a_{83} k_3 + a_{84} k_4 + a_{85} k_5 + a_{86} k_6 + a_{87} k_7))$$

$$k_9 = f(x_n + c_9h, y_n + h(a_{91}k_1 + a_{92}k_2 + a_{83}k_3 + a_{94}k_4 + a_{95}k_5 + a_{96}k_6 + a_{97}k_7 + a_{98}k_8))$$

$$k_{10} = f(x_n + c_{10}h, y_n + h(a_{101}k_1 + a_{102}k_2 + a_{103}k_3 + a_{104}k_4 + a_{105}k_5 + a_{106}k_6 + a_{107}k_7 + a_{108}k_8 + a_{109}k_9))$$

Untuk menentukan koefisien $b = [b_1, b_2, \dots, b_{10}]^T$ dan $\mathbf{A} = [a_{ij}]$ serta untuk mengetahui kekonvergenan metode Runge-Kutta orde sepuluh maka akan dikaji lebih lanjut dalam pembahasan.

2.7 Efektivitas dan Efisiensi

Penelitian ini bertujuan untuk mengetahui efektivitas dan efisiensi dari suatu metode numerik dalam menyelesaikan suatu permasalahan. Metode numerik yang digunakan dalam penelitian ini adalah metode Runge-Kutta orde sepuluh. Untuk itu akan dijelaskan tentang efektivitas dan efisiensi.

Menurut kamus besar bahasa Indonesia (dalam Wibisono, 2010), kata efektif berarti ada efeknya (akibatnya, pengaruhnya, kesannya); manjur atau mujarab (tentang obat); dapat membawa hasil; berhasil guna (tentang usaha, tindakan); mulai berlaku (tentang undang-undang, peraturan). Dalam penelitian ini arti efektif yang dipakai adalah dapat berhasil guna. Sedangkan definisi dari kata efektivitas yaitu suatu pencapaian tujuan secara tepat atau memilih tujuan-tujuan yang tepat dari serangkaian alternatif atau pilihan cara dan menentukan pilihan dari beberapa pilihan lainnya. Sehingga efektivitas dapat diartikan sebagai pengukuran keberhasilan dalam pencapaian tujuan-tujuan yang telah ditentukan. Misalnya jika suatu pekerjaan dapat selesai dengan pemilihan cara-cara yang telah ditentukan, maka cara tersebut adalah benar atau efektif.

Sedangkan arti kata efisien menurut kamus besar bahasa Indonesia (dalam Wibisono, 2010) yaitu tepat atau sesuai untuk mengerjakan (menghasilkan) sesuatu (dengan tidak membuang-buang waktu, tenaga, biaya), mampu menjalankan tugas dengan tepat dan cermat, berdaya guna, bertepatan guna. Sejalan dengan definisi tersebut maka efisiensi berarti penggunaan sumber daya secara minimum

guna pencapaian hasil yang optimum. Efisiensi menganggap bahwa tujuan-tujuan yang benar telah ditentukan dan berusaha untuk mencari berbagai cara yang terbaik untuk mencapai tujuan tersebut. Efisiensi hanya dapat dievaluasi dengan penilaian relatif serta membandingkan antara masukan dan keluaran yang diterima. Misalnya suatu pekerjaan dapat dikerjakan dengan cara X dan cara Y. Cara X dapat dikerjakan dalam waktu 2 jam sedangkan cara Y dapat dikerjakan dalam waktu 4 jam. Dengan demikian cara X lebih efisien bila dibandingkan dengan cara Y.

Berdasarkan penjelasan di atas kita dapat mengetahui perbedaan antara efektif dan efisien. Efektif berarti melakukan sesuatu yang benar. Sedangkan efisien berarti melakukan sesuatu dengan benar. Sesuatu yang efektif belum tentu efisien. Begitu juga sebaliknya, sesuatu yang efisien belum tentu efektif. Suatu penyelesaian sebaiknya mampu mencapai efektivitas dan efisiensi yang optimum untuk keduanya.

Dengan demikian suatu metode numerik dikatakan efisien bila waktu yang digunakan oleh algoritma *programming* suatu metode untuk mencapai batas toleransi (*error*) yang telah ditentukan lebih sedikit. Efektif bila solusi yang diberikan memiliki *error* yang lebih kecil atau solusi yang diperoleh lebih akurat dengan batas iterasi yang telah ditentukan. Waktu dan *error* suatu algoritma berkaitan dengan iterasi.

2.8 Jumlah Iterasi

Penelitian yang dilakukan menggunakan teknik komputerisasi yang membahas mengenai suatu programming sehingga indikator yang diambil adalah jumlah iterasi dan kecepatan CPU komputer dengan pengertiannya sebagai berikut.

Pengertian iterasi berkaitan dengan proses perhitungan berulang dalam komputer untuk mengevaluasi kesalahan. Misal domain masalah yang akan diselesaikan adalah $a \leq x \leq b$ maka teknik numerik dilakukan dengan membagi domain itu ke dalam n bagian (grid) dengan jarak antara bagian yang satu dan yang lain h satuan, sehingga kalkulasi diproses berdasarkan langkah tahap $x_i = a + ih$

untuk $i = 1, 2, \dots, n$. Jumlah iterasi secara kongkrit adalah tergantung pada sejauh mana i melangkah. Bila proses perhitungan mencapai $i = k$ maka komputer dikatakan melakukan k iterasi.

2.9 Algoritma dan Pemrograman

Algoritma adalah urutan langkah logis penyelesaian masalah yang disusun secara sistematis dan logis. Algoritma adalah jantung ilmu komputer atau informatika. Banyak cabang ilmu komputer yang diacu dalam terminologi algoritma. Peran komputer disini hanyalah salah satu pemroses. Agar dapat dilaksanakan oleh komputer, algoritma harus ditulis dalam notasi bahasa pemrograman sehingga dinamakan program. Jadi program adalah perwujudan atau implementasi teknis algoritma yang ditulis dalam bahasa pemrograman tertentu sehingga dapat dilaksanakan oleh komputer.

Untuk melihat faktor efisiensi dan efektifitas dari algoritma, dapat dilakukan analisis terhadap suatu algoritma dengan melihat pada:

1. Waktu tempuh (*Running Time*) dari suatu algoritma, yaitu satuan waktu yang ditempuh atau diperlukan oleh suatu algoritma dalam menyelesaikan suatu masalah. Hal-hal yang dapat mempengaruhi waktu tempuh adalah:
 - Banyaknya langkah. Makin banyak langkah atau instruksi yang digunakan dalam menyelesaikan masalah, maka makin lama waktu tempuh yang dibutuhkan dalam proses tersebut.
 - Besar dan jenis input data. Besar dan jenis input data pada suatu algoritma akan sangat berpengaruh pada proses perhitungan yang terjadi. Jika jenis data adalah tingkat ketelitian tunggal (*single precision*), maka waktu tempuh akan menjadi relatif lebih cepat dibandingkan dengan tingkat ketelitian ganda (*double precision*).
 - Jenis operasi. Waktu tempuh juga dipengaruhi oleh jenis operasi yang digunakan. Jenis operasi tersebut meliputi operasi matematika, nalar atau logika, atau yang lainnya. Sebagai contoh, operasi perkalian atau

pembagian akan memakan waktu lebih lama dibandingkan operasi penjumlahan atau pengurangan.

- Komputer dan kompilator. Hal terakhir yang mempengaruhi waktu tempuh suatu proses algoritma adalah komputer dan kompilatornya, walaupun sebenarnya faktor ini diluar tahap rancangan atau tahap pembuatan algoritma yang efisien. Algoritma dibuat untuk mencapai waktu tempuh yang seefektif dan seefisien mungkin, tetapi kesemuanya itu akan sangat bergantung pada kemampuan komputer yang tentunya harus sesuai dengan jumlah program atau langkah yang diperlukan oleh algoritma, begitu juga dengan kompilator tersebut.

2. Jumlah memori yang digunakan. Banyaknya langkah yang digunakan dan jenis variabel data yang dipakai dalam suatu algoritma akan sangat mempengaruhi penggunaan memori. Dalam hal ini, pengguna diharapkan dapat memperkirakan seberapa banyak kebutuhan memori yang diperlukan selama proses berlangsung hingga proses selesai dikerjakan. Dengan demikian, dapat disiapkan *storage* yang memadai agar proses suatu algoritma berjalan tanpa ada hambatan atau kekurangan memori.

Beberapa simbol dan kata-kata yang dipakai dalam algoritma, misalnya simbol periode (.) untuk menunjukkan akhir prosedur dan simbol titik koma (;) untuk memisahkan tugas dalam beberapa langkah. Adapun kata-kata yang dipakai adalah INPUT, OUTPUT, Set, Do dan lain-lain. Selain itu juga dikenal teknik loops (pengulangan), yang dinyatakan dengan "kontrol penyanggah",

For $i = 1, 2, \dots, n$

Set $x_i = a_i + i.h$

dan "kontrol bersyarat",

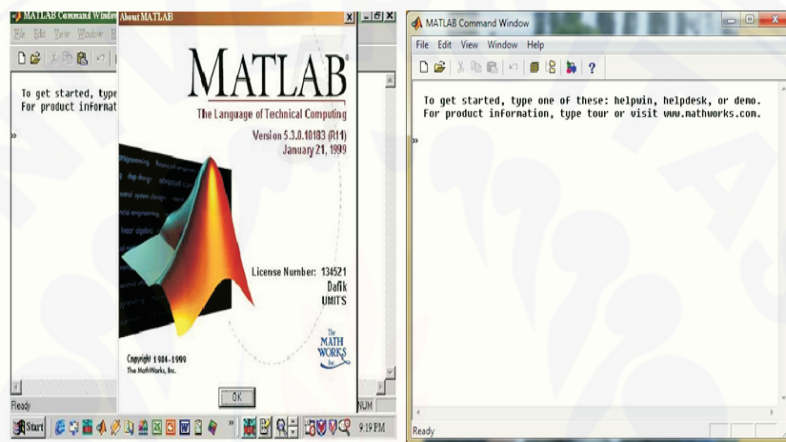
While $i < N$ do Step 3 - 6

IfThen,

IfThen.....Else

2.10 MATLAB Programming

MATLAB (*Matrix Laboratory*) adalah *software* aplikasi yang dilengkapi oleh fungsi-fungsi khusus sedemikian hingga mudah dan cepat menyelesaikan beberapa masalah terutama dalam masalah sains dan teknologi (Dafik,1999: 1). Software ini juga dilengkapi piranti program nonprosedural yang memberikan keleluasaan dan kemudahan bagi programmer untuk menyelesaikan dan mengembangkan suatu masalah. Prosedur untuk menjalankan fungsi-fungsi tersebut ditulis langsung pada lembar kerja MATLAB.



Gambar 2.2: Tampilan Awal dan Lembar Kerja MATLAB

Program MATLAB dapat ditulis dengan menggunakan perintah yang sangat sederhana, namun dapat mencakup tuntutan untuk menyelesaikan persoalan menganalisis data. Sekarang ini MATLAB adalah salah satu bahasa pemrograman yang banyak digunakan. MATLAB mampu menangani perhitungan sederhana seperti penambahan, pengurangan, perkalian dan pembagian. MATLAB juga mampu menyelesaikan perhitungan rumit yang meliputi bilangan kompleks, akar dan pangkat, logaritma dan fungsi trigonometri. Seperti kalkulator yang dapat diprogram, MATLAB dapat digunakan untuk menyimpan dan mengambil data, dalam MATLAB dapat dibuat sekumpulan perintah untuk mengotomatisasi suatu persamaan yang rumit, dan masih banyak lagi kemampuan lain dari MATLAB.

Fasilitas lain dari MATLAB adalah user dapat menggunakan MATLAB *pro-*

programming editor untuk menyusun prosedur-prosedur logis dalam program non-prosedural yang dapat dipahami langsung oleh MATLAB. Berhubung bahasa MATLAB adalah bahasa nonprosedural maka struktur bahasa yang dikembangkan tidak terlampaui hierarkikal dan banyak mengaitkan fungsi-fungsi yang sudah *build-in* dalam MATLAB *library*. Sehingga dalam hal ini MATLAB memberikan fleksibilitas yang luas terhadap para *user* untuk mengembangkan imajinasinya dalam penyelesaian masalah.

Selain itu, MATLAB juga dapat melakukan program besar sebagaimana *compiler* (bahasa pemrograman prosedural) lainnya. Dengan kemampuan ini user dapat mengembangkan suatu algoritma kemudian diimplementasikan dalam MATLAB *programming* untuk memecahkan masalah tertentu.



BAB 3. METODE PENELITIAN

3.1 Rancangan Penelitian

Rancangan penelitian merupakan suatu bentuk rencana kegiatan penelitian yang dibuat peneliti untuk memudahkan pelaksanaan penelitian. Rancangan penelitian adalah perencanaan untuk mengadakan penelitian yang didalamnya mencakup penjelasan secara rinci mengenai penelitian yang memuat penjelasan yang dibutuhkan dalam upaya memperoleh informasi serta mengolahnya dalam memecahkan masalah. Di dalam penelitian ini dibutuhkan prosedur penelitian yang merupakan suatu tahapan yang dilakukan sampai diperoleh data-data untuk di analisis hingga dicapai suatu kesimpulan yang sesuai dengan tujuan penelitian. Adapun prosedurnya adalah sebagai berikut:

1. menentukan sifat metode Runge-Kutta orde sepuluh berdasarkan persamaan 2.16, persamaan 2.17, dan persamaan 2.18;
2. membuat penurunan formula metode Runge-Kutta orde sepuluh dengan menggunakan sifat metode Runge-Kutta orde sepuluh;
3. menentukan konvergensi metode Runge-Kutta orde sepuluh secara teoritis dengan menggunakan definisi 2.6.5 dan definisi 2.6.6;
4. (tahap pemodelan) menggunakan model penyebaran virus Avian Influenza yang berbentuk sistem PDB yang ditulis oleh *Okosun*;
5. (tahap formulasi numerik) memformulasikan model secara numerik dengan pengubahan simbol dari model penyebaran virus Avian Influenza sehingga simbol yang baru dapat ditulis dan dibaca oleh MATLAB;
6. (tahap algoritma) membuat pola algoritma metode RK10;
7. (tahap pemrograman) membuat *programming* metode RK10;

8. (tahap operasional) melakukan eksekusi *programming* yang telah dibuat dengan menentukan nilai awal dan parameter berdasarkan jurnal yang ditulis oleh *Okosun*;
9. mengumpulkan data (error, iterasi, waktu tempuh, dan grafik) untuk menentukan efektivitas dan efisiensi metode RK10;
10. (tahap evaluasi) menganalisis data yang diperoleh dari data *ouput* dari hasil eksekusi metode RK10;
11. membuat kesimpulan berdasarkan hasil analisis.

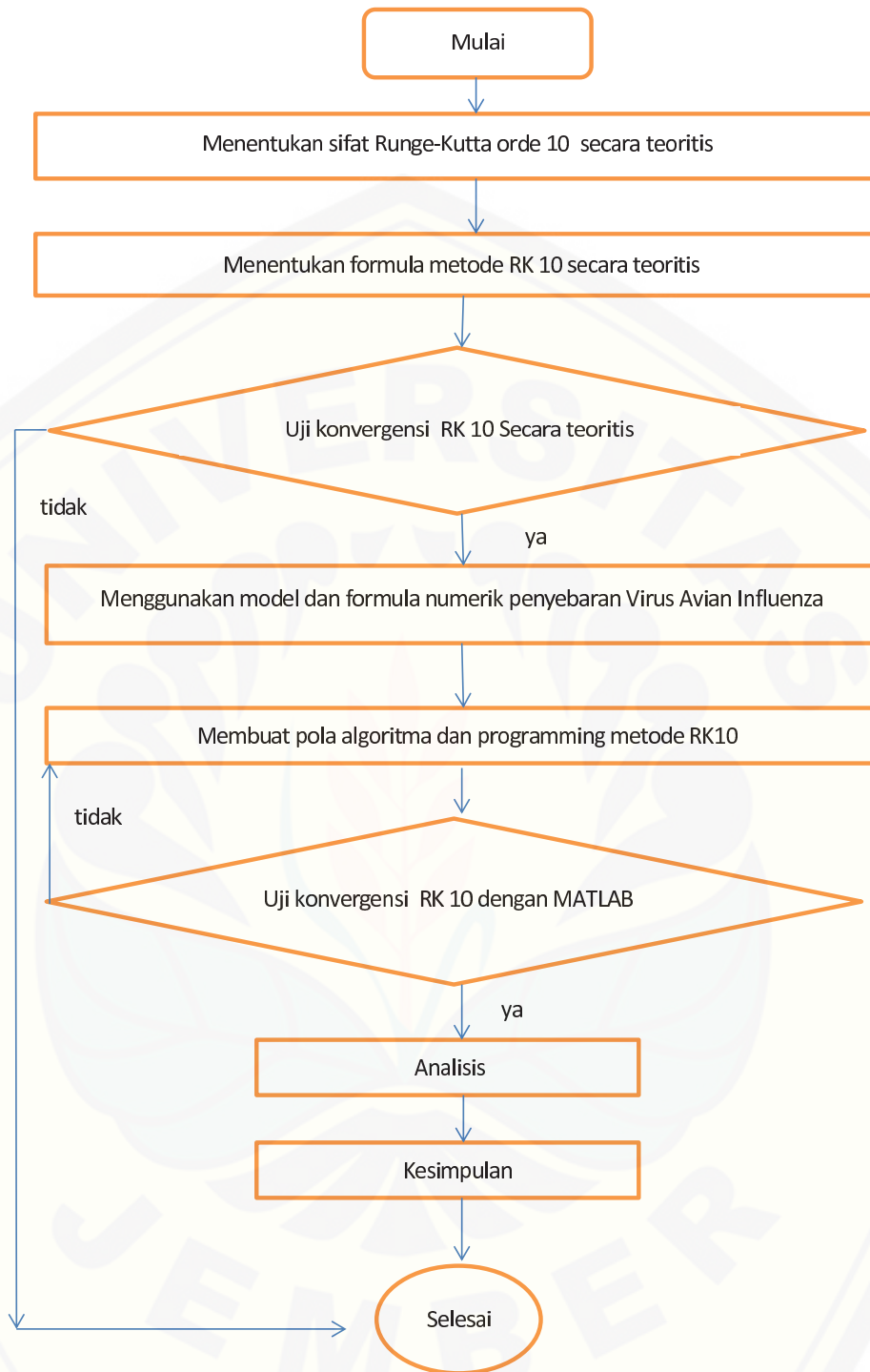
prosedur penelitian yang akan dilakukan sebagaimana disajikan pada Gambar 3.1.

3.2 Definisi Operasional

Metode Runge-Kutta adalah salah satu metode numerik yang dapat digunakan untuk mencari solusi persamaan diferensial. Metode ini memiliki acuan perhitungan fungsi pada titik dengan tepat. Metode Runge-Kutta yang dimaksud dalam penelitian ini adalah metode Runge-Kutta orde sepuluh yang telah diformulasikan oleh peneliti. Selanjutnya metode Runge-Kutta orde sepuluh ini digunakan untuk mencari solusi model persamaan diferensial biasa pada penyebaran virus Avian Influenza.

Avian Influenza atau flu burung adalah penyakit menular yang disebabkan virus Avian Influenza A dengan subtipe H1 sampai H16 dan N1 sampai N9. Virus ini menyerang berbagai jenis unggas, meliputi ayam, kalkun, unggas air, burung peliharaan dan burung liar. Terakhir virus ini dapat menginfeksi babi, harimau, kucing dan macan tutul. Pada umumnya virus avian influenza A tidak menyerang manusia, tetapi subtipe tertentu seperti H5N1, H7N7 yang bersifat sangat patogen dapat menyerang manusia dan mengakibatkan kematian.

Software yang digunakan dalam penelitian ini adalah MATLAB, dimana MATLAB merupakan sebuah aplikasi yang mampu menyelesaikan perhitungan yang rumit dalam menyelesaikan permasalahan diferensial. Solusi yang diberikan



Gambar 3.1: Prosedur Penelitian

oleh MATLAB berupa data mentah dan grafik maupun gambar yang selanjutnya akan digunakan oleh peneliti untuk menganalisis permasalahan dari penelitian ini.

3.3 Tempat Penelitian

Penelitian ini dilakukan di Laboratorium PSSI Universitas Jember dengan alasan di Labotatorium PSSI tersedia program MATLAB R2011b yang *original*, komputer yang berkecepatan tinggi dengan menggunakan prosesor Intel Xeon dan referensi yang dapat digunakan sebagai penunjang dalam penelitian ini.

3.4 Metode Pengumpulan Data

Metode pengumpulan data dalam kegiatan penelitian adalah suatu cara atau metode untuk memperoleh data. Pengumpulan data erat hubungannya dengan pemasalahan yang akan diselesaikan, sehingga metode yang digunakan dalam pengumpulan data berfungsi untuk mendukung penelitian dalam memperoleh data sesuai dengan tujuan penelitian.

Dalam penelitian ini, peneliti menggunakan metode dokumentasi dan metode eksperimen dalam proses pengumpulan data. Metode dokumentasi diartikan sebagai teknik pengumpulan data dengan menghimpun dan menganalisa dokumen-dokumen, baik dokumen tertulis, gambar, hasil karya, maupun elektronik. Dalam penelitian ini peneliti menggunakan jurnal yang ditulis oleh Okosun yaitu tentang model penyebaran virus Avian Influenza dan berbagai sumber lainnya yang relevan dengan penelitian ini.

Metode eksperimen adalah metode meneliti atau menyelidiki guna mencari jawaban atau pemecahan dari suatu kasus ataupun obyek penelitian. Pengumpulan data dengan metode eksperimen merupakan pengumpulan data yang dilakukan dengan cara mengadakan pengamatan dan pencatatan yang dilakukan secara sistematis terhadap indikator penelitian sesuai dengan perlakuan yang diberikan dan gejala-gejala apa yang akan terjadi. Hal-hal yang akan diamati dalam penelitian ini adalah error, jumlah iterasi, waktu tempuh dan grafik.

3.5 Analisis Data

Analisis data merupakan langkah terakhir dalam penelitian yang merupakan cara atau teknik untuk menyusun dan mengolah data yang terkumpul dalam penelitian agar dapat menghasilkan suatu kesimpulan yang dapat dipertanggungjawabkan. Data mentah yang diperoleh sebelumnya tidak akan berarti apa-apa jika tidak dianalisis. Dengan menganalisa data mentah yang telah dikumpulkan oleh peneliti dapat memberi arti dan makna sehingga dapat berguna untuk memecahkan masalah dan menguji hipotesis. Analisis data diarahkan untuk memberi argumentasi atau penjelasan mengenai tujuan yang diajukan dalam penelitian berdasarkan fakta yang diperoleh.

Dalam penelitian ini, data akan dianalisis secara numerik dimulai dari tahap pemasukan data (*input*), pengolahan data dan hasil (*output*). *Input* berupa tetapan parameter-parameter pada Tabel (2.1), juga nilai awal dari model penyebaran virus Avian Influenza. Data yang diperoleh tersebut dari *programming* metode Runge-Kutta orde sepuluh menghasilkan data berupa eror pada jumlah iterasi yang ditentukan, waktu tempuh eksekusi MATLAB sampai batas toleransi yang ditentukan, jumlah iterasi dan grafik dari populasinya.

Untuk konvergenitas dari metode tersebut dapat dilihat dari semakin kecilnya kesalahan yang ditimbulkan pada setiap iterasi atau mendekati nol, sehingga untuk

$$h \rightarrow 0 \quad \max_{0 \leq i \leq n} \|y(x_i) - y_i\| \rightarrow 0$$

Semakin kecil kesalahan yang dibuat metode untuk memberikan suatu solusi berarti metode tersebut semakin efektif. Dan sebaliknya, Semakin besar kesalahan yang dibuat metode untuk memberikan suatu solusi berarti metode tersebut dapat dikatakan kurang efektif.

BAB 4. HASIL DAN PEMBAHASAN

Bagian ini akan menyajikan hasil yang merupakan jawaban dari rumusan permasalahan. Prosedur dalam penelitian ini dimulai dari penurunan formula metode Runge-Kutta orde sepuluh dan pembuktian konvergenitas secara teoritis, penyusunan model penyebaran virus Avian Influenza, penyusunan pola algoritma dalam MATLAB. Hasil dari eksekusi program tersebut akan memuat data mengenai jumlah iterasi dan waktu yang dibutuhkan untuk mencapai konvergensi, serta data eror untuk menganalisis efektifitasnya. Dengan data tersebut dapat diketahui tingkat efektifitas metode Runge-Kutta orde sepuluh untuk menyelesaikan model penyebaran virus Avian Influenza.

4.1 Metode Runge-Kutta Orde Sepuluh

Formulasi metode Runge-Kutta orde sepuluh memiliki sepuluh tahap yaitu $k_1, k_2, k_3, \dots, k_{10}$ dan terdiri dari koefisien matrik c, b , dan a . Asumsi sebelumnya menyebutkan bahwa sifat metode Runge-Kutta secara umum adalah $\sum_{i=1}^m b_i = 1$ dan $c_i = \sum_{j=1}^m a_{ij}$. Tetapi dalam proses penurunan rumus memerlukan sifat lain dari metode Runge-Kutta untuk menentukan formulasi matrik pada formula yaitu $c_1, c_2, \dots, c_{10}, b_1, b_2, \dots, b_{10}$, dan $a_{21}, a_{31}, a_{32}, \dots, a_{109}$. Maka langkah pertama yang dilakukan adalah menentukan sifat baru dari metode Runge-Kutta secara umum pada pembuktian lema (4.1.1) berikut ini.

Lema 4.1.1. Metode Runge-Kutta orde sepuluh mempunyai sifat sebagai berikut:

$$\sum_{i=1}^m b_i = 1 \quad \text{dimana } m = 10$$

$$\sum_{i=2}^m b_i c_i^p = \frac{1}{p+1}, \quad \text{dimana } p = 1, 2, 3, 4, \dots, m-1$$

$$\sum_{i=3}^{m-1} b_i \left(\sum_{j=2}^{i-1} c_j^q a_{ij} \right) = \frac{1}{(q+1)(q+2)}, \quad \text{dimana } q = 1, 3, 4, \dots, m-3$$

Bukti

Dari persamaan (2.16), (2.17) dan (2.18), metode Runge-Kutta orde sepuluh dapat didefinisikan dengan:

$$\begin{aligned} y_{n+1} &= y_n + h \sum_{i=1}^m b_i k_i \quad \text{dimana } m = 10 \\ &= y_n + h(b_1 k_1 + b_2 k_2 + b_3 k_3 + b_4 k_4 + b_5 k_5 + b_6 k_6 + b_7 k_7 + \\ &\quad b_8 k_8 + b_9 k_9 + b_{10} k_{10}) \end{aligned} \tag{4.1}$$

dimana,

$$\begin{aligned} k_1 &= f(x_n, y_n) \\ k_2 &= f(x_n + c_2 h, y_n + h a_{21} k_1) \\ k_3 &= f(x_n + c_3 h, y_n + h(a_{31} k_1 + a_{32} k_2)) \\ k_4 &= f(x_n + c_4 h, y_n + h(a_{41} k_1 + a_{42} k_2 + a_{43} k_3)) \\ k_5 &= f(x_n + c_5 h, y_n + h(a_{51} k_1 + a_{52} k_2 + a_{53} k_3 + a_{54} k_4)) \\ k_6 &= f(x_n + c_6 h, y_n + h(a_{61} k_1 + a_{62} k_2 + a_{63} k_3 + a_{64} k_4 + a_{65} k_5)) \\ k_7 &= f(x_n + c_7 h, y_n + h(a_{71} k_1 + a_{72} k_2 + a_{73} k_3 + a_{74} k_4 + a_{75} k_5 + a_{76} k_6)) \\ k_8 &= f(x_n + c_8 h, y_n + h(a_{81} k_1 + a_{82} k_2 + a_{83} k_3 + a_{84} k_4 + a_{85} k_5 + a_{86} k_6 + a_{87} k_7)) \end{aligned}$$

$$\begin{aligned}
 k_9 &= f(x_n + c_9h, y_n + h(a_{91}k_1 + a_{92}k_2 + a_{83}k_3 + a_{94}k_4 + a_{95}k_5 + a_{96}k_6 + a_{97}k_7 \\
 &\quad + a_{98}k_8)) \\
 k_{10} &= f(x_n + c_{10}h, y_n + h(a_{101}k_1 + a_{102}k_2 + a_{103}k_3 + a_{104}k_4 + a_{105}k_5 + a_{106}k_6 + \\
 &\quad a_{107}k_7 + a_{108}k_8 + a_{109}k_9))
 \end{aligned}$$

Selanjutnya mengekspansi deret Taylor yang bertujuan untuk dibandingkan dengan persamaan (4.1) sehingga diperoleh koefisien matrik dari metode Runge-Kutta order sepuluh. Ekspansi deret Taylor tersebut adalah:

$$\begin{aligned}
 y_{(x_{n+1})} &= y_{(x_n)} + hy^{(1)}(x_n) + \frac{1}{2!}h^2y^{(2)}(x_n) + \frac{1}{3!}h^3y^{(3)}(x_n) + \\
 &\quad \frac{1}{4!}h^4y^{(4)}(x_n) + \frac{1}{5!}h^5y^{(5)}(x_n) + \frac{1}{6!}h^6y^{(6)}(x_n) + \frac{1}{7!}h^7y^{(7)}(x_n) + \\
 &\quad \frac{1}{8!}h^8y^{(8)}(x_n) + \frac{1}{9!}h^9y^{(9)}(x_n) + \frac{1}{10!}h^{10}y^{(10)}(x_n) + \frac{1}{11!}h^{11}y^{(11)}(x_n) + \dots
 \end{aligned} \tag{4.2}$$

Untuk melengkapinya dibutuhkan turunan $y^{(1)}(x_n), \dots, y^{(11)}(x_n)$ sedangkan $y^{(12)}(x_n)$ dan seterusnya merupakan sisa yang dihitung sebagai *error* kesalahan pemenggalan ($O(h^{12})$). Untuk menyederhanakan hasil turunan $y^{(1)}(x_n), \dots, y^{(11)}(x_n)$ maka dimisalkan,

$$\begin{aligned}
 J &= f_x + ff_y \\
 K &= f_{xx} + 2ff_{xy} + f^2f_{yy} \\
 L &= f_{xxx} + 3ff_{xxy} + 3f^2f_{xyy} + f^3f_{yyy} \\
 M &= f_{xxxx} + 4ff_{xxx} + 6f^2f_{xxy} + 4f^3f_{xyy} + f^4f_{yyy} \\
 N &= f_{xxxxx} + 5ff_{xxxx} + 10f^2f_{xxx} + 10f^3f_{xxy} + 5f^4f_{xyy} + f^5f_{yyy} \\
 O &= f_{xxxxx} + 6ff_{xxxx} + 15f^2f_{xxx} + 20f^3f_{xxy} + 15f^4f_{xyy} + 6f^5f_{yyy} + \\
 &\quad f^6f_{yyy} \\
 P &= f_{xxxxxx} + 7ff_{xxxxx} + 21f^2f_{xxxx} + 35f^3f_{xxx} + 35f^4f_{xxy} + \\
 &\quad 21f^5f_{xyy} + 7f^6f_{yyy} + f^7f_{yyy}
 \end{aligned}$$

$$\begin{aligned}
 Q &= f_{xxxxxxxx} + 8ff_{xxxxxxxxy} + 28f^2f_{xxxxxxxxyy} + 56f^3f_{xxxxxxxxyyy} + 70f^4f_{xxxxxxxxyyyy} + \\
 &\quad 56f^5f_{xxxxxxxxyyyyy} + 28f^6f_{xxxxxxxxyyyyyy} + 8f^7f_{xxxxxxxxyyyyyy} + f^8f_{xxxxxxxxyyyyyy} \\
 R &= f_{xxxxxxxx} + 9ff_{xxxxxxxxxy} + 36f^2f_{xxxxxxxxxyy} + 84f^3f_{xxxxxxxxxyyy} + 126f^4f_{xxxxxxxxxyyyy} + \\
 &\quad 126f^5f_{xxxxxxxxxyyyy} + 84f^6f_{xxxxxxxxxyyyy} + 36f^7f_{xxxxxxxxxyyyy} + 9f^8f_{xxxxxxxxxyyyy} + f^9f_{xxxxxxxxxyyyy}
 \end{aligned}$$

dengan aturan rantai didapatkan turunan pertama $y(x_n)$ hingga turunan kesepuluh yaitu:

$$\begin{aligned}
 y^{(1)} &= f(x, y) = f \\
 y^{(2)} &= f_x + f_y f \\
 &= J \\
 y^{(3)} &= f_{xx} + f_{xy} f + f_{yx} f + f_{yy} f f + f_y (f_x + f_y f) \\
 &= f_{xx} + 2f f_{xy} + f^2 f_{yy} + f_y (f_x + f_y f) \\
 &= K + J f_y \\
 y^{(4)} &= f_{xxx} + f_{xxy} f + 2((f_x + f_y f) f_{xy} + f(f_{xyx} + f_{xyy} f)) + 2f(f_x + f_y f) f_{yy} + \\
 &\quad f^2(f_{yyx} + f_{yyy} f) + (f_{yx} + f_{yy} f) f_x + f_y(f_{xx} + f_{xy} f) + 2f_y(f_{yx} + f_{yy} f) + \\
 &\quad f_y^2(f_x + f_y f) \\
 &= f_{xxx} + 3f f_{xxy} + 3f^2 f_{xyy} + f^3 f_{yyy} + f_y(f_{xx} + 2f f_{xy} + f^2 f_{yy}) + 3f_x f_{xy} + \\
 &\quad 3f f_x f_{yy} + 3f f_y f_{yy} + 2f_y f_{xy} + f_x f_y^2 + f f_y^3 \\
 &= L + K f_y
 \end{aligned}$$

Suku $3f_x f_{xy} + \dots + f f_y^3$ pada $y^{(4)}$ dianggap sebagai sisa yaitu kesalahan pemenggalan sehingga dapat diabaikan. Maka hasil $y^{(4)} = L + K f_y$, selanjutnya hasil penurunan $y^{(5)}, \dots, y^{(10)}$ analog dengan $y^{(4)}$

$$\begin{aligned}
 y_{(5)} &= f_{xxxx} + 4f f_{xxxxy} + 6f^2 f_{xxyy} + 4f^3 f_{xyyy} + f^4 f_{yyyy} + f_y(f_{xxx} + 3f f_{xxy} \\
 &\quad + 3f^2 f_{xyy} + f^3 f_{yyy}) + \dots + f f y^4 \\
 &= M + L f_y
 \end{aligned}$$

$$\begin{aligned}
 y_{(6)} &= f_{xxxxx} + 5f f_{xxxxxy} + 10f^2 f_{xxxxyy} + 10f^3 f_{xxxyyy} + 5f^4 f_{xyyyyy} + f^5 f_{yyyyyy} \\
 &\quad + f_y(f_{xxxx} + 4f f_{xxxxy} + 6f^2 f_{xxyy} + 4f^3 f_{xyyy} + f^4 f_{yyyy}) + \dots + f f y^5 \\
 &= N + M f_y
 \end{aligned}$$

$$\begin{aligned}
 y_{(7)} &= f_{xxxxxx} + 6f f_{xxxxxy} + 15f^2 f_{xxxxyy} + 20f^3 f_{xxxyyy} + 15f^4 f_{xyyyyy} + 6f^5 f_{xyyyyy} \\
 &\quad + f^6 f_{yyyyyy} + f_y(f_{xxxxx} + 5f f_{xxxxxy} + 10f^2 f_{xxxxyy} + 10f^3 f_{xxxyyy} + 5f^4 f_{xyyyyy} \\
 &\quad + f^5 f_{xyyyyy}) + \dots + f f y^6 \\
 &= O + N f_y
 \end{aligned}$$

$$\begin{aligned}
 y_{(8)} &= f_{xxxxxxx} + 7f f_{xxxxxxy} + 21f^2 f_{xxxxyy} + 35f^3 f_{xxxyyy} + 35f^4 f_{xyyyyy} + \\
 &\quad 21f^5 f_{xyyyyy} + 7f^6 f_{xyyyyy} + f^7 f_{yyyyyy} + f_y(f_{xxxxxx} + 6f f_{xxxxxy} + \\
 &\quad 15f^2 f_{xxxxyy} + 20f^3 f_{xxxyyy} + 15f^4 f_{xyyyyy} + 6f^5 f_{xyyyyy} + f^6 f_{xyyyyy}) + \dots + f f y^7 \\
 &= P + O f_y
 \end{aligned}$$

$$\begin{aligned}
 y_{(9)} &= f_{xxxxxxxx} + 8f f_{xxxxxxy} + 28f^2 f_{xxxxyy} + 56f^3 f_{xxxyyy} + 70f^4 f_{xyyyyy} + \\
 &\quad 56f^5 f_{xyyyyy} + 28f^6 f_{xyyyyy} + 8f^7 f_{xyyyyy} + f^8 f_{yyyyyy} + \dots + f f y^8 \\
 &= Q + P f_y
 \end{aligned}$$

$$\begin{aligned}
 y_{(10)} &= f_{xxxxxxxxx} + 9f f_{xxxxxxy} + 36f^2 f_{xxxxyy} + 84f^3 f_{xxxyyy} + \\
 &\quad 126f^4 f_{xyyyyy} + 126f^5 f_{xyyyyy} + 84f^6 f_{xyyyyy} + 36f^7 f_{xyyyyy} + \\
 &\quad 9f^8 f_{xyyyyy} + f^9 f_{xyyyyy} + \dots + f f y^8 \\
 &= R + Q f_y
 \end{aligned}$$

Sehingga hasil ekspansi deret Taylor (Persamaan 4.2) dengan mensubstitusikan

$y^{(1)}, y^{(2)}, \dots, y^{(10)}$ adalah:

$$\begin{aligned}
 y_{(x_{n+1})} = & y_{(x_n)} + hf + \frac{1}{2!}h^2J + \frac{1}{3!}h^3(K + Jf_y) + \frac{1}{4!}h^4(L + Kf_y) + & (4.3) \\
 & \frac{1}{5!}h^5(M + Lf_y) + \frac{1}{6!}h^6(N + Mf_y) + \frac{1}{7!}h^7(O + Nf_y) + \frac{1}{8!}h^8(P + Of_y) \\
 & \frac{1}{9!}h^9(Q + Pf_y) + \frac{1}{10!}h^{10}(R + Qf_y) + \frac{1}{11!}h^{11}y^{(11)}(x_n) + \dots
 \end{aligned}$$

Konstanta-konstanta dalam $k_1, k_2, k_3, k_4, k_5, \dots, k_{10}$ harus ditentukan sedemikian rupa sehingga persamaan (4.1) akan sama dengan Deret Taylor (Persamaan 4.3). Maka $k_1, k_2, k_3, k_4, k_5, \dots, k_{10}$ diekspansi menggunakan aturan ekspansi deret Taylor dua variabel sebagai berikut.

$$k_1 = f(x_n, y_n) = f$$

$$\begin{aligned}
 k_2 = & f(x_n + c_2h, y_n + ha_{21}k_1) \\
 = & f + c_2hJ + \frac{1}{2}c_2^2h^2K + \frac{1}{6}c_2^3h^3L + \frac{1}{24}c_2^4h^4M + \frac{1}{5!}c_2^5h^5N + \frac{1}{6!}c_2^6h^6O + \\
 & \frac{1}{7!}c_2^7h^7P + \frac{1}{8!}c_2^8h^8Q + \frac{1}{9!}c_2^9h^9R + \dots
 \end{aligned}$$

$$\begin{aligned}
 k_3 = & f(x_n + c_3h, y_n + h(a_{31}k_1 + a_{32}k_2)) \\
 = & f + c_3hJ + h^2[c_2a_{32}Jf_y + \frac{1}{2}c_3^2K] + h^3[\frac{1}{2}c_2^2a_{32}Kf_y + \frac{1}{6}c_3^3L] + h^4[\frac{1}{6}c_2^3a_{32}Lf_y + \\
 & \frac{1}{24}c_3^4M] + h^5[\frac{1}{24}c_2^4a_{32}Mf_y + \frac{1}{5!}c_3^5N] + h^6[\frac{1}{5!}c_2^5a_{32}Nf_y + \frac{1}{6!}c_3^6O] + \\
 & h^7[\frac{1}{6!}c_2^6a_{32}Of_y + \frac{1}{7!}c_3^7P] + h^8[\frac{1}{7!}c_2^7a_{32}Pf_y + \frac{1}{8!}c_3^8Q] + h^9[\frac{1}{9!}c_3^9R] + \dots
 \end{aligned}$$

$$\begin{aligned}
 k_4 = & f(x_n + c_4h, y_n + h(a_{41}k_1 + a_{42}k_2 + a_{43}k_3)) \\
 = & f + c_4hJ + h^2[(a_{42}c_2 + a_{43}c_3)Jf_y + \frac{1}{2}c_4^2K] + h^3[\frac{1}{2}(a_{42}c_2^2 + a_{43}c_3^2)Kf_y + \frac{1}{6}c_4^3L] \\
 & + h^4[\frac{1}{6}(a_{42}c_2^3 + a_{43}c_3^3)Lf_y + \frac{1}{24}c_4^4M] + h^5[\frac{1}{24}(a_{42}c_2^4 + a_{43}c_3^4)Mf_y + \frac{1}{5!}c_4^5N] + \\
 & h^6[\frac{1}{5!}(a_{42}c_2^5 + a_{43}c_3^5)Nf_y + \frac{1}{6!}c_4^6O] + h^7[\frac{1}{6!}(a_{42}c_2^6 + a_{43}c_3^6)Of_y + \frac{1}{7!}c_4^7P] + \\
 & + h^8[\frac{1}{7!}(c_2^7a_{42} + c_3^7a_{43})Pf_y + \frac{1}{8!}c_4^8Q] + h^9[\frac{1}{9!}c_4^9R] + \dots
 \end{aligned}$$

$$\begin{aligned}
 k_5 &= f(x_n + c_5h, y_n + h(a_{51}k_1 + a_{52}k_2 + a_{53}k_3 + a_{54}k_4)) \\
 &= f + c_5hJ + h^2[(\sum_{i=2}^4 c_i a_{5i})Jf_y + \frac{1}{2}c_5^2K] + h^3[\frac{1}{2}(\sum_{i=2}^4 c_i^2 a_{5i})Kf_y + \frac{1}{6}c_5^3L] + \\
 &\quad h^4[\frac{1}{6}(\sum_{i=2}^4 c_i^3 a_{5i})Lf_y + \frac{1}{24}c_5^4M] + h^5[\frac{1}{24}(\sum_{i=2}^4 c_i^4 a_{5i})Mf_y + \frac{1}{5!}c_5^5N] + \\
 &\quad h^6[\frac{1}{5!}(\sum_{i=2}^4 c_i^5 a_{5i})Nf_y + \frac{1}{6!}c_5^6O] + h^7[\frac{1}{6!}(\sum_{i=2}^4 c_i^6 a_{5i})Of_y + \frac{1}{7!}c_5^7P] + \\
 &\quad h^8[\frac{1}{7!}(\sum_{i=2}^4 c_i^7 a_{5i})Pf_y + \frac{1}{8!}c_5^8Q] + h^9[\frac{1}{9!}c_5^9R] + \dots \\
 k_6 &= f(x_n + c_6h, y_n + h(a_{61}k_1 + a_{62}k_2 + a_{63}k_3 + a_{64}k_4 + a_{65}k_5)) \\
 &= f + c_6hJ + h^2[(\sum_{i=2}^5 c_i a_{6i})Jf_y + \frac{1}{2}c_6^2K] + h^3[\frac{1}{2}(\sum_{i=2}^5 c_i^2 a_{6i})Kf_y + \frac{1}{6}c_6^3L] + \\
 &\quad h^4[\frac{1}{6}(\sum_{i=2}^5 c_i^3 a_{6i})Lf_y + \frac{1}{24}c_6^4M] + h^5[\frac{1}{24}(\sum_{i=2}^5 c_i^4 a_{6i})Mf_y + \frac{1}{5!}c_6^5N] + \\
 &\quad h^6[\frac{1}{5!}(\sum_{i=2}^5 c_i^5 a_{6i})Nf_y + \frac{1}{6!}c_6^6O] + h^7[\frac{1}{6!}(\sum_{i=2}^5 c_i^6 a_{6i})Of_y + \frac{1}{7!}c_6^7P] + \\
 &\quad h^8[\frac{1}{7!}(\sum_{i=2}^5 c_i^7 a_{6i})Pf_y + \frac{1}{8!}c_6^8Q] + h^9[\frac{1}{9!}c_6^9R] + \dots \\
 k_7 &= f(x_n + c_7h, y_n + h(\sum_{i=1}^6 a_{7i}k_i)) \\
 &= f + c_7hJ + h^2[(\sum_{i=2}^6 c_i a_{7i})Jf_y + \frac{1}{2}c_7^2K] + h^3[\frac{1}{2}(\sum_{i=2}^6 c_i^2 a_{7i})Kf_y + \frac{1}{6}c_7^3L] + \\
 &\quad h^4[\frac{1}{6}(\sum_{i=2}^6 c_i^3 a_{7i})Lf_y + \frac{1}{24}c_7^4M] + h^5[\frac{1}{24}(\sum_{i=2}^6 c_i^4 a_{7i})Mf_y + \frac{1}{5!}c_7^5N] + \\
 &\quad h^6[\frac{1}{5!}(\sum_{i=2}^6 c_i^5 a_{7i})Nf_y + \frac{1}{6!}c_7^6O] + h^7[\frac{1}{6!}(\sum_{i=2}^6 c_i^6 a_{7i})Of_y + \frac{1}{7!}c_7^7P] + \\
 &\quad h^8[\frac{1}{7!}(\sum_{i=2}^6 c_i^7 a_{7i})Pf_y + \frac{1}{8!}c_7^8Q] + h^9[\frac{1}{9!}c_7^9R] + \dots \\
 k_8 &= f(x_n + c_8h, y_n + h(\sum_{i=1}^7 a_{8i}k_i)) \\
 &= f + c_8hJ + h^2[(\sum_{i=2}^7 c_i a_{8i})Jf_y + \frac{1}{2}c_8^2K] + h^3[\frac{1}{2}(\sum_{i=2}^7 c_i^2 a_{8i})Kf_y + \frac{1}{6}c_8^3L] + \\
 &\quad h^4[\frac{1}{6}(\sum_{i=2}^7 c_i^3 a_{8i})Lf_y + \frac{1}{24}c_8^4M] + h^5[\frac{1}{24}(\sum_{i=2}^7 c_i^4 a_{8i})Mf_y + \frac{1}{5!}c_8^5N] + \\
 &\quad h^6[\frac{1}{5!}(\sum_{i=2}^7 c_i^5 a_{8i})Nf_y + \frac{1}{6!}c_8^6O] + h^7[\frac{1}{6!}(\sum_{i=2}^7 c_i^6 a_{8i})Of_y + \frac{1}{7!}c_8^7P] + \\
 &\quad h^8[\frac{1}{7!}(\sum_{i=2}^7 c_i^7 a_{8i})Pf_y + \frac{1}{8!}c_8^8Q] + h^9[\frac{1}{9!}c_8^9R] + \dots \\
 k_9 &= f(x_n + c_9h, y_n + h(\sum_{i=1}^8 a_{9i}k_i)) \\
 &= f + c_9hJ + h^2[(\sum_{i=2}^8 c_i a_{9i})Jf_y + \frac{1}{2}c_9^2K] + h^3[\frac{1}{2}(\sum_{i=2}^8 c_i^2 a_{9i})Kf_y + \frac{1}{6}c_9^3L] + \\
 &\quad h^4[\frac{1}{6}(\sum_{i=2}^8 c_i^3 a_{9i})Lf_y + \frac{1}{24}c_9^4M] + h^5[\frac{1}{24}(\sum_{i=2}^8 c_i^4 a_{9i})Mf_y + \frac{1}{5!}c_9^5N] + \\
 &\quad h^6[\frac{1}{5!}(\sum_{i=2}^8 c_i^5 a_{9i})Nf_y + \frac{1}{6!}c_9^6O] + h^7[\frac{1}{6!}(\sum_{i=2}^8 c_i^6 a_{9i})Of_y + \frac{1}{7!}c_9^7P] + \\
 &\quad h^8[\frac{1}{7!}(\sum_{i=2}^8 c_i^7 a_{9i})Pf_y + \frac{1}{8!}c_9^8Q] + h^9[\frac{1}{9!}c_9^9R] + \dots
 \end{aligned}$$

$$\begin{aligned}
 k_{10} &= f(x_n + c_{10}h, y_n + h(\sum_{i=1}^9 a_{10i}k_i)) \\
 &= f + hc_{10}J + h^2[(\sum_{i=2}^9 c_i a_{10i})Jf_y + \frac{1}{2!}c_{10}^2 K] + h^3[\frac{1}{2!}(\sum_{i=2}^9 c_i^2 a_{10i})Kf_y + \\
 &\quad \frac{1}{3!}c_{10}^3 L] + h^4[\frac{1}{3!}(\sum_{i=2}^9 c_i^3 a_{10i})Lf_y + \frac{1}{4!}c_{10}^4 M] + h^5[\frac{1}{4!}(\sum_{i=2}^9 c_i^4 a_{10i})Mf_y + \\
 &\quad \frac{1}{5!}c_{10}^5 N] + h^6[\frac{1}{5!}(\sum_{i=2}^9 c_i^5 a_{10i})Nf_y + \frac{1}{6!}c_{10}^6 O] + h^7[\frac{1}{6!}(\sum_{i=2}^9 c_i^6 a_{10i})Of_y + \\
 &\quad \frac{1}{7!}c_{10}^7 P] + h^8[\frac{1}{7!}(\sum_{i=2}^9 c_i^7 a_{10i})Pf_y + \frac{1}{8!}c_{10}^8 Q] + h^9[\frac{1}{9!}c_{10}^9 R] + \dots
 \end{aligned}$$

dengan menyubstitusikan persamaan $k_1, k_2, k_3, \dots, k_{10}$ ke persamaan (4.1) maka didapatkan,

$$\begin{aligned}
 y_{n+1} &= y_n + h(\sum_{i=1}^{10} b_i k_i) \\
 &= y_n + h[b_1 f + b_2(f + c_2 h J + \frac{1}{2}c_2^2 h^2 K + \frac{1}{6}c_2^3 h^3 L + \frac{1}{24}c_2^4 h^4 M + \frac{1}{5!}c_2^5 h^5 N \\
 &\quad + \frac{1}{6!}c_2^6 h^6 O + \frac{1}{7!}c_2^7 h^7 P + \frac{1}{8!}c_2^8 h^8 Q + \frac{1}{9!}c_2^9 h^9 R + \dots) + b_3(f + c_3 h J + \\
 &\quad h^2[c_2 a_{32} J f_y + \frac{1}{2}c_3^2 K] + h^3[\frac{1}{2}c_2^2 a_{32} K f_y + \frac{1}{6}c_3^3 L] + h^4[\frac{1}{6}c_2^3 a_{32} L f_y + \frac{1}{24}c_3^4 M] \\
 &\quad + h^5[\frac{1}{24}c_2^4 a_{32} M f_y + \frac{1}{5!}c_3^5 N] + h^6[\frac{1}{5!}c_2^5 a_{32} N f_y + \frac{1}{6!}c_3^6 O] + h^7[\frac{1}{6!}c_2^6 a_{32} O f_y + \\
 &\quad \frac{1}{7!}c_3^7 P] + h^8[\frac{1}{7!}c_2^7 a_{32} P f_y + \frac{1}{8!}c_3^8 Q] + h^9(\frac{1}{9!}c_3^9 R) + \dots) + b_4(f + c_4 h J + \\
 &\quad h^2[(a_{42}c_2 + a_{43}c_3)Jf_y + \frac{1}{2}c_4^2 K] + h^3[\frac{1}{2}(a_{42}c_2^2 + a_{43}c_3^2)Kf_y + \frac{1}{6}c_4^3 L] + \\
 &\quad h^4[\frac{1}{6}(a_{42}c_2^3 + a_{43}c_3^3)Lf_y + \frac{1}{24}c_4^4 M] + h^5[\frac{1}{24}(a_{42}c_2^4 + a_{43}c_3^4)Mf_y + \frac{1}{5!}c_4^5 N] + \\
 &\quad h^6[\frac{1}{5!}(a_{42}c_2^5 + a_{43}c_3^5)Nf_y + \frac{1}{6!}c_4^6 O] + h^7[\frac{1}{6!}(a_{42}c_2^6 + a_{43}c_3^6)Of_y + \frac{1}{7!}c_4^7 P] + \\
 &\quad h^8[\frac{1}{7!}(c_2^7 a_{42} + c_3^7 a_{43})Pf_y + \frac{1}{8!}c_4^8 Q] + h^9(\frac{1}{9!}c_4^9 R) + \dots) + b_5(f + c_5 h J + \\
 &\quad h^2[(\sum_{i=2}^4 a_{5i} c_i)Jf_y + \frac{1}{2}c_5^2 K] + h^3[\frac{1}{2}(\sum_{i=2}^4 a_{5i} c_i^2)Kf_y + \frac{1}{6}c_5^3 L] + h^4[\frac{1}{6}(\sum_{i=2}^4 a_{5i} c_i^3) \\
 &\quad Lf_y + \frac{1}{24}c_5^4 M] + h^5[\frac{1}{24}(\sum_{i=2}^4 a_{5i} c_i^4)Mf_y + \frac{1}{5!}c_5^5 N] + h^6[\frac{1}{5!}(\sum_{i=2}^4 a_{5i} c_i^5)Nf_y + \\
 &\quad \frac{1}{6!}c_5^6 O] + h^7[\frac{1}{6!}(\sum_{i=2}^4 a_{5i} c_i^6)Of_y + \frac{1}{7!}c_5^7 P] + h^8[\frac{1}{7!}(\sum_{i=2}^4 a_{5i} c_i^7)Pf_y + \frac{1}{8!}c_5^8 Q] + \\
 &\quad \frac{1}{9!}c_5^9 h^9 R + \dots) + b_6(f + c_6 h J + h^2[(\sum_{i=2}^5 a_{6i} c_i)Jf_y + \frac{1}{2}c_6^2 K] + h^3[\frac{1}{2}(\sum_{i=2}^5 a_{6i} c_i^2)
 \end{aligned}$$

$$\begin{aligned}
 & Kf_y + \frac{1}{6}c_6^3L] + h^4[\frac{1}{6}(\sum_{i=2}^5 a_{6i}c_i^3)Lf_y + \frac{1}{24}c_6^4M] + h^5[\frac{1}{24}(\sum_{i=2}^5 a_{6i}c_i^4)Mf_y + \\
 & \frac{1}{5!}c_6^5N] + h^6[\frac{1}{5!}(\sum_{i=2}^5 a_{6i}c_i^5)Nf_y + \frac{1}{6!}c_6^6O] + h^7[\frac{1}{6!}(\sum_{i=2}^5 a_{6i}c_i^6)Of_y + \frac{1}{7!}c_6^7P] + \\
 & h^8[\frac{1}{7!}(\sum_{i=2}^5 a_{6i}c_i^7)Pf_y + \frac{1}{8!}c_6^8Q] + \frac{1}{9!}c_6^9h^9R + \dots) + b_7(f + c_7hJ + h^2[(\sum_{i=2}^6 a_{7i}c_i) \\
 & Jf_y + \frac{1}{2}c_7^2K] + h^3[\frac{1}{2}(\sum_{i=2}^6 a_{7i}c_i^2)Kf_y + \frac{1}{6}c_7^3L] + h^4[\frac{1}{6}(\sum_{i=2}^6 a_{7i}c_i^3)Lf_y + \frac{1}{24}c_7^4M] \\
 & + h^5[\frac{1}{24}(\sum_{i=2}^6 a_{7i}c_i^4)Mf_y + \frac{1}{5!}c_7^5N] + h^6[\frac{1}{5!}(\sum_{i=2}^6 a_{7i}c_i^5)Nf_y + \frac{1}{6!}c_7^6O] + h^7[\frac{1}{6!} \\
 & (\sum_{i=2}^6 a_{7i}c_i^6)Of_y + \frac{1}{7!}c_7^7P] + h^8[\frac{1}{7!}(\sum_{i=2}^6 a_{7i}c_i^7)Pf_y + \frac{1}{8!}c_7^8Q] + \frac{1}{9!}c_7^9h^9R + \dots) + \\
 & b_8(f + c_8hJ + h^2[(\sum_{i=2}^7 a_{8i}c_i)Jf_y + \frac{1}{2}c_8^2K] + h^3[\frac{1}{2}(\sum_{i=2}^7 a_{8i}c_i^2)Kf_y + \frac{1}{6}c_8^3L] + \\
 & h^4[\frac{1}{6}(\sum_{i=2}^7 a_{8i}c_i^3)Lf_y + \frac{1}{24}c_8^4M] + h^5[\frac{1}{24}(\sum_{i=2}^7 a_{8i}c_i^4)Mf_y + \frac{1}{5!}c_8^5N] + h^6[\frac{1}{5!} \\
 & (\sum_{i=2}^7 a_{8i}c_i^5)Nf_y + \frac{1}{6!}c_8^6O] + h^7[\frac{1}{6!}(\sum_{i=2}^7 a_{8i}c_i^6)Of_y + \frac{1}{7!}c_8^7P] + h^8[\frac{1}{7!}(\sum_{i=2}^7 a_{8i}c_i^7) \\
 & Pf_y + \frac{1}{8!}c_8^8Q] + \frac{1}{9!}c_8^9h^9R + \dots) + b_9(f + c_9hJ + h^2[(\sum_{i=2}^8 a_{9i}c_i)Jf_y + \frac{1}{2}c_9^2K] \\
 & h^3[\frac{1}{2}(\sum_{i=2}^8 a_{9i}c_i^2)Kf_y + \frac{1}{6}c_9^3L] + h^4[\frac{1}{6}(\sum_{i=2}^8 a_{9i}c_i^3)Lf_y + \frac{1}{24}c_9^4M] + h^5[\frac{1}{24} \\
 & (\sum_{i=2}^8 a_{9i}c_i^4) + Mf_y + \frac{1}{5!}c_9^5N] + h^6[\frac{1}{5!}(\sum_{i=2}^8 a_{9i}c_i^5)Nf_y + \frac{1}{6!}c_9^6O] + h^7[\frac{1}{6!} \\
 & (\sum_{i=2}^8 a_{9i}c_i^6)Of_y + \frac{1}{7!}c_9^7P] + h^8[\frac{1}{7!}(\sum_{i=2}^8 a_{9i}c_i^7)Pf_y + \frac{1}{8!}c_9^8Q] + \frac{1}{9!}c_9^9h^9R + \dots) + \\
 & b_{10}(f + c_{10}hJ + h^2[(\sum_{i=2}^9 a_{10i}c_i)Jf_y + \frac{1}{2}c_{10}^2K] + h^3[\frac{1}{2}(\sum_{i=2}^9 a_{10i}c_i^2)Kf_y + \frac{1}{6}c_{10}^3L] \\
 & + h^4[\frac{1}{6}(\sum_{i=2}^9 a_{10i}c_i^3)Lf_y + \frac{1}{24}c_{10}^4M] + h^5[\frac{1}{24}(\sum_{i=2}^9 a_{10i}c_i^4) + Mf_y + \frac{1}{5!}c_{10}^5N] + \\
 & h^6[\frac{1}{5!}(\sum_{i=2}^9 a_{10i}c_i^5)Nf_y + \frac{1}{6!}c_{10}^6O] + h^7[\frac{1}{6!}(\sum_{i=2}^9 a_{10i}c_i^6)Of_y + \frac{1}{7!}c_{10}^7P] + h^8[\frac{1}{7!} \\
 & (\sum_{i=2}^9 a_{10i}c_i^7)Pf_y + \frac{1}{8!}c_{10}^8Q] + \frac{1}{9!}c_{10}^9h^9R + \dots)] \\
 = & y_n + hf(\sum_{i=1}^{10} b_i) + h^2(\sum_{i=2}^{10} c_i b_i)J + h^3(\frac{1}{2}(\sum_{i=2}^{10} c_i b_i)K + (b_3 a_{32} c_2 + b_4(\sum_{i=2}^3 a_{4i} c_i) \\
 & + b_5(\sum_{i=2}^4 a_{5i} c_i) + b_6(\sum_{i=2}^5 a_{6i} c_i) + b_7(\sum_{i=2}^6 a_{7i} c_i) + b_8(\sum_{i=2}^7 a_{8i} c_i) + b_9(\sum_{i=2}^8 a_{9i} c_i \\
 & + b_{10}(\sum_{i=2}^9 a_{10i} c_i))Jf_y) + h^4(\frac{1}{6}(\sum_{i=2}^{10} c_i^3 b_i)L + \frac{1}{2}(b_3 a_{32} c_2^2 + b_4(\sum_{i=2}^3 a_{4i} c_i^2) \\
 & + b_5(\sum_{i=2}^4 a_{5i} c_i^2) + b_6(\sum_{i=2}^5 a_{6i} c_i^2) + b_7(\sum_{i=2}^6 a_{7i} c_i^2) + b_8(\sum_{i=2}^7 a_{8i} c_i^2) + b_9(\sum_{i=2}^8 a_{9i} c_i^2) \\
 & + b_{10}(\sum_{i=2}^9 a_{10i} c_i^2))Kf_y) + h^5(\frac{1}{24}(\sum_{i=2}^{10} c_i^4 b_i)M + \frac{1}{6}(b_3 a_{32} c_2^3 + b_4(\sum_{i=2}^3 a_{4i} c_i^3) +
 \end{aligned}$$

$$\begin{aligned}
 & b_5(\sum_{i=2}^4 a_{5i} c_i^3) + b_6(\sum_{i=2}^5 a_{6i} c_i^3) + b_7(\sum_{i=2}^6 a_{7i} c_i^3) + b_8(\sum_{i=2}^7 a_{8i} c_i^3) + b_9(\sum_{i=2}^8 a_{9i} c_i^3) \\
 & + b_{10}(\sum_{i=2}^9 a_{10i} c_i^3) Lf_y + h^6 \left(\frac{1}{5!} (\sum_{i=2}^{10} c_i^5 b_i) N + \frac{1}{4!} (b_3 a_{32} c_2^4 + b_4 (\sum_{i=2}^3 a_{4i} c_i^4) + b_5 \right. \\
 & (\sum_{i=2}^4 a_{5i} c_i^4) + b_6 (\sum_{i=2}^5 a_{6i} c_i^4) + b_7 (\sum_{i=2}^6 a_{7i} c_i^4) + b_8 (\sum_{i=2}^7 a_{8i} c_i^4) + b_9 (\sum_{i=2}^8 a_{9i} c_i^4) + \\
 & b_{10} (\sum_{i=2}^9 a_{10i} c_i^4) Mf_y + h^7 \left(\frac{1}{6!} (\sum_{i=2}^{10} c_i^6 b_i) O + \frac{1}{5!} (b_3 a_{32} c_2^5 + b_4 (\sum_{i=2}^3 a_{4i} c_i^5) + b_5 \right. \\
 & (\sum_{i=2}^4 a_{5i} c_i^5) + b_6 (\sum_{i=2}^5 a_{6i} c_i^5) + b_7 (\sum_{i=2}^6 a_{7i} c_i^5) + b_8 (\sum_{i=2}^7 a_{8i} c_i^5) + b_9 (\sum_{i=2}^8 a_{9i} c_i^5) + \\
 & b_{10} (\sum_{i=2}^9 a_{10i} c_i^5) Nf_y + h^8 \left(\frac{1}{7!} (\sum_{i=2}^{10} c_i^7 b_i) P + \frac{1}{6!} (b_3 a_{32} c_2^6 + b_4 (\sum_{i=2}^3 a_{4i} c_i^6) + b_5 \right. \\
 & (\sum_{i=2}^4 a_{5i} c_i^6) + b_6 (\sum_{i=2}^5 a_{6i} c_i^6) + b_7 (\sum_{i=2}^6 a_{7i} c_i^6) + b_8 (\sum_{i=2}^7 a_{8i} c_i^6) + b_9 (\sum_{i=2}^8 a_{9i} c_i^6) + \\
 & b_{10} (\sum_{i=2}^9 a_{10i} c_i^6) Of_y + h^9 \left(\frac{1}{8!} (\sum_{i=2}^{10} c_i^8 b_i) Q + \frac{1}{7!} (b_3 a_{32} c_2^7 + b_4 (\sum_{i=2}^3 a_{4i} c_i^7) + b_5 \right. \\
 & (\sum_{i=2}^4 a_{5i} c_i^7) + b_6 (\sum_{i=2}^5 a_{6i} c_i^7) + b_7 (\sum_{i=2}^6 a_{7i} c_i^7) + b_8 (\sum_{i=2}^7 a_{8i} c_i^7) + b_9 (\sum_{i=2}^8 a_{9i} c_i^7) + \\
 & b_{10} (\sum_{i=2}^9 a_{10i} c_i^7) Pf_y + h^{10} \left(\frac{1}{9!} (\sum_{i=2}^{10} c_i^9 b_i) R \right. \tag{4.4}
 \end{aligned}$$

dengan membandingkan koefisien dari persamaan (4.4) dengan persamaan (4.3), didapat:

$$\begin{aligned}
 b_1 + b_2 + b_3 + b_4 + b_5 + b_6 + b_7 + b_8 + b_9 + b_{10} &= 1 \\
 c_2 b_2 + c_3 b_3 + c_4 b_4 + c_5 b_5 + c_6 b_6 + c_7 b_7 + c_8 b_8 + c_9 b_9 + c_{10} b_{10} &= \frac{1}{2} \\
 c_2^2 b_2 + c_3^2 b_3 + c_4^2 b_4 + c_5^2 b_5 + c_6^2 b_6 + c_7^2 b_7 + c_8^2 b_8 + c_9^2 b_9 + c_{10}^2 b_{10} &= \frac{1}{3} \\
 c_2^3 b_2 + c_3^3 b_3 + c_4^3 b_4 + c_5^3 b_5 + c_6^3 b_6 + c_7^3 b_7 + c_8^3 b_8 + c_9^3 b_9 + c_{10}^3 b_{10} &= \frac{1}{4} \\
 c_2^4 b_2 + c_3^4 b_3 + c_4^4 b_4 + c_5^4 b_5 + c_6^4 b_6 + c_7^4 b_7 + c_8^4 b_8 + c_9^4 b_9 + c_{10}^4 b_{10} &= \frac{1}{5} \\
 c_2^5 b_2 + c_3^5 b_3 + c_4^5 b_4 + c_5^5 b_5 + c_6^5 b_6 + c_7^5 b_7 + c_8^5 b_8 + c_9^5 b_9 + c_{10}^5 b_{10} &= \frac{1}{6} \\
 c_2^6 b_2 + c_3^6 b_3 + c_4^6 b_4 + c_5^6 b_5 + c_6^6 b_6 + c_7^6 b_7 + c_8^6 b_8 + c_9^6 b_9 + c_{10}^6 b_{10} &= \frac{1}{7} \\
 c_2^7 b_2 + c_3^7 b_3 + c_4^7 b_4 + c_5^7 b_5 + c_6^7 b_6 + c_7^7 b_7 + c_8^7 b_8 + c_9^7 b_9 + c_{10}^7 b_{10} &= \frac{1}{8}
 \end{aligned}$$

$$\begin{aligned} c_2^8 b_2 + c_3^8 b_3 + c_4^8 b_4 + c_5^8 b_5 + c_6^8 b_6 + c_7^8 b_7 + c_8^8 b_8 + c_9^8 b_9 + c_{10}^8 b_{10} &= \frac{1}{9} \\ c_2^9 b_2 + c_3^9 b_3 + c_4^9 b_4 + c_5^9 b_5 + c_6^9 b_6 + c_7^9 b_7 + c_8^9 b_8 + c_9^9 b_9 + c_{10}^9 b_{10} &= \frac{1}{10} \end{aligned}$$

$$\begin{aligned} &b_3 a_{32} c_2 + b_4 (a_{42} c_2 + a_{43} c_3) + b_5 (a_{52} c_2 + a_{53} c_3 a_{54} c_4) + b_6 (a_{62} c_2 + a_{63} c_3 \\ &+ a_{63} c_3 + a_{64} c_4 + a_{65} c_5) + b_7 (a_{72} c_2 + a_{73} c_3 + a_{74} c_4 + a_{75} c_5 + a_{76} c_6) + \\ &b_8 (a_{82} c_2 + a_{83} c_3 + a_{84} c_4 + a_{85} c_5 + a_{86} c_6 + a_{87} c_7) + b_9 (a_{92} c_2 + a_{93} c_3 + \\ &a_{94} c_4 + a_{95} c_5 + a_{96} c_6 + a_{97} c_7 + a_{98} c_8) + b_{10} (a_{102} c_2 + a_{103} c_3 + \\ &a_{104} c_4 + a_{105} c_5 + a_{106} c_6 + a_{107} c_7 + a_{108} c_8 + a_{109} c_9) = \frac{1}{6} \end{aligned}$$

$$\begin{aligned} &b_3 a_{32} c_2^2 + b_4 (a_{42} c_2^2 + a_{43} c_3^2) + b_5 (a_{52} c_2^2 + a_{53} c_3^2 + a_{54} c_4^2) + b_6 (a_{62} c_2^2 + \\ &a_{63} c_3^2 + a_{64} c_4^2 + a_{65} c_5^2) + b_7 (a_{72} c_2^2 + a_{73} c_3^2 + a_{74} c_4^2 + a_{75} c_5^2 + a_{76} c_6^2) + \\ &b_8 (a_{82} c_2^2 + a_{83} c_3^2 + a_{84} c_4^2 + a_{85} c_5^2 + a_{86} c_6^2 + a_{87} c_7^2) + b_9 (a_{92} c_2^2 + \\ &a_{93} c_3^2 + a_{94} c_4^2 + a_{95} c_5^2 + a_{96} c_6^2 + a_{97} c_7^2 + a_{98} c_8^2) + b_{10} (a_{102} c_2^2 + \\ &a_{103} c_3^2 + a_{104} c_4^2 + a_{105} c_5^2 + a_{106} c_6^2 + a_{107} c_7^2 + a_{108} c_8^2 + a_{109} c_9^2) = \frac{1}{12} \end{aligned}$$

$$\begin{aligned} &b_3 a_{32} c_2^3 + b_4 (a_{42} c_2^3 + a_{43} c_3^3) + b_5 (a_{52} c_2^3 + a_{53} c_3^3 + a_{54} c_4^3) + b_6 (a_{62} c_2^3 + \\ &a_{63} c_3^3 + a_{64} c_4^3 + a_{65} c_5^3) + b_7 (a_{72} c_2^3 + a_{73} c_3^3 + a_{74} c_4^3 + a_{75} c_5^3 + a_{76} c_6^3) + \\ &b_8 (a_{82} c_2^3 + a_{83} c_3^3 + a_{84} c_4^3 + a_{85} c_5^3 + a_{86} c_6^3 + a_{87} c_7^3) + b_9 (a_{92} c_2^3 + \\ &a_{93} c_3^3 + a_{94} c_4^3 + a_{95} c_5^3 + a_{96} c_6^3 + a_{97} c_7^3 + a_{98} c_8^3) + b_{10} (a_{102} c_2^3 + \\ &a_{103} c_3^3 + a_{104} c_4^3 + a_{105} c_5^3 + a_{106} c_6^3 + a_{107} c_7^3 + a_{108} c_8^3 + a_{109} c_9^3) = \frac{1}{20} \end{aligned}$$

$$\begin{aligned} &b_3 a_{32} c_2^4 + b_4 (a_{42} c_2^4 + a_{43} c_3^4) + b_5 (a_{52} c_2^4 + a_{53} c_3^4 + a_{54} c_4^4) + b_6 (a_{62} c_2^4 + \\ &a_{63} c_3^4 + a_{64} c_4^4 + a_{65} c_5^4) + b_7 (a_{72} c_2^4 + a_{73} c_3^4 + a_{74} c_4^4 + a_{75} c_5^4 + a_{76} c_6^4) + \\ &b_8 (a_{82} c_2^4 + a_{83} c_3^4 + a_{84} c_4^4 + a_{85} c_5^4 + a_{86} c_6^4 + a_{87} c_7^4) + b_9 (a_{92} c_2^4 + \\ &a_{93} c_3^4 + a_{94} c_4^4 + a_{95} c_5^4 + a_{96} c_6^4 + a_{97} c_7^4 + a_{98} c_8^4) + b_{10} (a_{102} c_2^4 + \\ &a_{103} c_3^4 + a_{104} c_4^4 + a_{105} c_5^4 + a_{106} c_6^4 + a_{107} c_7^4 + a_{108} c_8^4 + a_{109} c_9^4) = \frac{1}{30} \end{aligned}$$

$$\begin{aligned}
 & b_3 a_{32} c_2^5 + b_4 (a_{42} c_2^5 + a_{43} c_3^5) + b_5 (a_{52} c_2^5 + a_{53} c_3^5 + a_{54} c_4^5) + b_6 (a_{62} c_2^5 + \\
 & a_{63} c_3^5 + a_{64} c_4^5 + a_{65} c_5^5) + b_7 (a_{72} c_2^5 + a_{73} c_3^5 + a_{74} c_4^5 + a_{75} c_5^5 + a_{76} c_6^5) + \\
 & b_8 (a_{82} c_2^5 + a_{83} c_3^5 + a_{84} c_4^5 + a_{85} c_5^5 + a_{86} c_6^5 + a_{87} c_7^5) + b_9 (a_{92} c_2^5 + \\
 & a_{93} c_3^5 + a_{94} c_4^5 + a_{95} c_5^5 + a_{96} c_6^5 + a_{97} c_7^5 + a_{98} c_8^5) b_{10} (a_{102} c_2^5 + \\
 & a_{103} c_3^5 + a_{104} c_4^5 + a_{105} c_5^5 + a_{106} c_6^5 + a_{107} c_7^5 + a_{108} c_8^5) + a_{109} c_9^5 = \frac{1}{42}
 \end{aligned}$$

$$\begin{aligned}
 & b_3 a_{32} c_2^6 + b_4 (a_{42} c_2^6 + a_{43} c_3^6) + b_5 (a_{52} c_2^6 + a_{53} c_3^6 + a_{54} c_4^6) + b_6 (a_{62} c_2^6 + \\
 & a_{63} c_3^6 + a_{64} c_4^6 + a_{65} c_5^6) + b_7 (a_{72} c_2^6 + a_{73} c_3^6 + a_{74} c_4^6 + a_{75} c_5^6 + a_{76} c_6^6) + \\
 & b_8 (a_{82} c_2^6 + a_{83} c_3^6 + a_{84} c_4^6 + a_{85} c_5^6 + a_{86} c_6^6 + a_{87} c_7^6) + b_9 (a_{92} c_2^6 + \\
 & a_{93} c_3^6 + a_{94} c_4^6 + a_{95} c_5^6 + a_{96} c_6^6 + a_{97} c_7^6 + a_{98} c_8^6) + b_{10} (a_{102} c_2^6 + \\
 & a_{103} c_3^6 + a_{104} c_4^6 + a_{105} c_5^6 + a_{106} c_6^6 + a_{107} c_7^6 + a_{108} c_8^6 + a_{109} c_9^6) = \frac{1}{56}
 \end{aligned}$$

$$\begin{aligned}
 & b_3 a_{32} c_2^7 + b_4 (a_{42} c_2^7 + a_{43} c_3^7) + b_5 (a_{52} c_2^7 + a_{53} c_3^7 + a_{54} c_4^7) + b_6 (a_{62} c_2^7 + \\
 & a_{63} c_3^7 + a_{64} c_4^7 + a_{65} c_5^7) + b_7 (a_{72} c_2^7 + a_{73} c_3^7 + a_{74} c_4^7 + a_{75} c_5^7 + a_{76} c_6^7) + \\
 & b_8 (a_{82} c_2^7 + a_{83} c_3^7 + a_{84} c_4^7 + a_{85} c_5^7 + a_{86} c_6^7 + a_{87} c_7^7) + b_9 (a_{92} c_2^7 + \\
 & a_{93} c_3^7 + a_{94} c_4^7 + a_{95} c_5^7 + a_{96} c_6^7 + a_{97} c_7^7 + a_{98} c_8^7) + b_{10} (a_{102} c_2^7 + \\
 & a_{103} c_3^7 + a_{104} c_4^7 + a_{105} c_5^7 + a_{106} c_6^7 + a_{107} c_7^7 + a_{108} c_8^7 + a_{109} c_9^7) = \frac{1}{72}
 \end{aligned}$$

Atau dapat ditulis dalam notasi sigma sebagai berikut,

$$\sum_{i=1}^m b_i = 1 \quad \text{dimana } m = 10 \quad (4.5)$$

$$\sum_{i=2}^m b_i c_i^p = \frac{1}{p+1}, \quad \text{dimana } p = 1, 2, 3, 4, \dots, m-1 \quad (4.6)$$

$$\sum_{i=3}^{m-1} b_i \left(\sum_{j=2}^{i-1} c_j^q a_{ij} \right) = \frac{1}{(q+1)(q+2)}, \quad \text{dimana } q = 1, 3, 4, \dots, m-3 \quad (4.7)$$

Terbukti

Corollary 4.1 Formula Metode Runge-Kutta Orde Sepuluh (RK10A)

Untuk nilai ukuran langkah sepanjang h maka formula metode Runge-Kutta orde sepuluh adalah sebagai berikut:

$$y_{n+1} = y_n + \frac{h}{89600}(2857k_1 + 15741k_2 + 1080k_3 + 19344k_4 + 5778k_5 + 5778k_6 + 19344k_7 + 1080k_8 + 15741k_9 + 2857k_{10})$$

dengan,

$$\begin{aligned} k_1 &= f(x_n, y_n) \\ k_2 &= f\left(x_n + \frac{h}{9}, y_n + \frac{h}{9}k_1\right) \\ k_3 &= f\left(x_n + \frac{2h}{9}, y_n + \frac{h}{360}(-1617k_1 + 1697k_2)\right) \\ k_4 &= f\left(x_n + \frac{3h}{9}, y_n + \frac{h}{1080}(2501k_1 + 1000k_2 - 3141k_3)\right) \\ k_5 &= f\left(x_n + \frac{4h}{9}, y_n + \frac{h}{8667}(-154197k_1 - 8620k_2 - 8640k_3 + 175309k_4)\right) \\ k_6 &= f\left(x_n + \frac{5h}{9}, y_n + \frac{h}{5778}(99353k_1 - 5700k_2 - 5740k_3 + 5620k_4 - 90323k_5)\right) \\ k_7 &= f\left(x_n + \frac{6h}{9}, y_n + \frac{h}{5580}(-16343k_1 - 520k_2 + 5540k_3 - 5420k_4 - 5560k_5 + 26023k_6)\right) \\ k_8 &= f\left(x_n + \frac{7h}{9}, y_n + \frac{h}{450}(11546k_1 + 445k_2 - 440k_3 + 420k_4 + 320k_5 + 420k_6 - 12361k_7)\right) \\ k_9 &= f\left(x_n + \frac{8h}{9}, y_n + \frac{h}{4770}(-1281k_1 + 4720k_2 + 390k_3 + 1280k_4 - 2860k_5 + 2700k_6 - 4762k_7 + 4053k_8)\right) \\ k_{10} &= f\left(x_n + h, y_n + \frac{h}{1800}(-2500k_1 + 1000k_2 + 120k_3 + 1200k_4 - 600k_5 + 240k_6 + 1800k_7 + 540k_8)\right) \end{aligned}$$

Bukti. Untuk Membuktikan Corollary 4.1 harus diselesaikan sedemikian hingga semua koefisiennya dapat ditemukan. Sistem persamaan pada Lemma 4.1.1 memiliki banyak koefisien, namun dapat diselesaikan dengan menetapkan nilai dari

$c_1, c_2, c_3, \dots, c_{10}$ terlebih dahulu sehingga diperoleh konstanta-konstanta b_1, b_2, \dots, b_{10} dengan memenuhi syarat metode Runge-Kutta yaitu $c_i = \sum_{j=1}^{10} a_{ij}$ dan $\sum_{i=1}^{10} b_i = 1$. Ketetapan-ketetapan tersebut adalah $c_1 = 0, c_2 = \frac{1}{9}, c_3 = \frac{2}{9}, c_4 = \frac{3}{9}, c_5 = \frac{4}{9}, c_6 = \frac{5}{9}, c_7 = \frac{6}{9}, c_8 = \frac{7}{9}, c_9 = \frac{8}{9}$ dan, $c_{10} = 1$.

Selanjutnya Sistem persamaan (4.7) diselesaikan dengan mensubstitusikan nilai-nilai tetapan $c_1, c_2, c_3, \dots, c_{10}$ untuk mencari nilai koefisien matrik $a_{21}, a_{31}, a_{32}, \dots, a_{109}$. Namun sistem persamaan (4.7) sangat sulit untuk diselesaikan, sehingga sistem persamaan (4.7) perlu dimodifikasi menjadi,

$$\sum_{i=2}^9 c_i^k \left(\sum_{j=i+1}^{10} b_j a_{ji} \right), \quad k = 1, 2, \dots, 7 \quad (4.8)$$

dengan memisalkan:

$$\begin{aligned} \sum_{j=i+1}^{10} b_j a_{ji} &= A, \text{ untuk } i = 2, & \sum_{j=i+1}^{10} b_j a_{ji} &= B, \text{ untuk } i = 3 \\ \sum_{j=i+1}^{10} b_j a_{ji} &= C, \text{ untuk } i = 4, & \sum_{j=i+1}^{10} b_j a_{ji} &= D, \text{ untuk } i = 5 \\ \sum_{j=i+1}^{10} b_j a_{ji} &= E, \text{ untuk } i = 6, & \sum_{j=i+1}^{10} b_j a_{ji} &= F, \text{ untuk } i = 7 \\ \sum_{j=i+1}^{10} b_j a_{ji} &= G, \text{ untuk } i = 8, & \sum_{j=i+1}^{10} b_j a_{ji} &= H, \text{ untuk } i = 9 \end{aligned}$$

maka didapatkan sistem persamaan baru yang digunakan untuk mencari nilai A, B, C, D, \dots, H . Dengan ditemukannya nilai A, B, C, D, \dots, H , maka koefisien matrik yang dicari dapat ditemukan karena nilai A, B, C, D, \dots, H memuat koefisien $a_{21}, a_{31}, a_{32}, \dots, a_{109}$. Dengan memenuhi syarat metode Runge-Kutta yaitu $c_i = \sum_{j=1}^{10} a_{ij}$ dan $\sum_{i=1}^{10} b_i = 1$, tetapan-tetapan tersebut secara rinci dapat digambarkan pada *Butcher array* tabel (4.1).

Tabel 4.1: Matriks Koefisien Runge-Kutta Order Sepuluh (RK10A)

0	0									
$\frac{1}{9}$	$\frac{1}{9}$	0								
$\frac{2}{9}$	$-\frac{1617}{360}$	$\frac{1697}{360}$	0							
$\frac{3}{9}$	$\frac{2501}{1080}$	$\frac{1000}{1080}$	$-\frac{3141}{1080}$	0						
$\frac{4}{9}$	$-\frac{154197}{8667}$	$-\frac{8620}{8667}$	$-\frac{8640}{8667}$	$\frac{175309}{8667}$	0					
$\frac{5}{9}$	$\frac{99353}{5778}$	$-\frac{5700}{5778}$	$-\frac{5740}{5778}$	$\frac{5620}{5778}$	$-\frac{90323}{5778}$	0				
$\frac{6}{9}$	$-\frac{16343}{5580}$	$-\frac{520}{5580}$	$\frac{5540}{5580}$	$-\frac{5420}{5580}$	$-\frac{5560}{5580}$	$\frac{26023}{5580}$	0			
$\frac{7}{9}$	$\frac{11546}{450}$	$\frac{445}{450}$	$-\frac{440}{450}$	$\frac{420}{450}$	$\frac{320}{450}$	$\frac{420}{450}$	$-\frac{12360}{450}$	0		
$\frac{8}{9}$	$-\frac{1281}{4770}$	$\frac{4720}{4770}$	$\frac{390}{4770}$	$\frac{1280}{4770}$	$-\frac{2860}{4770}$	$\frac{2700}{4770}$	$-\frac{4762}{4770}$	$\frac{4053}{4770}$	0	
1	$-\frac{2500}{1800}$	$\frac{1000}{1800}$	$\frac{120}{1800}$	$\frac{1200}{1800}$	$-\frac{600}{1800}$	$\frac{240}{1800}$	$\frac{1800}{1800}$	$\frac{540}{1800}$	0	0
	$\frac{2857}{89600}$	$\frac{15741}{89600}$	$\frac{1080}{89600}$	$\frac{19344}{89600}$	$\frac{5778}{89600}$	$\frac{5778}{89600}$	$\frac{19344}{89600}$	$\frac{1080}{89600}$	$\frac{15741}{89600}$	$\frac{2857}{89600}$

Masing-masing koefisien jika disubstitusikan ke dalam persamaan (4.1) diperoleh:

$$y_{n+1} = y_n + \frac{h}{89600} (2857k_1 + 15741k_2 + 1080k_3 + 19344k_4 + 5778k_5 + 5778k_6 + 19344k_7 + 1080k_8 + 15741k_9 + 2857k_{10})$$

dengan :

$$\begin{aligned}
k_1 &= f(x_n, y_n) \\
k_2 &= f\left(x_n + \frac{h}{9}, y_n + \frac{h}{9}k_1\right) \\
k_3 &= f\left(x_n + \frac{2h}{9}, y_n + \frac{h}{360}(-1617k_1 + 1697k_2)\right) \\
k_4 &= f\left(x_n + \frac{3h}{9}, y_n + \frac{h}{1080}(2501k_1 + 1000k_2 - 3141k_3)\right) \\
k_5 &= f\left(x_n + \frac{4h}{9}, y_n + \frac{h}{8667}(-154197k_1 - 8620k_2 - 8640k_3 + 175309k_4)\right) \\
k_6 &= f\left(x_n + \frac{5h}{9}, y_n + \frac{h}{5778}(99353k_1 - 5700k_2 - 5740k_3 + 5620k_4 - 90323k_5)\right) \\
k_7 &= f\left(x_n + \frac{6h}{9}, y_n + \frac{h}{5580}(-16343k_1 - 520k_2 + 5540k_3 - 5420k_4 - 5560k_5 + 26023k_6)\right) \\
k_8 &= f\left(x_n + \frac{7h}{9}, y_n + \frac{h}{450}(11546k_1 + 445k_2 - 440k_3 + 420k_4 + 320k_5 + 420k_6 - 12361k_7)\right) \\
k_9 &= f\left(x_n + \frac{8h}{9}, y_n + \frac{h}{4770}(-1281k_1 + 4720k_2 + 390k_3 + 1280k_4 - 2860k_5 + 2700k_6 - 4762k_7 + 4053k_8)\right) \\
k_{10} &= f\left(x_n + h, y_n + \frac{h}{1800}(-2500k_1 + 1000k_2 + 120k_3 + 1200k_4 - 600k_5 + 240k_6 + 1800k_7 + 540k_8)\right)
\end{aligned}$$

Berdasarkan definisi (2.6.3) tentang order, metode Rung-Kutta order sepuluh harus memenuhi $I_n = \phi(h^{10+1})$. Dengan demikian penjabaran persamaan (4.4) adalah sebagai berikut:

$$\begin{aligned}
 y_{n+1} = & y_n + hf(\sum_{i=1}^{10} b_i) + h^2(\sum_{i=2}^{10} c_i b_i)J + h^3(\frac{1}{2}(\sum_{i=2}^{10} c_i b_i)K + (b_3 a_{32} c_2 + b_4(\sum_{i=2}^3 a_{4i} c_i) \\
 & + b_5(\sum_{i=2}^4 a_{5i} c_i) + b_6(\sum_{i=2}^5 a_{6i} c_i) + b_7(\sum_{i=2}^6 a_{7i} c_i) + b_8(\sum_{i=2}^7 a_{8i} c_i) + b_9(\sum_{i=2}^8 a_{9i} c_i \\
 & + b_{10}(\sum_{i=2}^9 a_{10i} c_i))Jf_y) + h^4(\frac{1}{6}(\sum_{i=2}^{10} c_i^3 b_i)L + \frac{1}{2}(b_3 a_{32} c_2^2 + b_4(\sum_{i=2}^3 a_{4i} c_i^2) \\
 & + b_5(\sum_{i=2}^4 a_{5i} c_i^2) + b_6(\sum_{i=2}^5 a_{6i} c_i^2) + b_7(\sum_{i=2}^6 a_{7i} c_i^2) + b_8(\sum_{i=2}^7 a_{8i} c_i^2) + b_9(\sum_{i=2}^8 a_{9i} c_i^2) \\
 & + b_{10}(\sum_{i=2}^9 a_{10i} c_i^2))Kf_y) + h^5(\frac{1}{24}(\sum_{i=2}^{10} c_i^4 b_i)M + \frac{1}{6}(b_3 a_{32} c_2^3 + b_4(\sum_{i=2}^3 a_{4i} c_i^3) + \\
 & b_5(\sum_{i=2}^4 a_{5i} c_i^3) + b_6(\sum_{i=2}^5 a_{6i} c_i^3) + b_7(\sum_{i=2}^6 a_{7i} c_i^3) + b_8(\sum_{i=2}^7 a_{8i} c_i^3) + b_9(\sum_{i=2}^8 a_{9i} c_i^3) \\
 & + b_{10}(\sum_{i=2}^9 a_{10i} c_i^3))Lf_y) + h^6(\frac{1}{5!}(\sum_{i=2}^{10} c_i^5 b_i)N + \frac{1}{4!}(b_3 a_{32} c_2^4 + b_4(\sum_{i=2}^3 a_{4i} c_i^4) + b_5 \\
 & (\sum_{i=2}^4 a_{5i} c_i^4) + b_6(\sum_{i=2}^5 a_{6i} c_i^4) + b_7(\sum_{i=2}^6 a_{7i} c_i^4) + b_8(\sum_{i=2}^7 a_{8i} c_i^4) + b_9(\sum_{i=2}^8 a_{9i} c_i^4) + \\
 & b_{10}(\sum_{i=2}^9 a_{10i} c_i^4))Mf_y) + h^7(\frac{1}{6!}(\sum_{i=2}^{10} c_i^6 b_i)O + \frac{1}{5!}(b_3 a_{32} c_2^5 + b_4(\sum_{i=2}^3 a_{4i} c_i^5) + b_5 \\
 & (\sum_{i=2}^4 a_{5i} c_i^5) + b_6(\sum_{i=2}^5 a_{6i} c_i^5) + b_7(\sum_{i=2}^6 a_{7i} c_i^5) + b_8(\sum_{i=2}^7 a_{8i} c_i^5) + b_9(\sum_{i=2}^8 a_{9i} c_i^5) + \\
 & b_{10}(\sum_{i=2}^9 a_{10i} c_i^5))Nf_y) + h^8(\frac{1}{7!}(\sum_{i=2}^{10} c_i^7 b_i)P + \frac{1}{6!}(b_3 a_{32} c_2^6 + b_4(\sum_{i=2}^3 a_{4i} c_i^6) + b_5 \\
 & (\sum_{i=2}^4 a_{5i} c_i^6) + b_6(\sum_{i=2}^5 a_{6i} c_i^6) + b_7(\sum_{i=2}^6 a_{7i} c_i^6) + b_8(\sum_{i=2}^7 a_{8i} c_i^6) + b_9(\sum_{i=2}^8 a_{9i} c_i^6) + \\
 & b_{10}(\sum_{i=2}^9 a_{10i} c_i^6))Of_y) + h^9(\frac{1}{8!}(\sum_{i=2}^{10} c_i^8 b_i)Q + \frac{1}{7!}(b_3 a_{32} c_2^7 + b_4(\sum_{i=2}^3 a_{4i} c_i^7) + b_5 \\
 & (\sum_{i=2}^4 a_{5i} c_i^7) + b_6(\sum_{i=2}^5 a_{6i} c_i^7) + b_7(\sum_{i=2}^6 a_{7i} c_i^7) + b_8(\sum_{i=2}^7 a_{8i} c_i^7) + b_9(\sum_{i=2}^8 a_{9i} c_i^7) + \\
 & b_{10}(\sum_{i=2}^9 a_{10i} c_i^7))Pf_y) + h^{10}(\frac{1}{9!}(\sum_{i=2}^{10} c_i^9 b_i)R)
 \end{aligned}$$

$$\begin{aligned}
 y_{n+1} = & y_n + hf + h^2(\frac{1}{2})J + h^3[\frac{1}{2}(\frac{1}{3})K + \frac{1}{6}Jf_y] + h^4[\frac{1}{3!}(\frac{1}{4})L + \frac{1}{2}(\frac{1}{12})Kf_y] \\
 & + h^5[\frac{1}{4!}(\frac{1}{5})M + \frac{1}{3!}(\frac{1}{20})Lf_y] + h^6[\frac{1}{5!}(\frac{1}{6})N + \frac{1}{4!}(\frac{1}{30})Mf_y] + h^7[\frac{1}{6!}(\frac{1}{7})O \\
 & + \frac{1}{5!}(\frac{1}{42})Nf_y] + h^8[\frac{1}{7!}(\frac{1}{8})P + \frac{1}{6!}(\frac{1}{56})Of_y] + h^9[\frac{1}{8!}(\frac{1}{9})Q + (\frac{1}{9!})(\frac{1}{72})Pf_y] \\
 & + h^{10}[\frac{1}{9!}(\frac{1}{10})R]
 \end{aligned}$$

$$\begin{aligned}
 y_{n+1} = & y_n + hf + h^2\left(\frac{1}{2}\right)J + h^3\frac{1}{6}[K + Jf_y] + h^4\frac{1}{24}[L + Kf_y] + h^5\frac{1}{120}[M + \\
 & Lf_y] + h^6\frac{1}{720}[N + Mf_y] + h^7\frac{1}{5040}[O + Nf_y] + h^8\frac{1}{40320}[P + Of_y] \\
 & + h^9\frac{1}{362880}(Q + Pf_y) + h^{10}\frac{1}{3628800}R
 \end{aligned} \tag{4.9}$$

Kesalahan pemenggalan lokal adalah selisih dari persamaan Deret Taylor $y(x_{n+1})$ dengan persamaan solusi Numerik Metode Runge-Kutta y_{n+1} , sehingga diperoleh:

$$\begin{aligned}
 I_n &= y(x_{n+1}) - y_{n+1} \\
 I_n &= [y(x_n) + hy^{(1)}(x_n) + \frac{1}{2!}h^2y^{(2)}(x_n) + \frac{1}{3!}h^3y^{(3)}(x_n) + \frac{1}{4!}h^4y^{(4)}(x_n) + \\
 & \frac{1}{5!}h^5y^{(5)}(x_n) + \frac{1}{6!}h^6y^{(6)}(x_n) + \frac{1}{7!}h^7y^{(7)}(x_n) + \frac{1}{8!}h^8y^{(8)}(x_n) + \\
 & \frac{1}{9!}h^9y^{(9)}(x_n) + \frac{1}{10!}h^{10}y^{(10)}(x_n) + \frac{1}{11!}h^{11}y^{(11)}(x_n) + \dots] \\
 & - [y_n + hf + h^2\left(\frac{1}{2}\right)J + h^3\frac{1}{6}[K + Jf_y] + h^4\frac{1}{24}[L + Kf_y] + h^5\frac{1}{120}[M + \\
 & Lf_y] + h^6\frac{1}{720}[N + Mf_y] + h^7\frac{1}{5040}[O + Nf_y] + h^8\frac{1}{40320}[P + Of_y] \\
 & + h^9\left(\frac{1}{40320}\right)(Q + Pf_y) + h^{10}\left(\frac{1}{3628800}\right)R] \\
 & = [y(x_n) + hy^{(1)}(x_n) + \frac{1}{2!}h^2y^{(2)}(x_n) + \frac{1}{3!}h^3y^{(3)}(x_n) + \frac{1}{4!}h^4y^{(4)}(x_n) + \\
 & \frac{1}{5!}h^5y^{(5)}(x_n) + \frac{1}{6!}h^6y^{(6)}(x_n) + \frac{1}{7!}h^7y^{(7)}(x_n) + \frac{1}{8!}h^8y^{(8)}(x_n) + \\
 & \frac{1}{9!}h^9y^{(9)}(x_n) + \frac{1}{10!}h^{10}y^{(10)}(x_n) + \frac{1}{11!}h^{11}y^{(11)}(x_n) + \dots] \\
 & - [y_n + hy_n^{(1)} + \frac{1}{2!}h^2y_n^{(2)} + \frac{1}{3!}h^3y_n^{(3)} + \frac{1}{4!}h^4y_n^{(4)} + \frac{1}{5!}h^5y_n^{(5)} + \frac{1}{6!}h^6y_n^{(6)} \\
 & + \frac{1}{7!}h^7y_n^{(7)} + \frac{1}{8!}h^8y_n^{(8)} + \frac{1}{9!}h^9y_n^{(9)} + \frac{1}{10!}h^{10}y_n^{(10)}] \\
 I_n &= y(x_n) - y_n + h(y^{(1)}(x_n) - y_n^{(1)}) + \frac{1}{2!}h^2(y^{(2)}(x_n) - y_n^{(2)}) + \frac{1}{3!}h^3(y^{(3)}(x_n) \\
 & - y_n^{(3)}) + \frac{1}{4!}h^4(y^{(4)}(x_n) - y_n^{(4)}) + \frac{1}{5!}h^5(y^{(5)}(x_n) - y_n^{(5)}) + \frac{1}{6!}h^6(y^{(6)}(x_n) \\
 & - y_n^{(6)}) + \frac{1}{7!}h^7(y^{(7)}(x_n) - y_n^{(7)}) + \frac{1}{8!}h^8(y^{(8)}(x_n) - y_n^{(8)}) + \frac{1}{9!}h^9(y^{(9)}(x_n) \\
 & - y_n^{(9)}) + \frac{1}{10!}h^{10}(y^{(10)}(x_n) - y_n^{(10)}) + \frac{1}{11!}h^{11}y^{(11)}(x_n) + \dots
 \end{aligned}$$

Karena $y_n \approx y(x_n)$, maka:

$$\begin{aligned}
 I_n &= \frac{1}{11!} h^{11} y^{(11)}(x_n) + \dots \\
 &= \frac{1}{11!} + \dots \\
 &= \phi(h^{11}) \\
 &= \phi(h^{10+1})
 \end{aligned} \tag{4.10}$$

Dari penjabaran di atas, dapat disimpulkan bahwa metode ini merupakan Metode Runge-Kutta orde sepuluh dengan kesalahan pemenggalan lokal $\phi(h^{10+1})$ serta merupakan metode yang konsisten karena memenuhi syarat orde minimal satu. **Terbukti.**

4.2 Konvergensi Metode Runge-Kutta Orde Sepuluh

Setelah mendapatkan formula metode Runge-Kutta orde sepuluh, diperlukan uji konvergensi secara teoritis untuk mengetahui apakah metode Runge-Kutta orde sepuluh dapat digunakan dalam menyelesaikan suatu persamaan diferensial biasa non linier.

Teorema 4.1 (Konvergensi Metode Runge-Kutta Orde Sepuluh) *Metode Runge-Kutta order sepuluh merupakan metode yang konvergen karena telah memenuhi sifat $\|e_n\| \leq \frac{h^{10} M_{11}}{11! \check{L}} (e^{(x_n - x_0) \check{L}} - 1)$, dimana \check{L} adalah konstanta Lipschitz.*

Bukti.

Pembuktian konvergensi metode Runge-Kutta order sepuluh didasarkan pada **definisi 2.6.5 dan definisi 2.6.6**. Penyelesaian yang eksak dari persamaan diferensial pada $x = x_n$ disebut $y(x_n)$, dan penyelesaian aproksimasi (numerik) dinamakan y_n . Solusi numerik metode Runge-Kutta orde sepuluh yang diperoleh adalah:

$$\begin{aligned}
y_{n+1} &= y_n + h(b_1k_1 + b_2k_2 + b_3k_3 + b_4k_4 + b_5k_5 + b_6k_6 + b_7k_7 + b_8k_8 + \\
&\quad b_9k_9 + b_{10}k_{10}) \\
&= y_n + \frac{h}{89600}(2857k_1 + 15741k_2 + 1080k_3 + 19344k_4 + 5778k_5 \\
&\quad + 5778k_6 + 19344k_7 + 1080k_8 + 15741k_9 + 2857k_{10})
\end{aligned}$$

dengan :

$$\begin{aligned}
k_1 &= f(x_n, y_n) \\
k_2 &= f\left(x_n + \frac{h}{9}, y_n + \frac{h}{9}k_1\right) \\
k_3 &= f\left(x_n + \frac{2}{9}h, y_n + \frac{h}{360}(-1617k_1 + 1697k_2)\right) \\
k_4 &= f\left(x_n + \frac{3}{9}h, y_n + \frac{h}{1080}(2501k_1 + 1000k_2 - 3141k_3)\right) \\
k_5 &= f\left(x_n + \frac{4}{9}h, y_n + \frac{h}{8667}(-154197k_1 - 8620k_2 - 8640k_3 + 175309k_4)\right) \\
k_6 &= f\left(x_n + \frac{5}{9}h, y_n + \frac{h}{5778}(99353k_1 - 5700k_2 - 5740k_3 + 5620k_4 - 90323k_5)\right) \\
k_7 &= f\left(x_n + \frac{6}{9}h, y_n + \frac{h}{5580}(-16343k_1 - 520k_2 + 5540k_3 - 5420k_4 - 5560k_5 \right. \\
&\quad \left. + 26023k_6)\right) \\
k_8 &= f\left(x_n + \frac{7}{9}h, y_n + \frac{h}{450}(11546k_1 + 445k_2 - 440k_3 + 420k_4 + 320k_5 \right. \\
&\quad \left. + 420k_6 - 12361k_7)\right) \\
k_9 &= f\left(x_n + \frac{8}{9}h, y_n + \frac{h}{4770}(-1281k_1 + 4720k_2 + 390k_3 + 1280k_4 - 2860k_5 \right. \\
&\quad \left. + 2700k_6 - 4762k_7 + 4053k_8)\right) \\
k_{10} &= f\left(x_n + h, y_n + \frac{h}{1800}(-2500k_1 + 1000k_2 + 120k_3 + 1200k_4 - 600k_5 + 240k_6 \right. \\
&\quad \left. + 1800k_7 + 540k_8)\right)
\end{aligned}$$

dengan mensubstitusikan kembali k_1, k_2, \dots, k_{10} ke persamaan diatas dan semua koefisien yang telah didapat berdasarkan pada persamaan (4.3), maka diperoleh :

$$\begin{aligned}
 y_{n+1} = & y_n + hf + h^2\left(\frac{1}{2}\right)J + h^3\left[\frac{1}{2}\left(\frac{1}{3}\right)K + \frac{1}{6}Jf_y\right] + h^4\left[\frac{1}{3!}\left(\frac{1}{4}\right)L + \frac{1}{2}\left(\frac{1}{12}\right)Kf_y\right] \\
 & + h^5\left[\frac{1}{4!}\left(\frac{1}{5}\right)M + \frac{1}{3!}\left(\frac{1}{20}\right)Lf_y\right] + h^6\left[\frac{1}{5!}\left(\frac{1}{6}\right)N + \frac{1}{4!}\left(\frac{1}{30}\right)Mf_y\right] + h^7\left[\frac{1}{6!}\left(\frac{1}{7}\right)O \right. \\
 & \left. + \frac{1}{5!}\left(\frac{1}{42}\right)Nf_y\right] + h^8\left[\frac{1}{7!}\left(\frac{1}{8}\right)P + \frac{1}{6!}\left(\frac{1}{56}\right)Of_y\right] + h^9\left[\frac{1}{8!}\left(\frac{1}{9}\right)Q + \left(\frac{1}{9!}\right)\left(\frac{1}{72}Pf_y\right) \right. \\
 & \left. + h^{10}\left[\frac{1}{9!}\left(\frac{1}{10}\right)R\right]\right]
 \end{aligned}$$

$$\begin{aligned}
 y_{n+1} = & y_n + hf + h^2\left(\frac{1}{2}\right)J + h^3\frac{1}{6}[K + Jf_y] + h^4\frac{1}{24}[L + Kf_y] + h^5\frac{1}{120}[M + \\
 & Lf_y] + h^6\frac{1}{720}[N + Mf_y] + h^7\frac{1}{5040}[O + Nf_y] + h^8\frac{1}{40320}[P + Of_y] \\
 & + h^9\frac{1}{362880}(Q + Pf_y) + h^{10}\frac{1}{3628800}R
 \end{aligned}$$

$$\begin{aligned}
 y_{n+1} = & y_n + hy_n^{(1)} + \frac{1}{2!}h^2y_n^{(2)} + \frac{1}{3!}h^3y_n^{(3)} + \frac{1}{4!}h^4y_n^{(4)} + \frac{1}{5!}h^5y_n^{(5)} + \frac{1}{6!}h^6y_n^{(6)} \\
 & + \frac{1}{7!}h^7y_n^{(7)} + \frac{1}{8!}h^8y_n^{(8)} + \frac{1}{9!}h^9y_n^{(9)} + \frac{1}{10!}h^{10}y_n^{(10)}
 \end{aligned}$$

Selanjutnya, kesalahan global e_n dicari berdasarkan definisi kesalahan global, yaitu selisih dari solusi eksak $y(x_n)$ dengan solusi aproksimasi y_n metode Runge-Kutta orde sepuluh.

$$e_n = y(x_n) - y_n$$

Apabila y_0 eksak sebagaimana yang diasumsikan, maka $e_0 = 0$. Dengan asumsi bahwa turunan yang sesuai ada, maka $y(x_{n+1})$ dapat diperluas sekitar $x = x_n$ dengan menggunakan Deret Taylor.

$$\begin{aligned}
 y(x_{n+1}) = & y(x_n) + hy^{(1)}(x_n) + \frac{1}{2!}h^2y^{(2)}(x_n) + \frac{1}{3!}h^3y^{(3)}(x_n) + \frac{1}{4!}h^4y^{(4)}(x_n) + \\
 & \frac{1}{5!}h^5y^{(5)}(x_n) + \frac{1}{6!}h^6y^{(6)}(x_n) + \frac{1}{7!}h^7y^{(7)}(x_n) + \frac{1}{8!}h^8y^{(8)}(x_n) + \\
 & \frac{1}{9!}h^9y^{(9)}(x_n) + \frac{1}{10!}h^{10}y^{(10)}(x_n) + \frac{1}{11!}h^{11}y^{(11)}(\eta)
 \end{aligned}$$

dengan $x_n \leq \eta_n \leq x_{n+1}$. Besaran $\frac{1}{11!}h^{11}y^{(11)}(\eta)$ disebut kesalahan lokal, yaitu

kesalahan yang dibuat dalam satu langkah dari x_n ke x_{n+1} , dengan asumsi bahwa $y(x_n), y^{(1)}(x_n), \dots, y^{(10)}(x_n)$ diketahui secara eksak pada titik $x = x_n$.

Maka,

$$\begin{aligned}
 e_n &= y(x_n) - y_n \\
 e_{n+1} &= y(x_{n+1}) - y_{n+1} \\
 &= [y(x_n) + hy^{(1)}(x_n) + \frac{1}{2!}h^2y^{(2)}(x_n) + \frac{1}{3!}h^3y^{(3)}(x_n) + \frac{1}{4!}h^4y^{(4)}(x_n) + \\
 &\quad \frac{1}{5!}h^5y^{(5)}(x_n) + \frac{1}{6!}h^6y^{(6)}(x_n) + \frac{1}{7!}h^7y^{(7)}(x_n) + \frac{1}{8!}h^8y^{(8)}(x_n) + \\
 &\quad \frac{1}{9!}h^9y^{(9)}(x_n) + \frac{1}{10!}h^{10}y^{(10)}(x_n) + \frac{1}{11!}h^{11}y^{(11)}(\eta)] - \\
 &\quad [y_n + hy_n^{(1)} + \frac{1}{2!}h^2y_n^{(2)} + \frac{1}{3!}h^3y_n^{(3)} + \frac{1}{4!}h^4y_n^{(4)} + \frac{1}{5!}h^5y_n^{(5)} + \frac{1}{6!}h^6y_n^{(6)} \\
 &\quad + \frac{1}{7!}h^7y_n^{(7)} + \frac{1}{8!}h^8y_n^{(8)} + \frac{1}{9!}h^9y_n^{(9)} + \frac{1}{10!}h^{10}y_n^{(10)}] \\
 &= y(x_n) - y_n + (hy^{(1)}(x_n) - hy_n^{(1)}) + (\frac{1}{2!}h^2y^{(2)}(x_n) - \frac{1}{2!}h^2y_n^{(2)}) + (\frac{1}{3!}h^3y^{(3)}(x_n) \\
 &\quad - \frac{1}{3!}h^3y_n^{(3)}) + (\frac{1}{4!}h^4y^{(4)}(x_n) - \frac{1}{4!}h^4y_n^{(4)}) + (\frac{1}{5!}h^5y^{(5)}(x_n) - \frac{1}{5!}h^5y_n^{(5)}) + (\frac{1}{6!}h^6y^{(6)}(x_n) \\
 &\quad - \frac{1}{6!}h^6y_n^{(6)}) + (\frac{1}{7!}h^7y^{(7)}(x_n) - \frac{1}{7!}h^7y_n^{(7)}) + (\frac{1}{8!}h^8y^{(8)}(x_n) - \frac{1}{8!}h^8y_n^{(8)}) + (\frac{1}{9!}h^9y^{(9)}(x_n) \\
 &\quad - \frac{1}{9!}h^9y_n^{(9)}) + (\frac{1}{10!}h^{10}y^{(10)}(x_n) - \frac{1}{10!}h^{10}y_n^{(10)}) + \frac{1}{11!}h^{11}y^{(11)}(\eta) \\
 &= e_n + h(y^{(1)}(x_n) - y_n^{(1)}) + \frac{1}{2!}h^2(y^{(2)}(x_n) - y_n^{(2)}) + \frac{1}{3!}h^3(y^{(3)}(x_n) \\
 &\quad - y_n^{(3)}) + \frac{1}{4!}h^4(y^{(4)}(x_n) - y_n^{(4)}) + \frac{1}{5!}h^5(y^{(5)}(x_n) - y_n^{(5)}) + \frac{1}{6!}h^6(y^{(6)}(x_n) \\
 &\quad - y_n^{(6)}) + \frac{1}{7!}h^7(y^{(7)}(x_n) - y_n^{(7)}) + \frac{1}{8!}h^8(y^{(8)}(x_n) - y_n^{(8)}) + \frac{1}{9!}h^9(y^{(9)}(x_n) \\
 &\quad - y_n^{(9)}) + \frac{1}{10!}h^{10}(y^{(10)}(x_n) - y_n^{(10)}) + \frac{1}{11!}h^{11}y^{(11)}(\eta)
 \end{aligned}$$

berdasarkan syarat Lipschitz pada Definisi (2.6.5) yaitu $\|f(t, y_1) - f(t, y_2)\| \leq L\|y_1 - y_2\|$ dan diasumsikan $|y(\eta)^{(11)}| < M_{11}$ maka,

$$\|e_{n+1}\| \leq \|e_n + hL_1e_n + \frac{1}{2!}h^2L_2e_n + \frac{1}{3!}h^3L_3e_n + \frac{1}{4!}h^4L_4e_n + \frac{1}{5!}h^5L_5e_n + \frac{1}{6!}h^6L_6e_n + \frac{1}{7!}h^7L_7e_n + \frac{1}{8!}h^8L_8e_n + \frac{1}{9!}h^9L_9e_n + \frac{1}{10!}h^{10}L_{10}e_n + \frac{h^{11}}{11!}M_{11}\|$$

berdasarkan definisi norm vektor (definisi 2.4.1),

$$\begin{aligned} \|e_{n+1}\| &\leq \|e_n\| + hL_1\|e_n\| + \frac{1}{2!}h^2L_2\|e_n\| + \frac{1}{3!}h^3L_3\|e_n\| + \frac{1}{4!}h^4L_4\|e_n\| + \frac{1}{5!}h^5L_5\|e_n\| \\ &\quad + \frac{1}{6!}h^6L_6\|e_n\| + \frac{1}{7!}h^7L_7\|e_n\| + \frac{1}{8!}h^8L_8\|e_n\| + \frac{1}{9!}h^9L_9\|e_n\| + \frac{1}{10!}h^{10}L_{10}\|e_n\| + \frac{h^{11}}{11!}M_{11} \\ &= (1 + hL_1 + \frac{1}{2!}h^2L_2 + \frac{1}{3!}h^3L_3 + \frac{1}{4!}h^4L_4 + \frac{1}{5!}h^5L_5 + \frac{1}{6!}h^6L_6 + \frac{1}{7!}h^7L_7 \\ &\quad + \frac{1}{8!}h^8L_8 + \frac{1}{9!}h^9L_9 + \frac{1}{10!}h^{10}L_{10})\|e_n\| + \frac{h^{11}}{11!}M_{11} \\ &= (1 + h(L_1 + \frac{1}{2!}hL_2 + \frac{1}{3!}h^2L_3 + \frac{1}{4!}h^3L_4 + \frac{1}{5!}h^4L_5 + \frac{1}{6!}h^5L_6 + \frac{1}{7!}h^6L_7 \\ &\quad + \frac{1}{8!}h^7L_8 + \frac{1}{9!}h^8L_9 + \frac{1}{10!}h^9L_{10}))\|e_n\| + \frac{h^{11}}{11!}M_{11} \\ &= (1 + h(\check{L}_1 + \check{L}_2 + \check{L}_3 + \check{L}_4 + \check{L}_5 + \check{L}_6 + \check{L}_7 + \check{L}_8 + \check{L}_9 + \check{L}_{10}))\|e_n\| + \frac{h^{11}}{11!}M_{11} \\ &= (1 + h\check{L})\|e_n\| + \frac{h^{11}}{11!}M_{11} \\ \|e_{n+1}\| &\leq (1 + h\check{L})\|e_n\| + \frac{h^{11}}{11!}M_{11} \end{aligned}$$

fakta sebelumnya menyebutkan bahwa,

$$\begin{aligned} \|e_0\| &= 0 \\ \|e_1\| &\leq \frac{h^{11}}{11!}M_{11} \\ \|e_2\| &\leq (1 + h\check{L})\frac{h^{11}}{11!}M_{11} + \frac{h^{11}}{11!}M_{11} \end{aligned}$$

$$\begin{aligned} \|e_3\| &\leq (1 + h\check{L})(1 + h\check{L})\frac{h^{11}}{11!}M_{11} + (1 + h\check{L})\frac{h^{11}}{11!}M_{11} + \frac{h^{11}}{11!}M_{11} \\ &\vdots \\ \|e_n\| &\leq (1 + (1 + h\check{L}) + (1 + h\check{L})^2 + \dots + (1 + h\check{L})^{n-1})\frac{h^{11}}{11!}M_{11} \end{aligned}$$

berdasarkan deret geometri, $S_n = \frac{r^n - 1}{r - 1}$, maka

$$\begin{aligned} \|e_n\| &\leq \left(\frac{(1 + h\check{L})^n - 1}{(1 + h\check{L}) - 1}\right)\frac{h^{11}}{11!}M_{11} = \left(\frac{(1 + h\check{L})^n - 1}{(h\check{L})}\right)\frac{h^{11}}{11!}M_{11} \\ \|e_n\| &\leq \frac{h^{10}M_{11}}{11!\check{L}}(1 + h\check{L})^n - 1 \end{aligned}$$

untuk $h, \check{L} > 0$ berlaku,

$$(1 + h\check{L})^n \leq e^{nh\check{L}}, \quad h = \frac{x_n - x_0}{n}$$

sehingga,

$$\begin{aligned} \|e_n\| &\leq \frac{h^{10}M_{11}}{11!\check{L}}(e^{(x_n - x_0)\check{L}} - 1) \\ \lim_{h \rightarrow 0} \|e_n\| &\leq \lim_{h \rightarrow 0} \frac{h^{10}M_{11}}{11!\check{L}}(e^{(x_n - x_0)\check{L}} - 1) \\ \lim_{h \rightarrow 0} \|e_n\| &\leq 0 \\ \lim_{h \rightarrow 0} \|e_n\| &= 0 \end{aligned}$$

dimana $\lim_{h \rightarrow 0} \|e_n\| \leq 0$ nilainya selalu positif.

Sehingga diperoleh $\lim_{h \rightarrow 0} \|e_n\| = 0$, dengan demikian metode Runge-Kutta orde sepuluh adalah metode yang konvergen.

Terbukti.

Formula di atas dinamakan metode RK10A, sebab akibat dari lemma 4.1.1 dapat dicari juga formula metode Runge-Kutta orde sepuluh dengan tetapan c_1, c_2, \dots, c_{10} yang sama tapi menghasilkan koefisien matrik $a_{21}, a_{31}, a_{32}, \dots, a_{109}$

yang berbeda. Metode RK10A merupakan metode Runge-Kutta orde sepuluh dengan tetapan nilai $c_1 = 0$, $c_2 = \frac{1}{9}$, $c_3 = \frac{2}{9}$, $c_4 = \frac{3}{9}$, $c_5 = \frac{4}{9}$, $c_6 = \frac{5}{9}$, $c_7 = \frac{6}{9}$, $c_8 = \frac{7}{9}$, $c_9 = \frac{8}{9}$ dan $c_{10} = 1$. dengan koefisien matrik sedikit nol (*non sparse matrix*). Formula metode Runge-Kutta yang lain dinamakan metode RK10B yang memiliki tetapan memiliki c_1, c_2, \dots, c_{10} yang sama tapi koefisien matriknya memiliki banyak nol (*sparse matrix*). Sedangkan untuk pembuktiannya sama dengan pembuktian pada metode RK10A. Untuk lebih menjamin keakuratannya, maka peneliti menggunakan software MATLAB yang terdapat dalam lampiran. Berikut merupakan metode RK10B yang disajikan secara lengkap melalui tabel *Butcher Array*.

Corollary 4.2 Formula Metode Runge-Kutta Orde Sepuluh (RK10B)

Untuk ukuran langkah sepanjang h maka formula metode Runge-Kutta orde sepuluh adalah sebagai berikut:

$$y_{n+1} = y_n + \frac{h}{89600}(2857k_1 + 15741k_2 + 1080k_3 + 19344k_4 + 5778k_5 + 5778k_6 + 19344k_7 + 1080k_8 + 15741k_9 + 2857k_{10})$$

dengan,

$$\begin{aligned} k_1 &= f(x_n, y_n) \\ k_2 &= f\left(x_n + \frac{h}{9}, y_n + \frac{h}{9}k_1\right) \\ k_3 &= f\left(x_n + \frac{2}{9}h, y_n + \frac{h}{45}(-1156k_1 + 1166k_2)\right) \\ k_4 &= f\left(x_n + \frac{3}{9}h, y_n + \frac{h}{4836}(13645k_1 - 12033k_3)\right) \\ k_5 &= f\left(x_n + \frac{4}{9}h, y_n + \frac{h}{2889}(-54136 + 55420k_4)\right) \\ k_6 &= f\left(x_n + \frac{5}{9}h, y_n + \frac{h}{963}(20405k_1 - 19870k_5)\right) \\ k_7 &= f\left(x_n + \frac{6}{9}h, y_n + \frac{h}{1209}(-5476k_1 + 6282k_6)\right) \\ k_8 &= f\left(x_n + \frac{7}{9}h, y_n + \frac{h}{270}(10841k_1 - 10631k_7)\right) \end{aligned}$$

$$k_9 = f\left(x_n + \frac{8h}{9}, y_n + \frac{h}{5247}(-80k_1 + 4744k_8)\right)$$

$$k_{10} = f(x_n + h, y_n + hk_1)$$

Tabel 4.2: Matriks koefisien Runge-Kutta Orde Sepuluh (RK10B)

0	0									
$\frac{1}{9}$	$\frac{1}{9}$	0								
$\frac{2}{9}$	$-\frac{1156}{45}$	$\frac{1166}{45}$	0							
$\frac{3}{9}$	$\frac{13645}{4836}$	0	$-\frac{12033}{4836}$	0						
$\frac{4}{9}$	$-\frac{54136}{2889}$	0	0	$\frac{55420}{2889}$	0					
$\frac{5}{9}$	$\frac{20405}{963}$	0	0	0	$-\frac{19870}{963}$	0				
$\frac{6}{9}$	$-\frac{5476}{1209}$	0	0	0	0	$\frac{6282}{1209}$	0			
$\frac{7}{9}$	$\frac{10841}{270}$	0	0	0	0	0	$-\frac{10631}{270}$	0		
$\frac{8}{9}$	$-\frac{80}{5247}$	0	0	0	0	0	0	$\frac{4744}{5247}$	0	
1	1	0	0	0	0	0	0	0	0	0
	$\frac{2857}{89600}$	$\frac{15741}{89600}$	$\frac{1080}{89600}$	$\frac{19344}{89600}$	$\frac{5778}{89600}$	$\frac{5778}{89600}$	$\frac{19344}{89600}$	$\frac{1080}{89600}$	$\frac{15741}{89600}$	$\frac{2857}{89600}$

4.3 Pemrograman MATLAB

Tahap berikutnya dalam penelitian ini adalah tahap pemodelan, tahap formulasi numerik, dan tahap algoritma. Kemudian akan dilanjutkan ke tahap pem-

rograman dalam bahasa MATLAB dari metode Runge-Kutta orde sepuluh untuk menyelesaikan PDB non linier orde satu pada model penyebaran virus Avian Influenza.

4.3.1 Tahap Pemodelan

Pada pemodelan, peneliti mengambil model penyebaran virus Avian Influenza berupa sistem PDB non linier orde satu yang dikembangkan oleh Okosun, 2007. Model matematika tersebut dapat dilihat dalam persamaan (2.1) sampai (2.4). pengambilan model matematika tersebut disebabkan sistem persamaan tersebut merupakan PDB non linier yang sulit untuk diselesaikan secara analitik, sehingga cara yang tepat adalah dengan menggunakan metode Numerik, khususnya *one step method* Runge-Kutta orde sepuluh.

4.3.2 Tahap Formulasi Numerik

Formulasi numerik yang dimaksud ialah proses pengubahan simbol dari model matematika yang diambil menjadi simbol-simbol matematika yang mudah dibaca dan ditulis dalam bahasa MATLAB tanpa mengurangi maknanya. Formulasi numerik dari model penyebaran virus Avian Influenza ialah:

$$y1' = A * a + (1 - c) * D - f * y1 * (y2/A) - d * y1$$

$$y2' = f * y1 * (y2/A) - (d + m) * y2 + c * D$$

$$y3' = B * b - ((g * y3 * y2)/A) - e * y3 + p + y4$$

$$y4' = ((g * y3 * y2)/A) - (e + v + p) * y4$$

Keterangan:

$A = y1 + y2$ = Jumlah unggas

$B = y3 + y4$ = Jumlah manusia

D = Jumlah burung migrasi

$y1$ = Jumlah suspect unggas

$y2$ = Jumlah unggas yang terinfeksi

$y3$ = Jumlah suspect manusia

- y_4 = Jumlah manusia yang terinfeksi
- a = Rata-rata tingkat kelahiran unggas
- b = Rata-rata tingkat kelahiran manusia
- c = Peluang infeksi pada unggas migrasi
- d = Tingkat kematian alami unggas
- e = Tingkat kematian alami manusia
- f = Tingkat transmisi infeksi dari unggas ke unggas
- g = Tingkat transmisi infeksi dari unggas ke manusia
- m = Tingkat kematian unggas karena infeksi flu burung
- v = Tingkat kematian manusia karena infeksi flu burung
- p = Tingkat kesembuhan manusia (per hari)

Tabel 4.3: Interpretasi Parameter

Parameter	Deskripsi	Nilai Estimasi
A	Jumlah total dari unggas di lokasi	1.000
B	Jumlah total dari manusia di lokasi	variabel
a	Rata-rata tingkat kelahiran unggas	0,03
b	Rata-rata tingkat kelahiran manusia	0,001
c	Peluang infeksi pada unggas migrasi	0,01
D	Jumlah total unggas migrasi (per hari)	10
d	Tingkat kematian alami unggas	$1/(365 \times 2)$
e	Tingkat kematian alami manusia	$1/(365 \times 75)$
f	Tingkat transmisi infeksi dari unggas ke unggas	0,9
g	Tingkat transmisi infeksi dari unggas ke manusia	0,1
m	Tingkat kematian unggas karena infeksi flu burung	0,99
v	Tingkat kematian manusia karena infeksi flu burung	0,009
p	Tingkat kesembuhan manusia (per hari)	$1/7$

4.3.3 Pola Algoritma Metode Runge-Kutta Orde Sepuluh

Setelah tahap formulasi numerik, tahap selanjutnya adalah algoritma. Penyusunan algoritma merupakan langkah awal membuat program. Adapun susunan

algoritma dan format *programming* dari metode Runge-Kutta orde sepuluh adalah sebagai berikut.

• **Pola Algoritma Efektivitas Metode Runge Kutta Orde Sepuluh (RK10A)**

INPUT : nilai awal t_0, y_0 , ukuran langkah (h), iterasi (i)

OUTPUT : Nilai aproksimasi atau hampiran (y_{n+1}) bagi solusi ($y(x_{n+1})$)

di $x_{j+1} = x_j + (j + 1)h$, dengan $j = 0, 1, 2, \dots, n$

Step 1. For $i = 1, 2, \dots, n$

Step 2. Set $x_i = a_i + ih$

$$\begin{aligned}
 k_1 &= f(x_n, y_n) \\
 k_2 &= f\left(x_n + \frac{h}{9}, y_n + \frac{h}{9}k_1\right) \\
 k_3 &= f\left(x_n + \frac{2}{9}h, y_n + \frac{h}{360}(-1617k_1 + 1697k_2)\right) \\
 k_4 &= f\left(x_n + \frac{3}{9}h, y_n + \frac{h}{1080}(2501k_1 + 1000k_2 - 3141k_3)\right) \\
 k_5 &= f\left(x_n + \frac{4}{9}h, y_n + \frac{h}{8667}(-154197k_1 - 8620k_2 - 8640k_3 + 175309k_4)\right) \\
 k_6 &= f\left(x_n + \frac{5}{9}h, y_n + \frac{h}{5778}(99353k_1 - 5700k_2 - 5740k_3 + 5620k_4 - 90323k_5)\right) \\
 k_7 &= f\left(x_n + \frac{6}{9}h, y_n + \frac{h}{5580}(-16343k_1 - 520k_2 + 5540k_3 - 5420k_4 - 5560k_5 \right. \\
 &\quad \left. + 26023k_6)\right) \\
 k_8 &= f\left(x_n + \frac{7}{9}h, y_n + \frac{h}{450}(11546k_1 + 445k_2 - 440k_3 + 420k_4 + 320k_5 \right. \\
 &\quad \left. + 420k_6 - 12361k_7)\right) \\
 k_9 &= f\left(x_n + \frac{8}{9}h, y_n + \frac{h}{4770}(-1281k_1 + 4720k_2 + 390k_3 + 1280k_4 - 2860k_5 \right. \\
 &\quad \left. + 2700k_6 - 4762k_7 + 4053k_8)\right) \\
 k_{10} &= f\left(x_n + h, y_n + \frac{h}{1800}(-2500k_1 + 1000k_2 + 120k_3 + 1200k_4 - 600k_5 + 240k_6 \right. \\
 &\quad \left. + 1800k_7 + 540k_8)\right) \\
 y_{n+1} &= y_n + \frac{h}{89600}(2857k_1 + 15741k_2 + 1080k_3 + 19344k_4 + 5778k_5 \\
 &\quad + 5778k_6 + 19344k_7 + 1080k_8 + 15741k_9 + 2857k_{10})
 \end{aligned}$$

OUTPUT y_{j+1}

Step 3. set error = norm ($y_{j+1} - y_j$)

Step 4. set perbaharui $y_j = y_{j+1}$

Step 5. set $x_{j+1} = x_j + h$

Step 6. OUTPUT y_{j+1}

end

• **Pola Algoritma Efisiensi Metode Runge Kutta Orde Sepuluh (RK10B)**

INPUT : nilai awal t_0, y_0 , ukuran langkah (h), toleransi (e)

OUTPUT : Nilai aproksimasi atau hampiran (y_{n+1}) bagi solusi ($y(x_{n+1})$)

di $x_{j+1} = x_j + (j + 1)h$, dengan $j = 0, 1, 2, \dots, n$

step 1. Set while error do step 2-6

step 2.

$$\begin{aligned}
 k_1 &= f(x_n, y_n) \\
 k_2 &= f\left(x_n + \frac{h}{9}, y_n + \frac{h}{9}k_1\right) \\
 k_3 &= f\left(x_n + \frac{2h}{9}, y_n + \frac{h}{45}(-1156k_1 + 1166k_2)\right) \\
 k_4 &= f\left(x_n + \frac{3h}{9}, y_n + \frac{h}{4836}(13645k_1 - 12033k_3)\right) \\
 k_5 &= f\left(x_n + \frac{4h}{9}, y_n + \frac{h}{2889}(-54136 + 55420k_4)\right) \\
 k_6 &= f\left(x_n + \frac{5h}{9}, y_n + \frac{h}{963}(20405k_1 - 19870k_5)\right) \\
 k_7 &= f\left(x_n + \frac{6h}{9}, y_n + \frac{h}{1209}(-5476k_1 + 6282k_6)\right) \\
 k_8 &= f\left(x_n + \frac{7h}{9}, y_n + \frac{h}{270}(10841k_1 - 10631k_7)\right) \\
 k_9 &= f\left(x_n + \frac{8h}{9}, y_n + \frac{h}{5247}(-80k_1 + 4744k_8)\right) \\
 k_{10} &= f(x_n + h, y_n + hk_1) \\
 y_{n+1} &= y_n + \frac{h}{89600}(2857k_1 + 15741k_2 + 1080k_3 + 19344k_4 + 5778k_5 \\
 &\quad + 5778k_6 + 19344k_7 + 1080k_8 + 15741k_9 + 2857k_{10})
 \end{aligned}$$

OUTPUT y_{j+1}
 Step 3. set error = norm ($y_{j+1} - y_j$)
 Step 4. set perbaharui $y_j = y_{j+1}$
 Step 5. set $x_{j+1} = x_j + h$
 Step 6. OUTPUT y_{j+1}
 end

4.3.4 Format Pemrograman Metode Runge-Kutta Orde Sepuluh

- **Format Programming Efektifitas**

Format *programming* yang dikembangkan dari Metode Runge-Kutta orde sepuluh adalah sebagai berikut: (ada pada lampiran A).

- **Format Programming Efisiensi**

Pada dasarnya penulisan format pemrograman efisiensi metode RK10A sama dengan penulisan format pemrograman efektifitas metode RK10A. Hal yang berbeda adalah jenis data yang diinput. Jika pada format pemrograman efektifitas metode RK10A yang diinput adalah banyaknya iterasi (i), maka data yang diinput pada format pemrograman efisiensi metode RK10A adalah toleransi (e). Selain itu, setelah mendefinisikan nilai tetapan yang diketahui:

```
tol=input('Toleransi (e)='); h=input('Ukuran langkah (h)=');
y10=input('jumlah suspect unggas (y1)='); y20=input('Jumlah unggas
yang terinfeksi (y2)='); y30=input('Jumlah suspect manusia (y3)=');
y40=input('Jumlah manusia yang terinfeksi (y4)=');
```

```
A=y10+y20; B=y30+y40; a=0.03; b=0.001; c=0.01; d=1/(365*2);
e=1/(365*75); f=0.9; g=0.1; m=0.99; v=0.009; p=1/7; D=10;
```

```
t(:,1)=[0;0;0;0]; y(:,1)=[y10;y20;y30;y40]; error1=norm(y(:,1),inf);
j=1; errvec1=[]; fprintf('\n itn error');
```

```
fprintf('\-----> n '); while error1>=tol
    fprintf('\n%3.0f %9.3e',j,error1);
```

Kemudian dilanjutkan dengan menulis format k , $y(:, j + 1)$, dan seterusnya sebagaimana format efektivitas metode RK10A.

4.4 Efektivitas dan Efisiensi Metode Runge-Kutta Orde Sepuluh

Tahap selanjutnya adalah menjalankan *programming* dalam MATLAB dengan data yang telah ditentukan. Dari hasil eksekusi, dihasilkan grafik konvergensi yang konvergen, data output berupa grafik dan waktu tempuh serta banyaknya iterasi untuk menganalisis tingkat efisiensi metode Runge-Kutta orde sepuluh. Sedangkan untuk menganalisis tingkat efektivitas metode Runge Kutta orde sepuluh, diambil data output berupa grafik, data error dan banyaknya iterasi. Dari semua data yang diperoleh, maka dapat dilakukan tahap evaluasi, yaitu menginterpretasikan dan menganalisis secara deskriptif berdasarkan fakta-fakta yang ada untuk penarikan suatu kesimpulan.

Sebelum membahas hasil *programming*, tingkat efektivitas dan tingkat efisiensi metode Runge Kutta orde sepuluh, terlebih dahulu dibahas mengenai simulasi pemodelan yang digunakan dalam penelitian.

4.4.1 Simulasi Pemodelan

Setelah menyusun *format programming* dari metode Runge Kutta orde sepuluh, tahap berikutnya adalah *tahap operasional* yakni menjalankan *format programming* dengan data parameter persamaan yang ada yang berasal dari penelitian dalam Jurnal Internasional Medwell yang ditulis oleh Okosun (2007:10). Selanjutnya diadakan tahap evaluasi yakni menginterpretasi dan menganalisis hasil *programming* untuk mencapai suatu kesimpulan.

Guswai (2007:2) mengartikan *efektif* sebagai keadaan mampu mencapai tujuan dan sasaran, dan karena tujuan menggunakan teknis numerik ini adalah untuk mendapatkan solusi hampiran terdekat dalam menyelesaikan suatu masalah maka dalam menentukan tingkat efektivitas metode akan dibandingkan seberapa

dekat metode tersebut terhadap solusi sebenarnya. Tingkat seberapa dekat solusi tersebut terhadap solusi eksak dapat dilihat dari nilai errornya dengan menggunakan Metode Runge-Kutta orde sepuluh dalam menyelesaikan model penyebaran virus *Avian Influenza*.

Model penyebaran virus flu burung yang akan dianalisis adalah model SIRS, yakni model yang mengkategorikan suatu kasus menjadi kategori *susceptible*-rentan (S), *infectious*-terinfeksi (I), dan *recovered but susceptible*-sembuh namun rentan (RS). Beberapa dari kasus hanya membagi model menjadi dua katagori saja yaitu *susceptible* - rentan (S), dan *infectious* - terinfeksi (I).

Simulasi kasus didasarkan terhadap populasi unggas dan manusia dalam model penyebaran virus Avian Influenza yaitu populasi burung (unggas) yang rentan - susceptible birds ($S_B(t)$), burung (unggas) yang terinfeksi - infections birds ($I_B(t)$), manusia yang rentan - susceptible human ($S_H(t)$), dan manusia yang terinfeksi - infeksi human ($I_H(t)$).

Format program metode runge-kutta orde sepuluh dijalankan terhadap kasus penyebaran virus Avian Influenza di Kabupaten Jember, Jawa Timur. Data simulasi diambil berdasarkan keterangan dalam situs resmi yang dipublikasikan oleh Antara Jatim (awal Januari 2015). Dalam situs tersebut disebutkan bahwa 165 unggas dan 7 orang dikecamatan Silo kabupaten Jember terjangkit virus Avian Influenza. Untuk melengkapi data simulasi peneliti mengansumsikan data banyaknya unggas dan manusia yang mengalami suspect virus Avian Influenza dikecamatan Silo sebesar 300 ekor dan 60 orang.

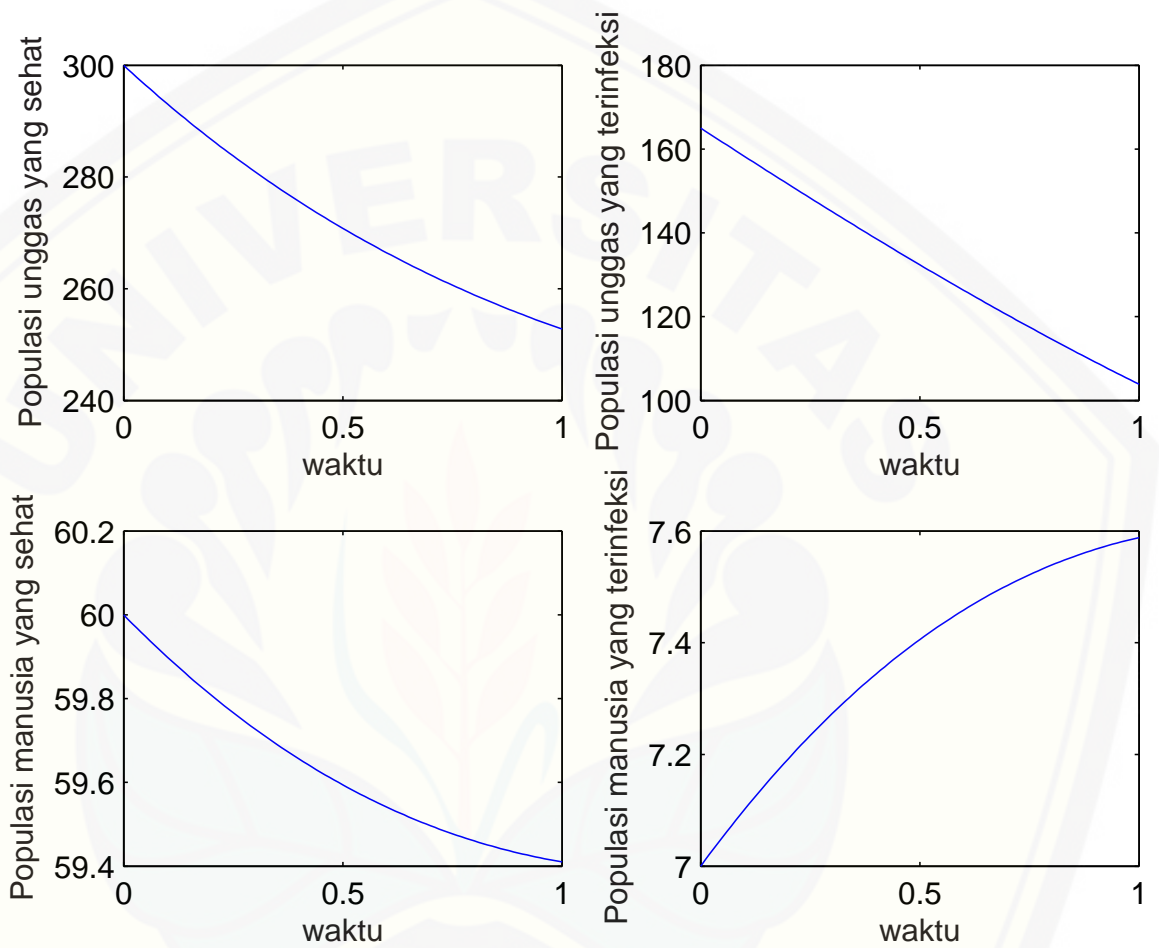
Keempat jenis data tersebut di inputkan bersama nilai toleransi dan ukuran langkah 0.01 pada listing program simulasi untuk dieksekusi. Ukuran langkah 0.01 memiliki arti bahwa pengamatan dilakukan setiap hari dengan peningkatan seperseratus setiap waktunya sejak dimasukkannya nilai koefisien atau variabel populasi manusia dan unggas. Peneliti menggunakan nilai toleransi 10^{-1} , 10^{-2} , 10^{-3} , 10^{-4} dan 10^{-5} sebagai jaminan pembuktian tingkat efektivitas dan akurasi metode Runge-kutta orde sepuluh terhadap model penyebaran virus Avian Influenza. Toleransi tersebut dalam teknik numerik sudah cukup menggambarkan

akurasi yang baik karena kesalahan yang terjadi sudah cukup kecil. Data nilai awal dan parameter digunakan untuk menyelesaikan sistem persamaan model tersebut dengan menggunakan metode Runge-Kutta orde sepuluh. penyelesaian secara numerik juga memerlukan ukuran langkah (h) dan toleransi (e) yang ditetapkan untuk menentukan efisiensi serta jumlah iterasi (i) untuk menentukan efektivitas pada metode tersebut. Sehingga tingkat efisiensi dan efektivitas dari metode tersebut dapat dianalisis.

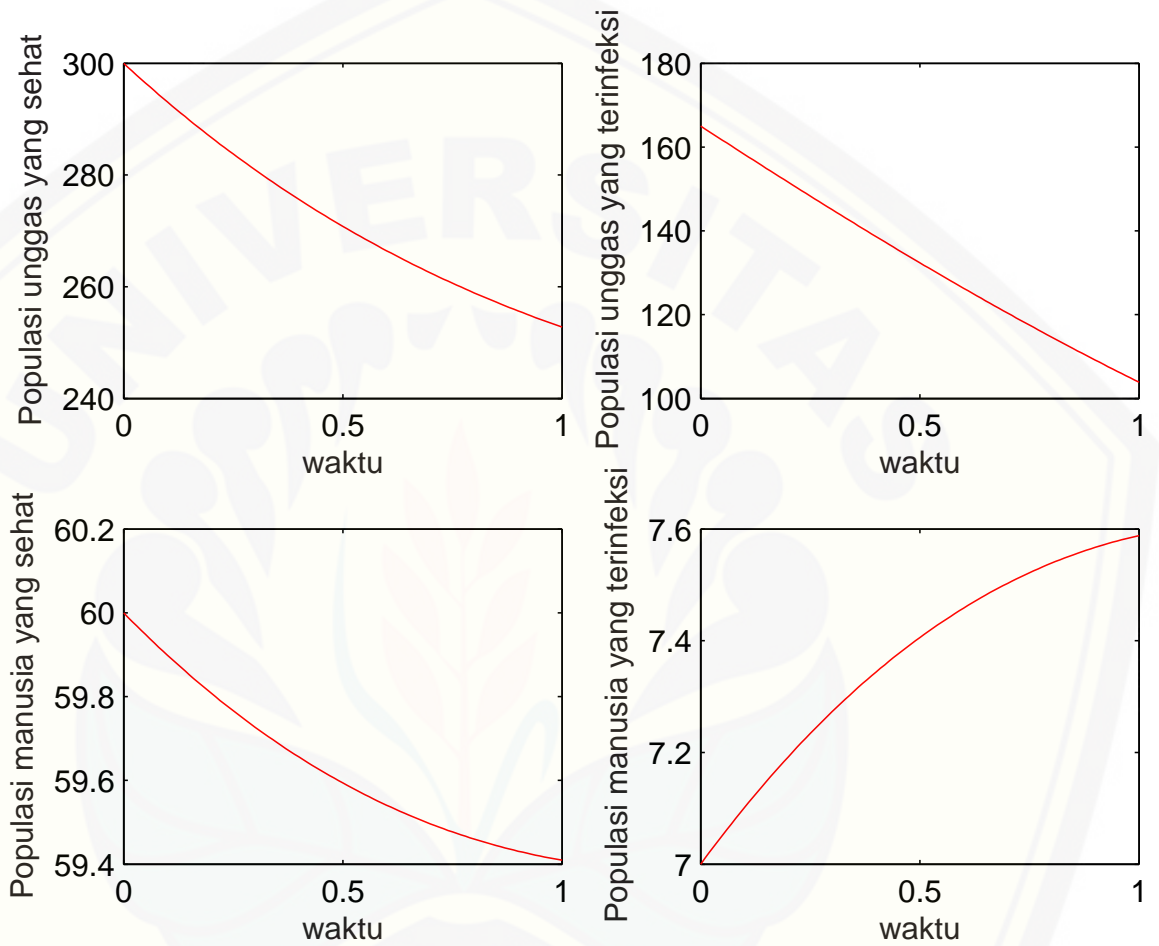
4.4.2 Hasil Komputasi Metode Runge-Kutta Orde Sepuluh dengan MATLAB

Pada penelitian ini penelitian menggunakan nilai iterasi (i) 50, 100, 250, 500, 1.000, 2.500, 5.000, 10.000, 25.000, 50.000, 100.000 dan 200.000 untuk menentukan efektivitas metode Runge-Kutta orde sepuluh. Pengambilan nilai iterasi tersebut telah memiliki rentang yang cukup jauh sehingga dapat mengetahui tingkat akurasi solusi y_1 , y_2 , y_3 dan y_4 dari metode tersebut. Grafik yang dihasilkan memiliki bentuk yang mirip antara RK10A dan RK10B pada setiap iterasi, baik pada populasi manusia ataupun populasi unggas terhadap waktu (t). Gambar (4.1) sampai dengan gambar (4.10) adalah sebagian visualisasi yang dihasilkan berdasarkan iterasi yang telah ditetapkan. Visualisasi yang ditampilkan adalah grafik efektivitas dan efisiensi dari metode Range-Kutta orde sepuluh yakni RK10A dan RK10B. Berikut adalah tampilan visualisasi efektivitas dan efisiensi hasil eksekusi dari metode Runge-Kutta orde sepuluh dengan ukuran langkah ($h= 0.01$).

Gambar (4.1) dan (4.2) merupakan grafik hasil eksekusi metode Runge-Kutta orde sepuluh RK10A dan RK10B pada iterasi 100 dan $h=0.01$, hasil eksekusi grafik antara RK10A dengan RK10B mirip. Analisis grafik waktu sama dengan 0 (nol) pada grafik artinya waktu dimulainya pengamatan terhadap populasi manusia dan unggas. Pada grafik tersebut terlihat bahwa populasi unggas yang sehat terhadap waktu t dapat diketahui mengalami penurunan dalam waktu sekitar satu hari sekitar 50 unggas. Hal ini juga ditunjukkan pada populasi unggas yang terinfeksi mengalami penurunan juga sekitar 60 unggas dalam waktu sekitar



Gambar 4.1: Grafik eksekusi RK10A dengan iterasi 100 pada populasi manusia dan unggas terhadap waktu (t) dengan $h=0.01$

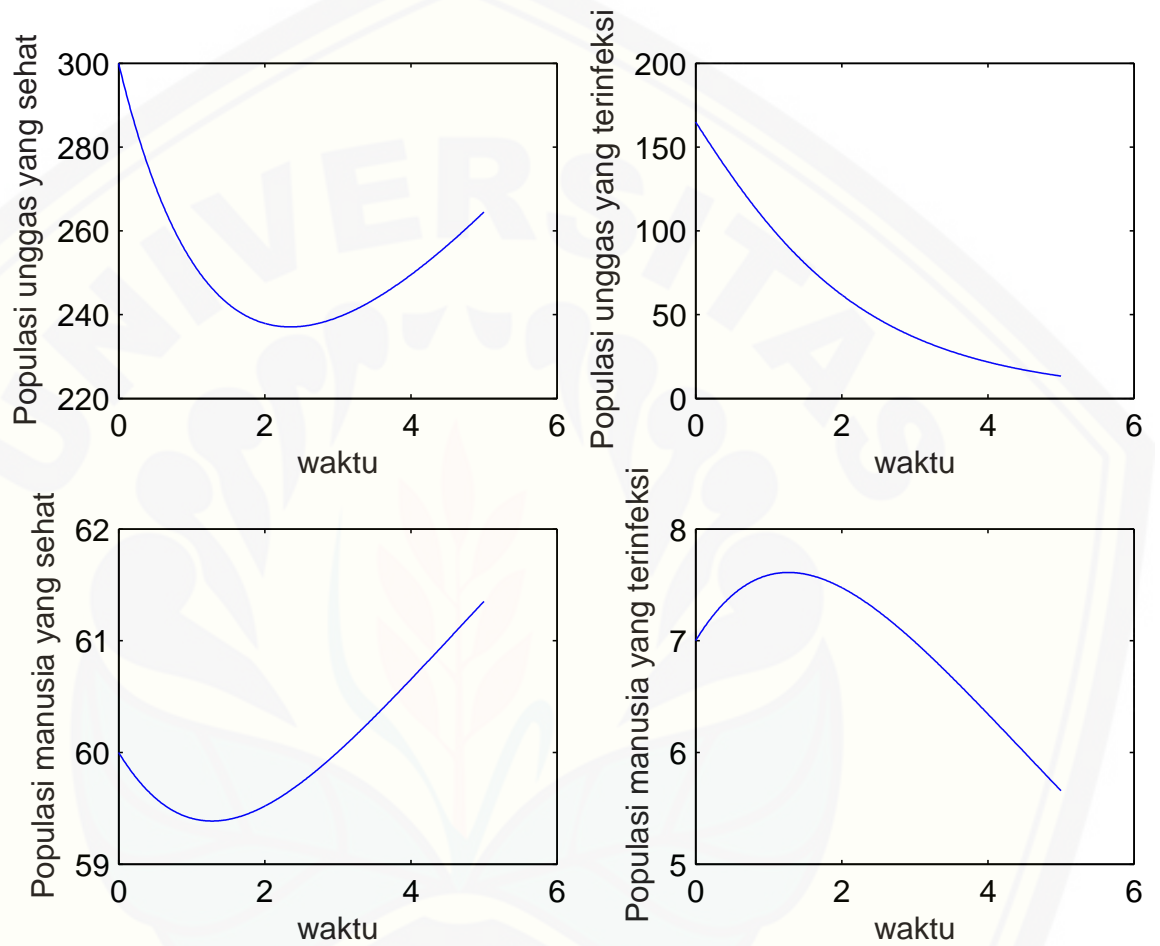


Gambar 4.2: Grafik eksekusi RK10B dengan iterasi 100 pada populasi manusia dan unggas terhadap waktu (t) dengan $h=0.01$

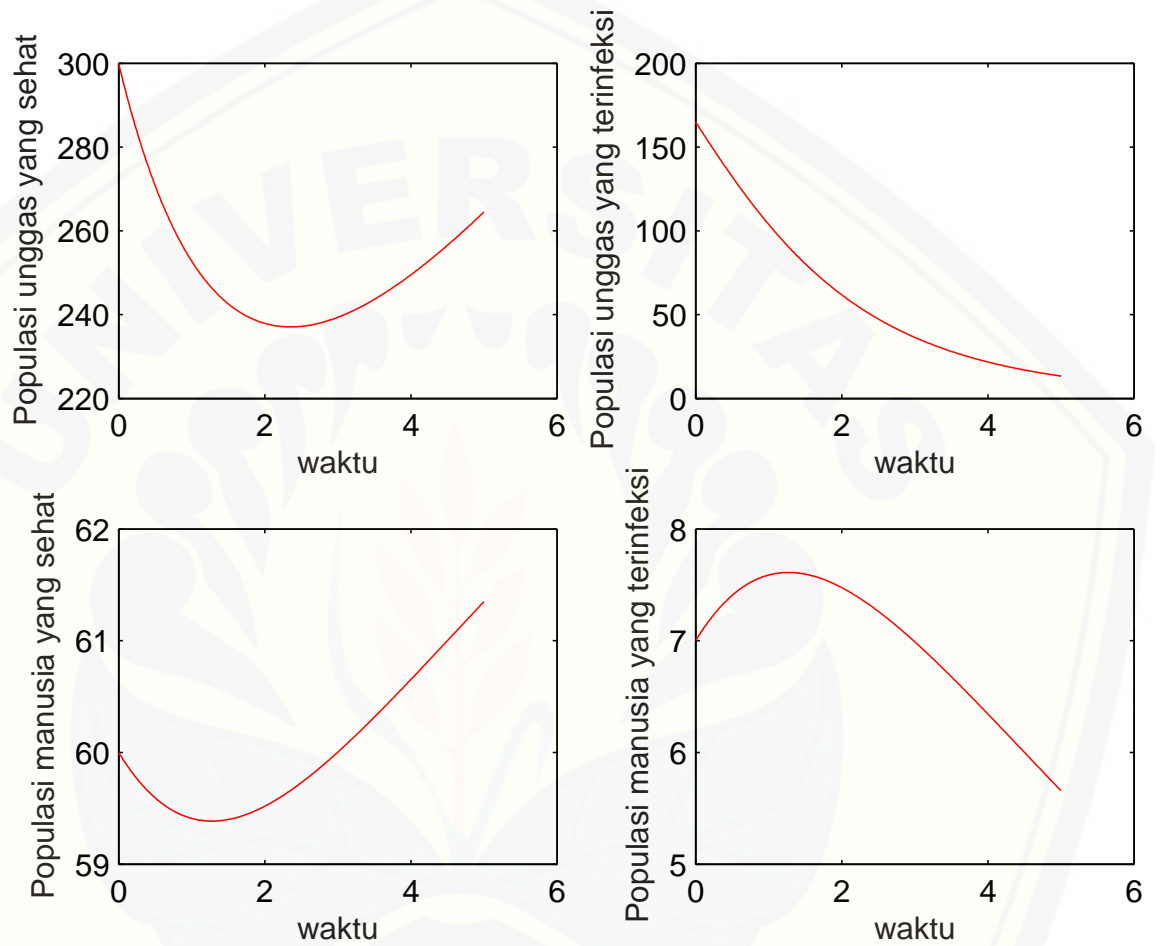
satu hari. Untuk populasi manusia yang sehat juga mengalami penurunan dalam waktu sekitar satu hari sekitar 1 orang, lain halnya dengan populasi manusia yang terinfeksi mengalami kenaikan dalam waktu satu hari sekitar satu orang. Untuk diskripsi grafik RK10B sama dengan RK10A. Pada kasus Avian Influenza (flu burung), kondisi *recovered but susceptible* - sembuh namun rentan (RS) untuk burung (unggas) memiliki kemungkinan yang kecil sekali bahkan tidak ada karena virus ini sangat mematikan untuk burung (unggas). Sedangkan untuk kategori manusia yang sembuh (*recovered*) adalah faktor atau unsur yang kembali lagi menjadi kategori rentan (*susceptible*) disebabkan kemungkinan untuk terinfeksi kembali setelah sembuh dari Avian Influenza, walaupun tentu saja terdapat kemungkinan timbul imun alami dalam tubuh manusia, namun imun yang terbentuk kemungkinan besar hanya bersifat sementara, sehingga dihasilkan visualisasi grafik yang terlihat pada gambar 4.1 dan 4.2.

Pada Gambar (4.3) dan (4.4) merupakan hasil eksekusi Runge-Kutta orde sepuluh pada iterasi 500 dengan $h=0.01$. Hasil eksekusi grafik antara RK10A dengan RK10B pada iterasi ini adalah mirip. Pada grafik tersebut terlihat bahwa populasi unggas yang sehat terhadap waktu t dapat diketahui mengalami penurunan dalam waktu sekitar sampai dua hari dengan penurunan populasi unggas yang sehat sekitar 65 unggas dan di hari kedua sampai hari ke lima populasi unggas mengalami kenaikan sekitar 30 unggas. Hal ini juga ditunjukkan pada populasi unggas yang terinfeksi mengalami penurunan sekitar 150 unggas sampai waktu hari kelima. Untuk populasi manusia yang sehat juga mengalami penurunan sampai waktu hari pertama dan setelah hari pertama sampai hari kelima mengalami kenaikan populasi manusia yang sehat sekitar dua orang, lain halnya dengan populasi manusia yang terinfeksi mengalami kenaikan dalam waktu satu hari di hari pertama dan selanjutnya sampai hari kelima mengalami penurunan sekitar dua orang. Untuk diskripsi grafik RK10B sama dengan RK10A.

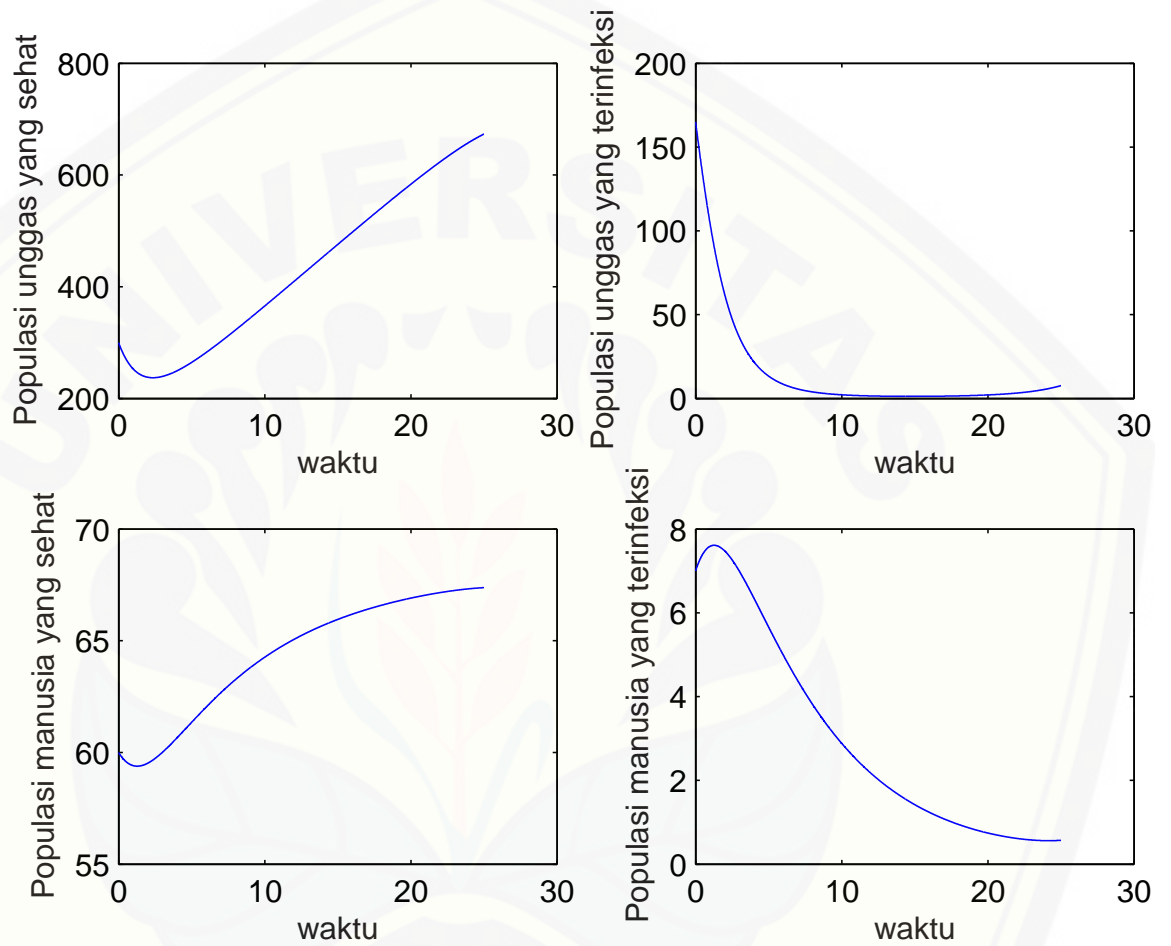
Visualisasi grafik RK10A dan RK10B hasil eksekusi Runge-Kutta orde sepuluh pada iterasi 2500 untuk *non sparse matrix* dan *sparse matrix* dengan ukuran langkah $h = 0,01$ menghasilkan penafsiran bahwa hubungan populasi manusia



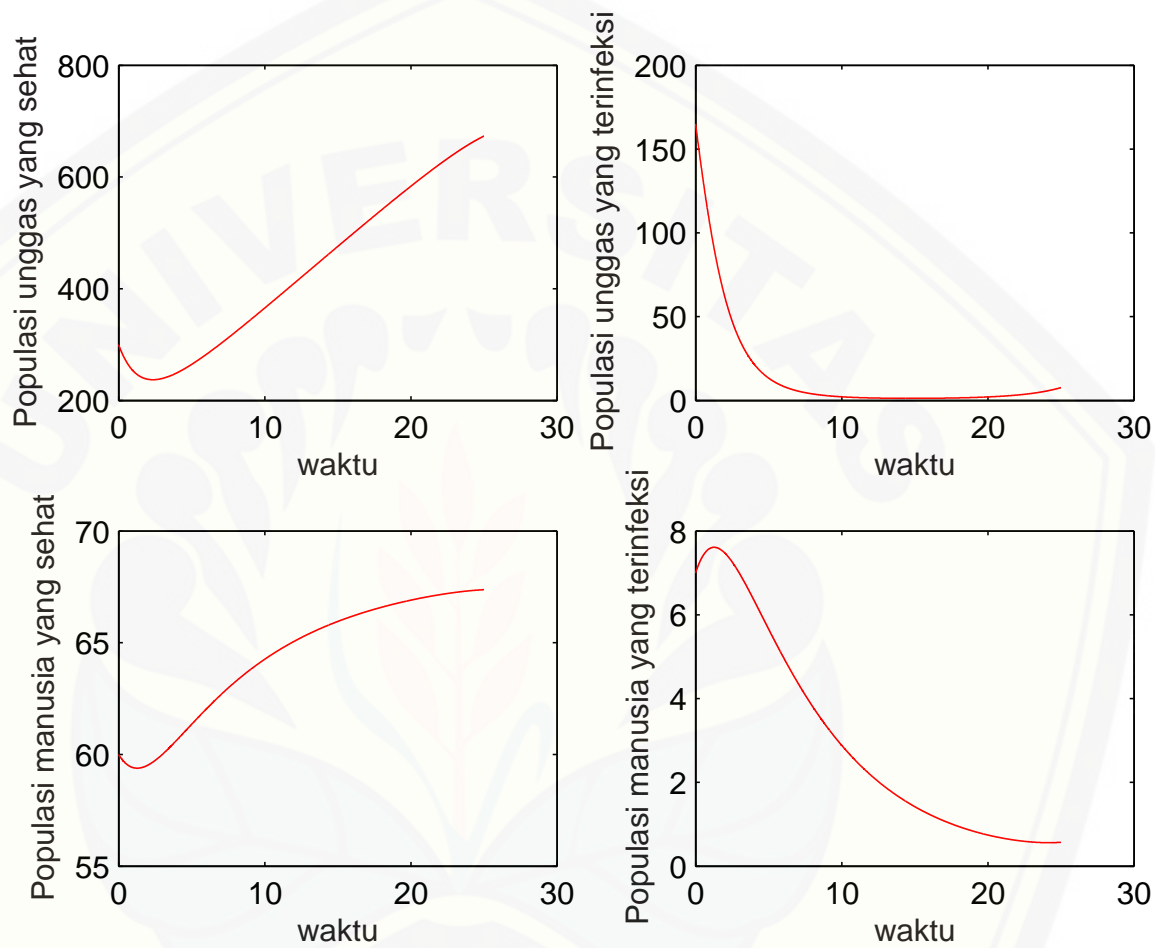
Gambar 4.3: Grafik eksekusi RK10A dengan iterasi 500 pada populasi manusia dan unggas terhadap waktu (t) dengan $h=0.01$



Gambar 4.4: Grafik eksekusi RK10B dengan iterasi 500 pada populasi manusia dan unggas terhadap waktu (t) dengan $h=0.01$



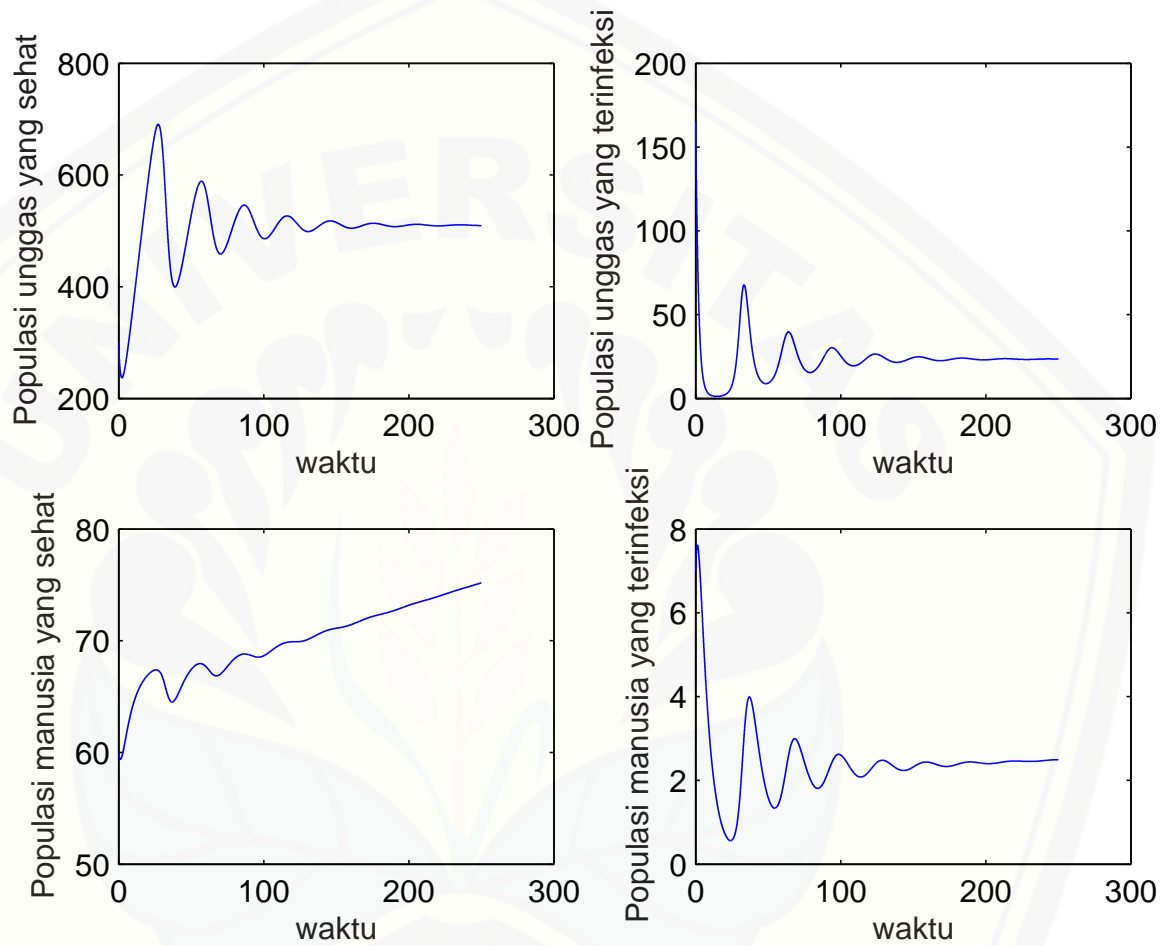
Gambar 4.5: Grafik eksekusi RK10A dengan iterasi 2500 pada populasi manusia dan unggas terhadap waktu (t) dengan $h=0.01$



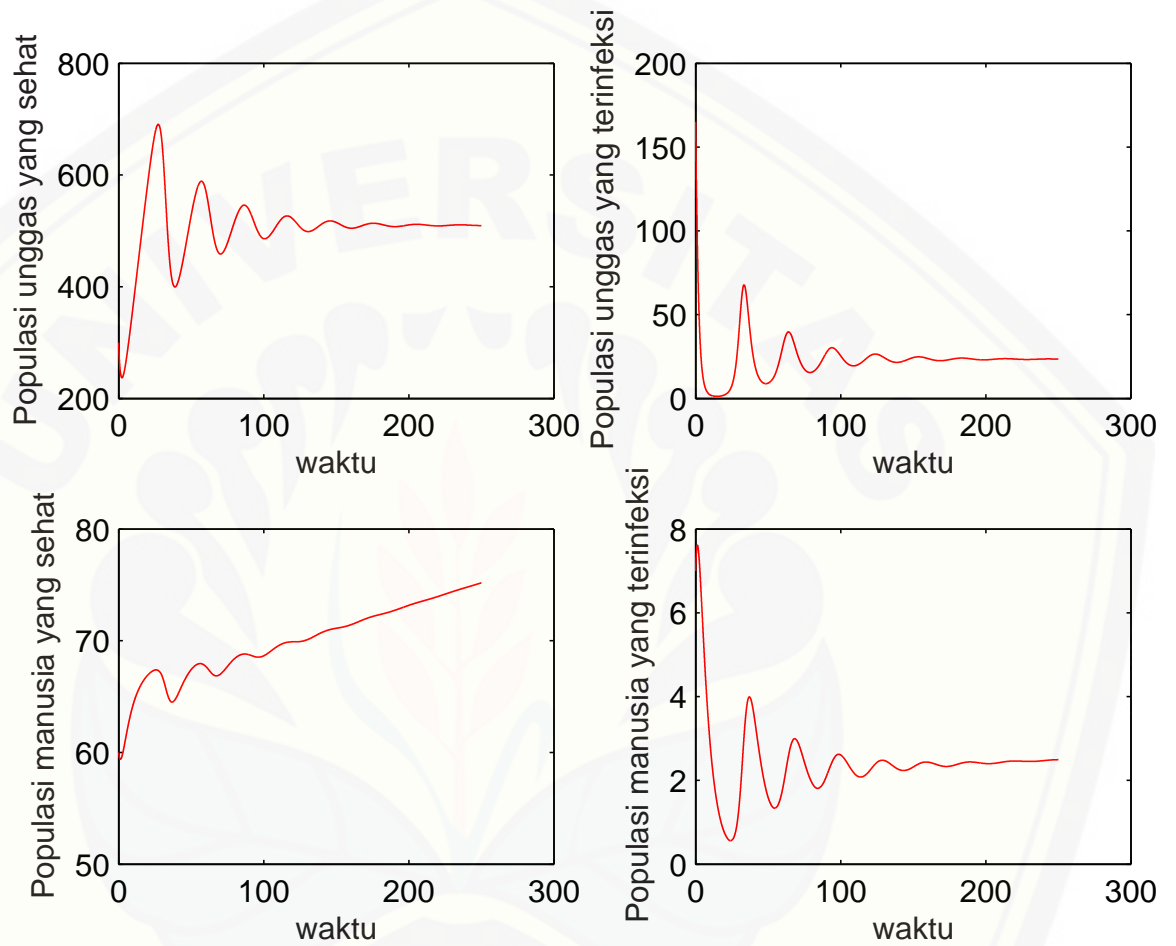
Gambar 4.6: Grafik eksekusi RK10B dengan iterasi 2500 pada populasi manusia dan unggas terhadap waktu (t) dengan $h=0.01$

yang sehat dengan waktu (t) adalah pada saat hari pertama sampai hari kedua populasi manusia yang sehat mengalami sedikit penurunan, akan tetapi setelah hari ketiga sampai hari ke 25, populasi manusia yang sehat mengalami kenaikan sekitar 7 orang. pada grafik hubungan populasi manusia yang terinfeksi dari waktu mula-mula sampai hari ke 2 mengalami sedikit kenaikan dan selanjutnya sampai hari ke dua puluh lima mengalami penurunan yang drastis sekitar 7 orang. Pada populasi unggas yang sehat saat awal sampai hari ke 2 mengalami penurunan, akan tetapi setelah setelah hari ke 2 sampai hari ke 25 mengalami kenaikan yang berarti sekitar 400 ekor unggas, sedangkan pada populasi unggas yang terinfeksi mengalami penurunan pada waktu mula-mula sampai hari ke 10 sekitar 160 ekor, pada hari 10 sampai hari ke 20 terlihat konstan dititik nol dan setelah hari ke 20 sampai hari ke 25 mulai landai naik kembali. Eksekusi Runge-Kutta orde sepuluh pada iterasi 2500 dengan $h=0.01$ ini bisa di lihat visualisasi grafiknya pada gambar (4.5) dan (4.6).

Visualisasi hasil eksekusi programming metode Runge-Kutta orde sepuluh RK10A dan RK10B pada iterasi 25000 dengan $h=0.01$ adalah mirip dan ini bisa di lihat hasil visualisasi grafiknya pada gambar (4.7) dan (4.8), setelah memasukkan nilai langkah 0,01 dan nilai parameter model virus Avian Influenza. Hasil eksekusi Deskripsi grafik tersebut menjelaskan bahwa dari hari pertama sampai hari ke 20 populasi unggas yang sehat mengalami peningkatan sekitar 500 unggas, selanjutnya dari hari ke 20 mengalami fluktuasi hingga stabil pada hari ke 150 yaitu sekitar 500 unggas yang sehat. Begitu juga populasi unggas yang terinfeksi, dinamika populasi banyaknya unggas yang terinfeksi mengalami penurunan dari mula-mula sampai hari ke 10 sekitar 165 unggas, dari hari ke 10 sampai hari ke 150 mengalami fluktuasi dan setelah hari ke 150 cenderung stabil dengan banyak populai unggas sekitar 25 unggas. populasi manusia yang sehat dari hari pertama sampai hari ke 150 mengalami peningkatan yang flutuasi dan akhirnya stabil meningkat setelah hari ke 150 sampai hari ke 250. Untuk populasi manusia yang terinfeksi virus Avian Influenza cenderung menurun dari hari pertama sampai hari ke 20 sekitar 7 orang, kemudian mengalami fluktuasi sampai hari ke 150

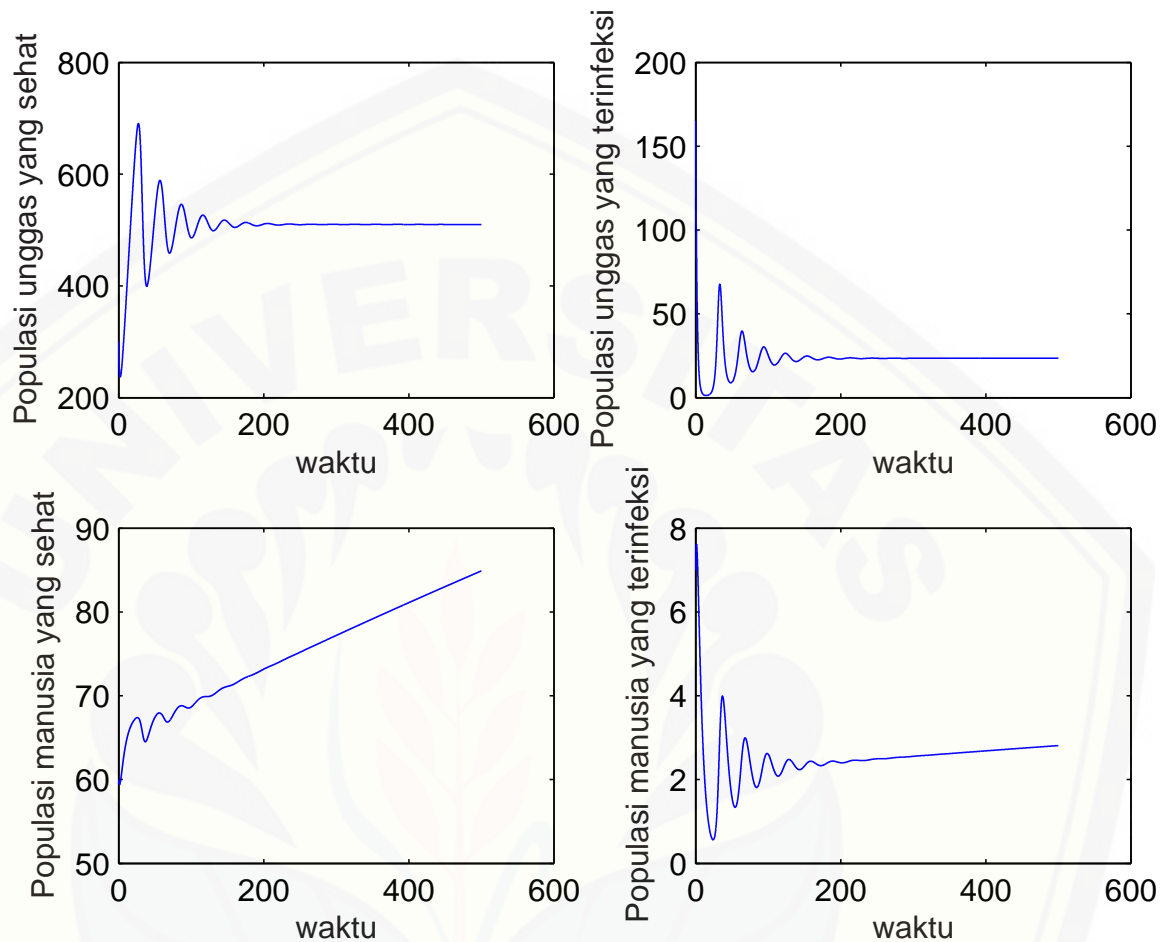


Gambar 4.7: Grafik eksekusi RK10A dengan iterasi 25000 pada populasi manusia dan unggas terhadap waktu (t) dengan $h=0.01$



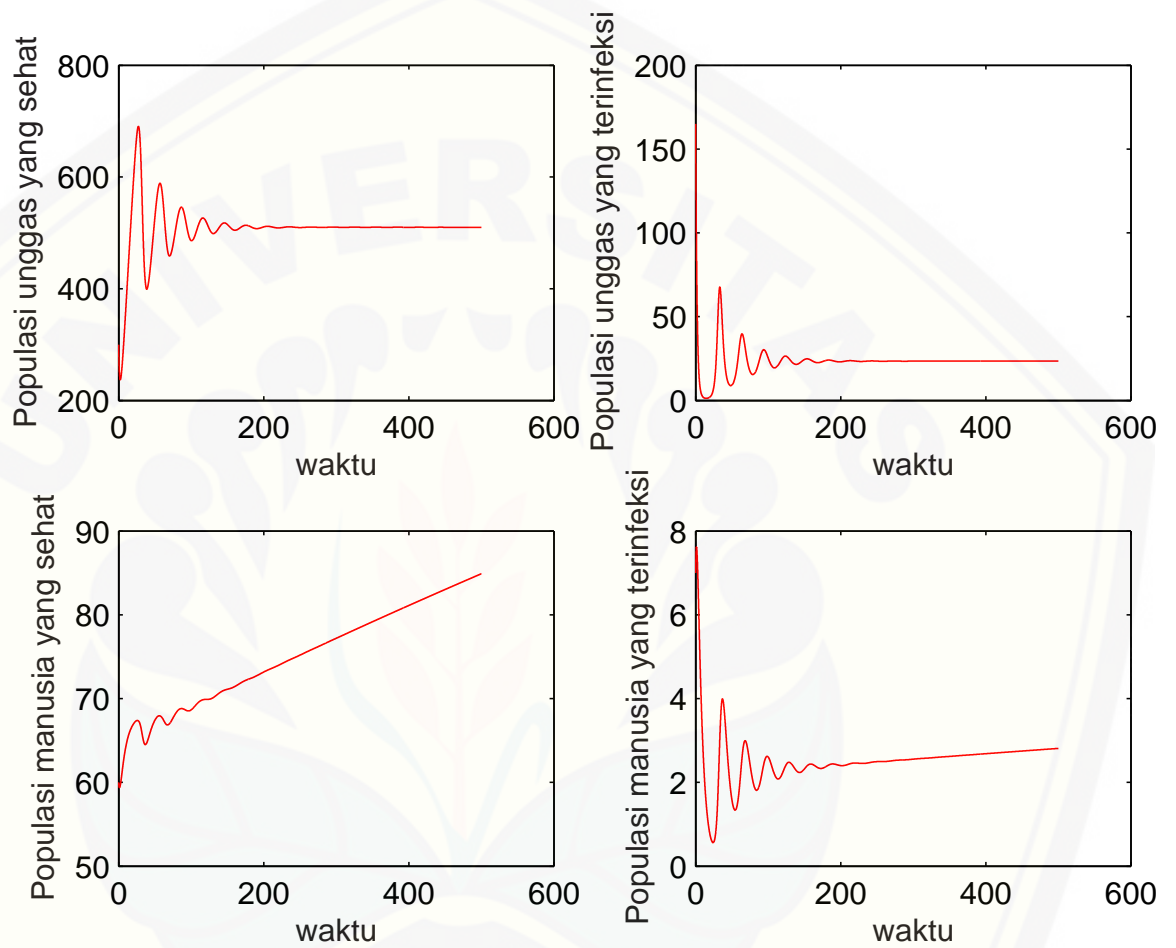
Gambar 4.8: Grafik eksekusi RK10B dengan iterasi 25000 pada populasi manusia dan unggas terhadap waktu (t) dengan $h=0.01$

dan mengalami kestabilan setelah hari ke 150 sampai hari ke 250 dengan populasi manusia yang terinfeksi pada saat stabil sekitar 2 orang.



Gambar 4.9: Grafik eksekusi RK10A dengan iterasi 50000 pada populasi manusia dan unggas terhadap waktu (t) dengan $h=0.01$

Pada gambar (4.9) dan (4.10), merupakan visualisasi hasil eksekusi programming metode Runge-Kutta orde sepuluh RK10A dan RK10B pada iterasi 50000 dengan $h=0.01$. Hasil eksekusi Deskripsi grafik tersebut menjelaskan bahwa populasi manusia yang sehat dari awal sampai hari ke 150 mengalami flutuasi meningkat dan akhirnya stabil meningkat setelah hari ke 150 sampai hari ke 500. Untuk populasi manusia yang terinfeksi virus Avian Influenza cenderung menu-

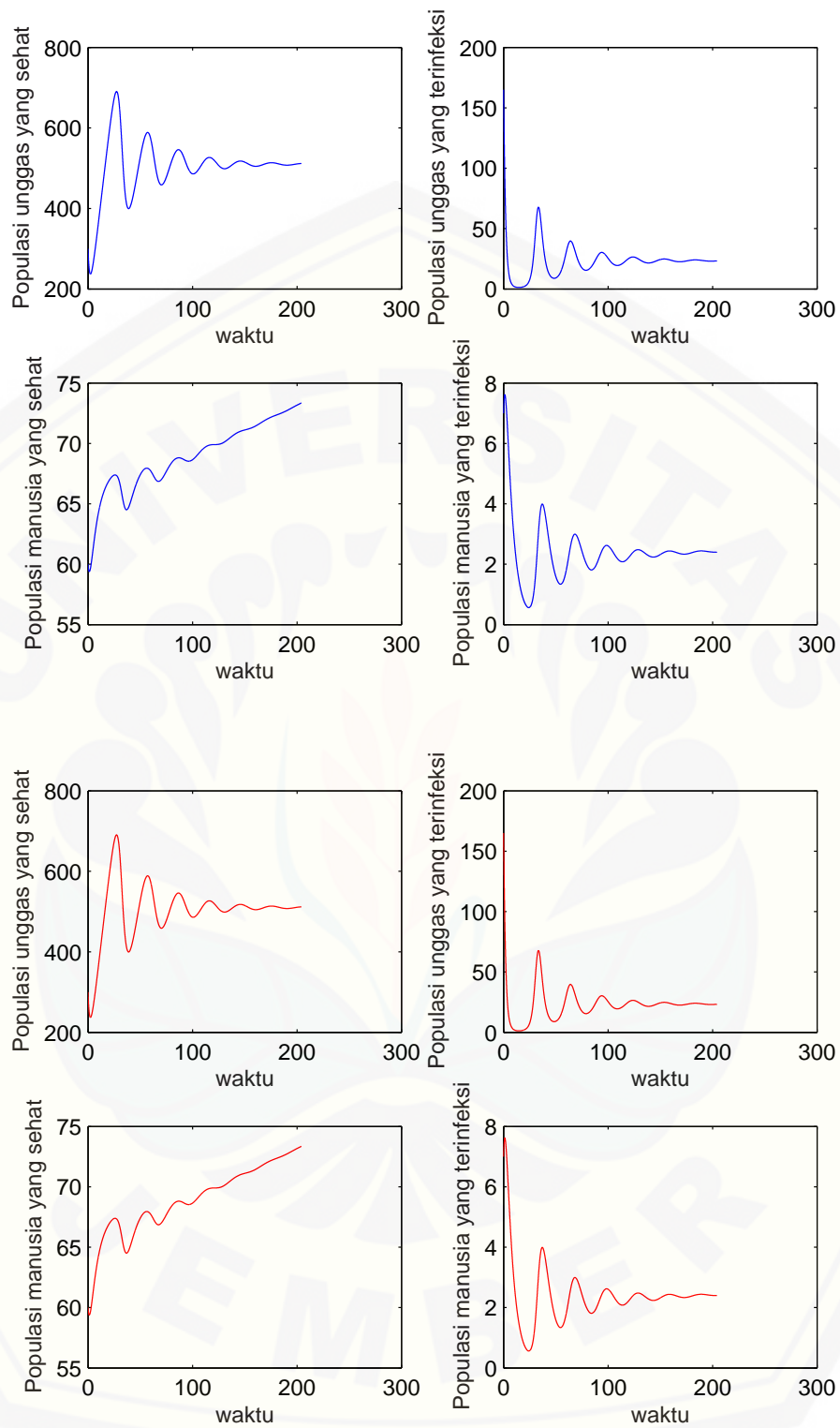


Gambar 4.10: Grafik eksekusi RK10B dengan iterasi 50000 pada populasi manusia dan unggas terhadap waktu (t) dengan $h=0.01$

run dari hari pertama sampai hari ke 20 sekitar 7 orang, kemudian mengalami fluktuasi sampai hari ke 200 dan selanjutnya mengalami kestabilan setelah hari ke 200 sampai hari ke 500 dengan populasi manusia yang terinfeksi pada saat stabil sekitar 3 orang. Dari hari pertama sampai hari ke 20 populasi unggas yang sehat banyaknya mengalami peningkatan sekitar 450 unggas, selanjutnya dari hari ke 20 sampai hari ke 150 mengalami fluktuasi hingga stabil pada hari ke 150 sampai hari ke 500 yaitu ada sekitar 500 unggas yang sehat. Begitu juga populasi unggas yang terinfeksi, dinamika populasi banyaknya unggas yang terinfeksi mengalami penurunan dari hari pertama sampai hari ke 10 sekitar 165 unggas, dari hari ke 10 sampai hari ke 150 mengalami fluktuasi dan setelah hari ke 150 sampai 500 cenderung stabil dengan banyak populai unggas sekitar 25 unggas.

Disisi lain akan ditampilkan hasil eksekusi berupa visualisasi grafik konvergensi metode Runge-Kutta orde sepuluh RK10A dan RK10B pada nilai toleransi (ϵ) 10^{-3} , dan 10^{-4} dari hubungan populasi manusia dan unggas terhadap waktu dengan nilai $h = 0.01$ adalah sebagai berikut:

Hasil visualisasi grafik dari eksekusi Runge-Kutta orde sepuluh RK10A dan RK10B dengan tol 10^{-3} pada populasi manusia dan unggas pada gambar (4.11) dengan ukuran langkah $h = 0,01$ menghasilkan penafsiran bahwa hasil eksekusi deskripsi grafik tersebut menjelaskan bahwa populasi unggas yang sehat mengalami kenaikan awal sampai hari ke 20 sekitar 450 dan setelah itu mengalami fluktuasi hingga stabil pada hari ke 150 yaitu ada sekitar 500 unggas yang sehat. Begitu juga populasi unggas yang terinfeksi mengalami penurunan sampai hari ke 20 sekitar 165 unggas, setelah itu mengalami fluktuasi hingga hari ke 150 dan setelah hari ke 150 sampai hari ke 200, mengalami kestabilan dengan banyak unggas yang terinfeksi pada saat stabil sekitar 25 ekor. Sedangkan populasi manusia yang sehat dari awal sampai hari ke 150 mengalami fluktuasi meningkat dan akhirnya stabil meningkat setelah hari ke 150 sampai hari ke 200. Sedangkan banyaknya populasi manusia yang terinfeksi virus Avian Influenza (flu burung) cenderung mengalami penurunan dari awal sampai hari ke 20 sekitar 7 orang dan setelah hari ke 20 mengalami fluktuasi sampai hari ke 150, pada saat hari ke 150

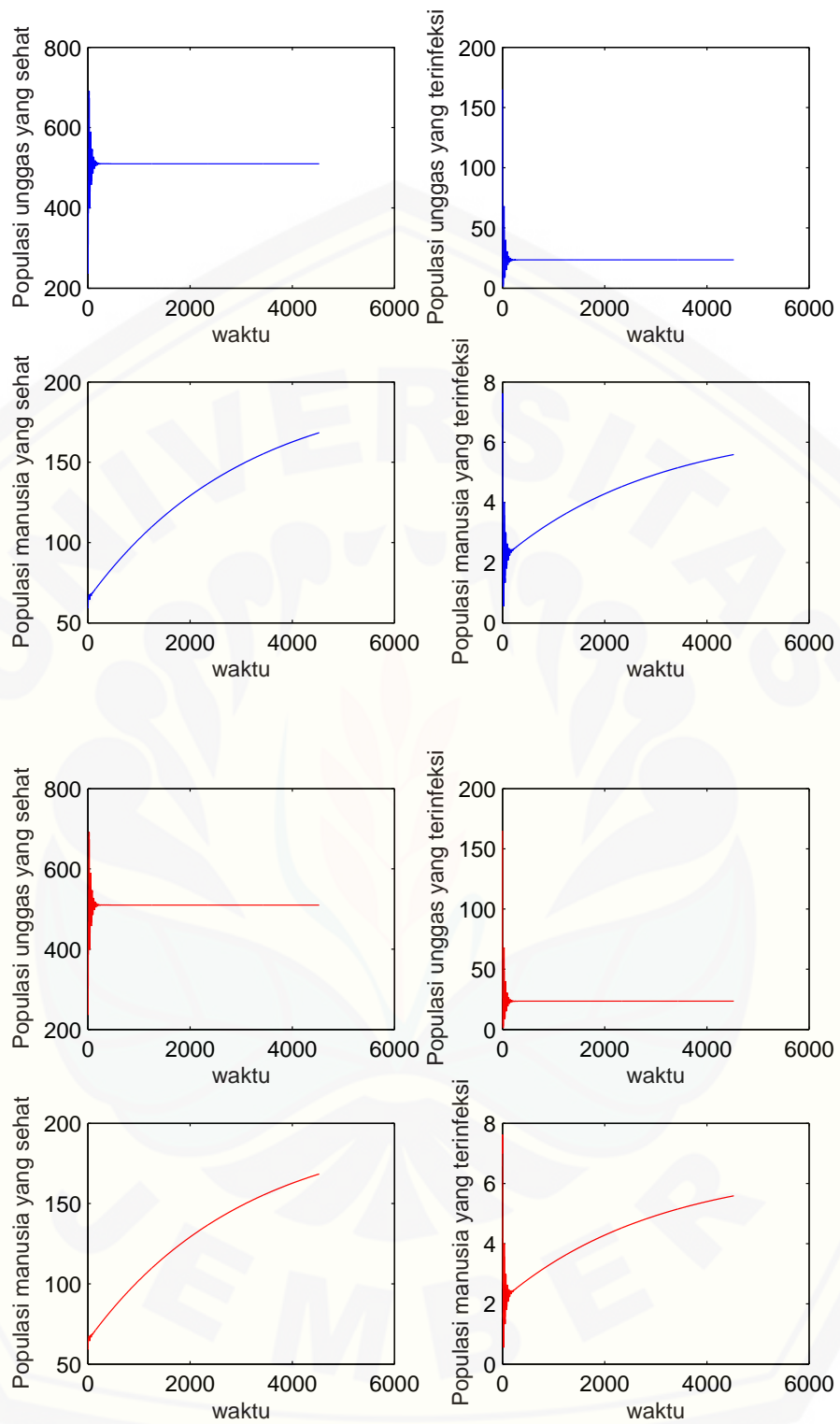


Gambar 4.11: Grafik eksekusi RK10A dan RK10B dengan tol 10^{-3} pada populasi manusia dan unggas terhadap waktu (t) dengan $h=0.01$

sampai 200 populasi manusia yang terinfeksi stabil yaitu sekitar 3 orang.

Pada gambar 4.12 merupakan Hasil visualisasi grafik dari eksekusi Runge-Kutta orde sepuluh RK10A dan RK10B dengan tol 10^{-4} dengan ukuran langkah $h = 0,01$ pada populasi manusia dan unggas menghasilkan penafsiran bahwa hasil eksekusi diskripsi grafik tersebut dari awal banyaknya populasi unggas yang sehat mengalami fluktuasi sampai hari ke 150 dan setelah itu mengalami stabil hingga pada hari ke 5000 yaitu ada sekitar 500 unggas yang sehat. Begitu juga populasi unggas yang terinfeksi, dinamika populasi banyaknya unggas yang terinfeksi mengalami fluktuasi sampai hari ke 150, dan setelah hari ke 150 mengalami kestabilan dengan banyak unggas yang terinfeksi pada hari sekitar 25 ekor. Sedangkan populasi manusia yang sehat dari awal sampai hari ke 150 mengalami fluktuasi meningkat dan akhirnya stabil meningkat setelah hari ke 150 sampai hari ke 5000. Sedangkan banyaknya populasi manusia yang terinfeksi virus Avian Influenza (flu burung) cenderung mengalami fluktuasi dari awal sampai hari ke 150, setelah hari ke 150 sampai 5000 populasi manusia yang terinfeksi mengalami peningkatan sekitar sebesar 3 orang.

Berikut ini akan dipaparkan hasil eksekusi grafik konvergensi dari berbagai iterasi, diantaranya pada iterasi $(i)=100, 500, 2500, 25000, 50000$. dengan nilai $h=0.01$ ditetapkan untuk mengetahui efektifitas metode Runge-Kutta orde sepuluh pada RK10A dan RK10B. Pengambilan nilai iterasi tersebut sudah cukup menggambarkan akurasi yang baik karena kesalahan (*error*) yang terjadi sudah cukup kecil. Grafik hasil eksekusi konvergensi metode Range-Kutta orde sepuluh memiliki bentuk yang hampir sama pada tingkat itersi yang sama pada RK10A dan RK10B baik pada populasi manusia ataupun populasi unggas terhadap waktu (t) sebagaimana pada gambar (4.13) sampai dengan gambar (4.17). Gambar tersebut menunjukkan bahwa semakin banyak iterasi maka semakin kecil error yang terjadi hingga mendekati nol. Hal tersebut membuktikan bahwa selain teruji konvergen secara teoritis, metode Runge-Kutta orde sepuluh juga telah teruji konvergen secara programming. Di sisi lain, grafik konvergensi yang dihasilkan oleh RK10A dan RK10B memiliki bentuk yang mirip. Artinya, kedua metode memiliki



Gambar 4.12: Grafik eksekusi RK10A dan RK10B dengan tol 10^{-4} pada populasi manusia dan unggas terhadap waktu (t) dengan $h=0.01$

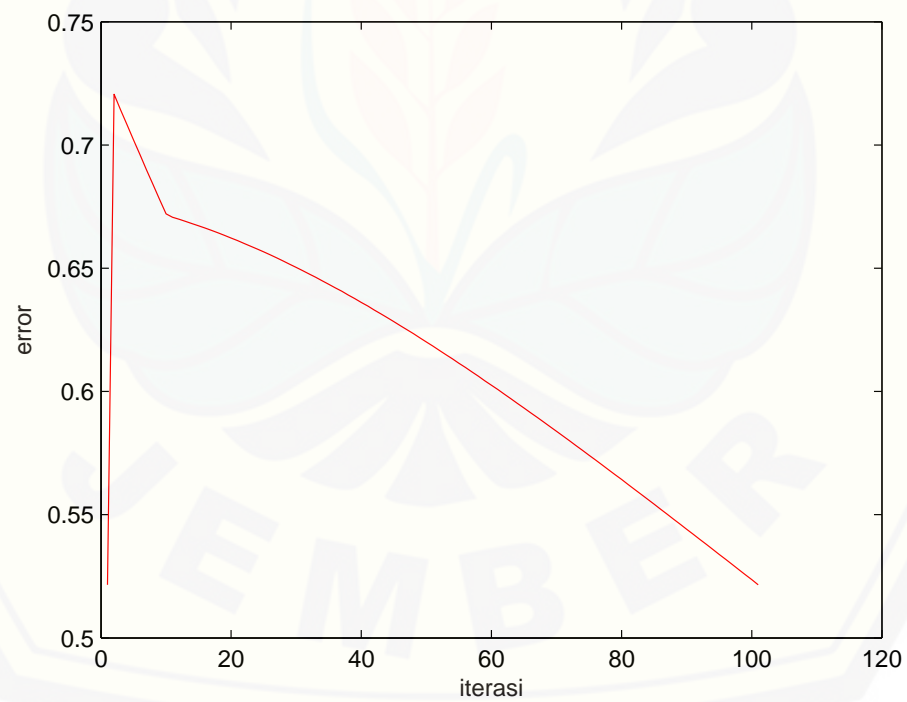
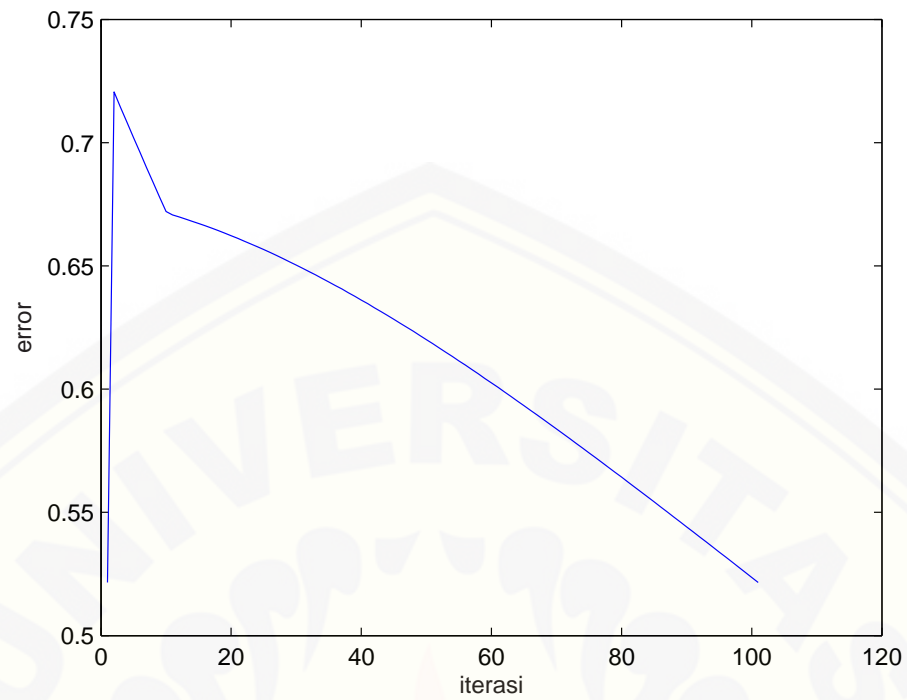
selisih error yang cukup kecil pada setiap iterasi.

Gambar 4.13 di iterasi 100 menunjukkan bahwa pada RK10A di iterasi 0 sampai dengan sekitar 5, nilai eror mengalami peningkatan dan setelah itu mengalami penurunan yang sangat tajam sampai pada itersi 100 dengan nilai eror sekitar 0,521549315967718. Hampir sama dengan RK10B hanya berbeda nilai erornya di iterasi 100 yaitu sebesar 0,521549436927813. Sedangkan pada gambar 4.14 untuk RK10A pada iterasi 500 terlihat bahwa pada iterasi 0 sampai dengan sekitar 375 mengalami penurunan dan setelah iterasi sekitar 375 sampai dengan 500 mengalami kenaikan, nilai eror pada iterasi ke 500 adalah 0,166445688963904. Hal yang mempengaruhi naik turunnya *error* itu adalah penetapan nilai awal populasi burung (unggas) yang rentan - susceptible birds ($S_B(t)$), burung (unggas) yang terinfeksi - infectious birds ($I_B(t)$), manusia yang rentan - susceptible humans ($S_H(t)$), dan manusia yang terinfeksi - infectious humans ($I_H(t)$). Tidak jauh berbeda dengan RK10A, RK10B pada iterasi ke 500 memiliki nilai eror sebesar 0,166445743277336.

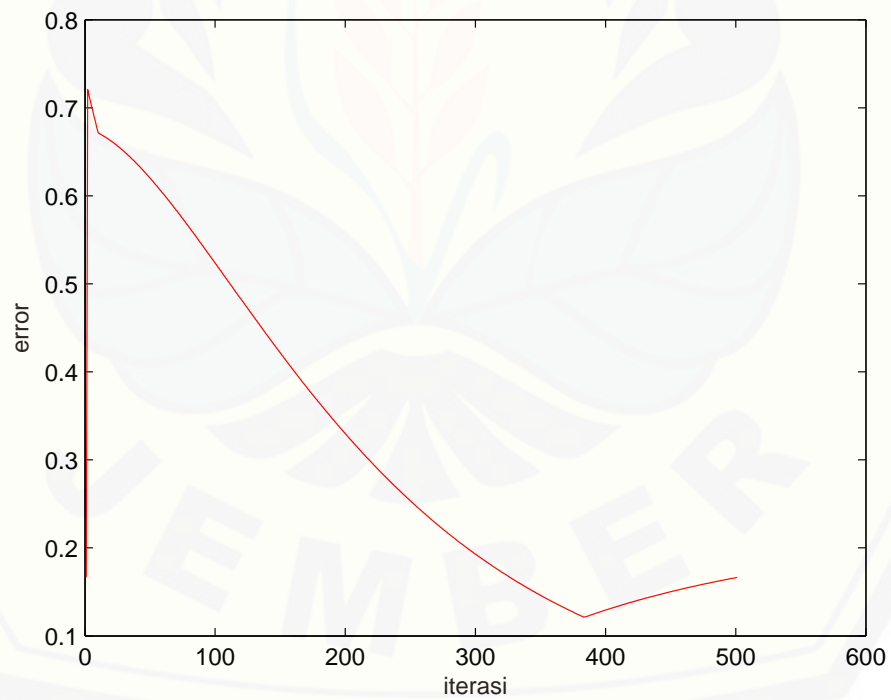
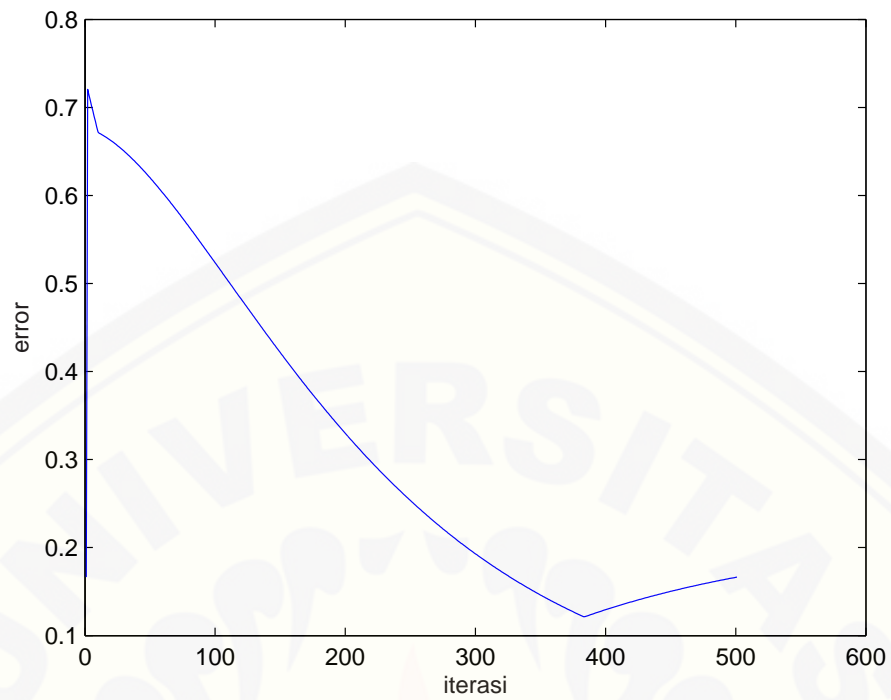
Gambar 4.15 di iterasi 2.500 pada RK10A terlihat bahwa dari iterasi 0 sampai iterasi sekitar 375 mengalami penurunan, setelah iterasi 375 sampai iterasi 1.000 mengalami kenaikan dan kembali lagi turun sampai pada iterasi 2.500 dengan nilai eror pada iterasi 2.5000 sebesar 0,129929791659833, tidak jauh berbeda dengan RK10B pada iterasi 2.500 mempunyai nilai eror sebesar 0,129929940257512. Untuk keterangan RK10B sama dengan RK10A.

Pada gambar 4.16, untuk keterangan RK10A sama dengan RK10B. terlihat bahwa pada RK10A di iterasi 25.000 nilai eror mengalami penurunan dari iterasi 0 sampai dengan iterasi 2.500 dan setelah itu mengalami kenaikan lagi pada iterasi 2.500 sampai dengan sekitar 2.510, pada iterasi 2510 sampai dengan iterasi 25.000 mengalami penurunan eror yang bersifat fluktuatif pada setiap iterasi, sehingga pada iterasi 25.000 nilai eror hampir mendekati nol yaitu sebesar 0,0004100342862187745 sedangkan pada RK10B tidak jauh berbeda pada iterasi 25.000 dan mempunyai nilai eror sebesar 0,000410035051956.

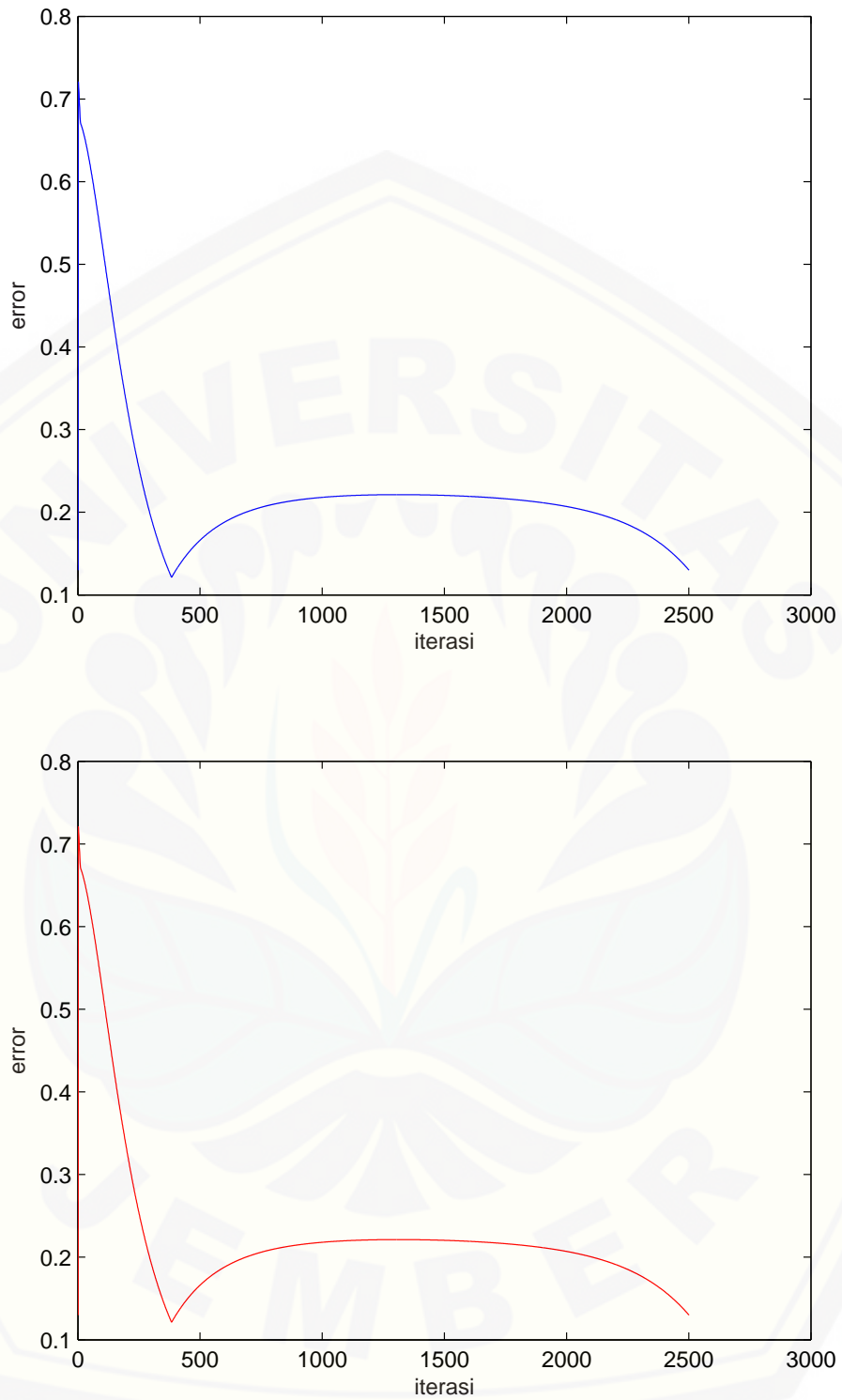
Pada RK10A di iterasi 50.000 di gambar 4.17 terlihat bahwa nilai eror men-



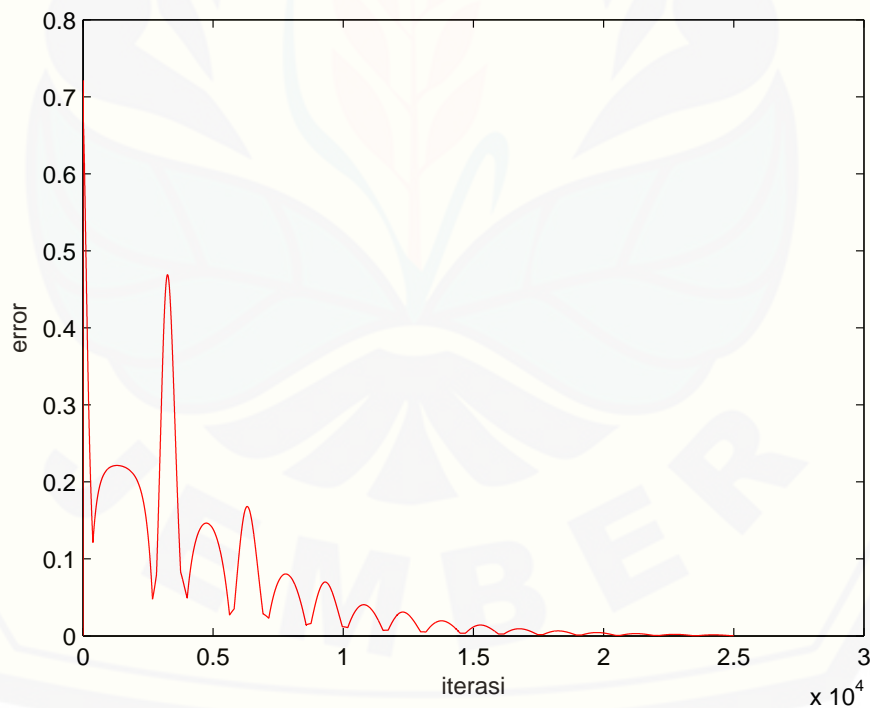
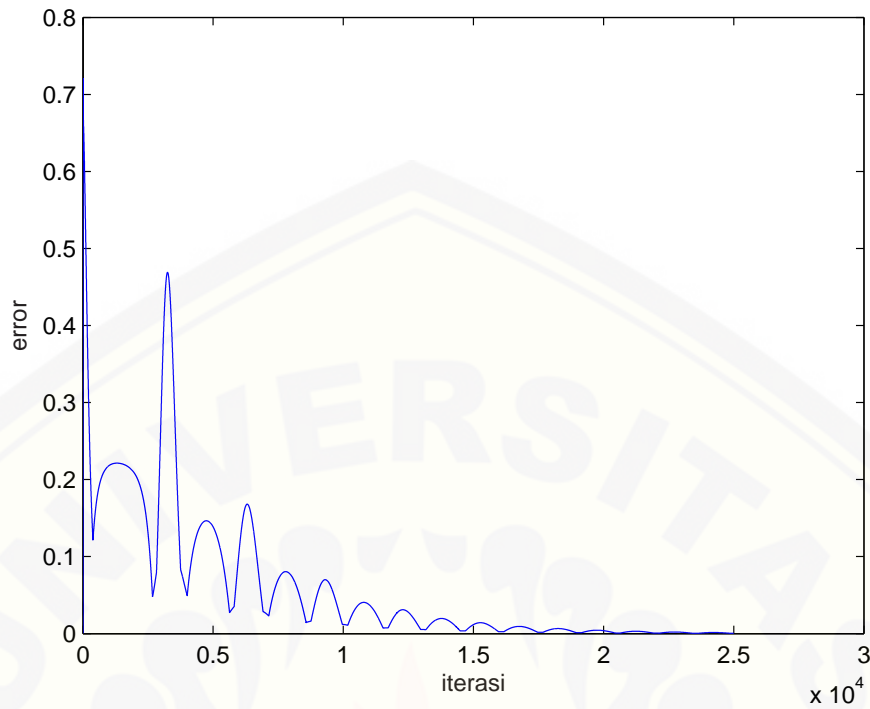
Gambar 4.13: Grafik konvergensi metode RK10A dan RK10B dengan iterasi 100 dan $h=0.01$ pada populasi manusia dan unggas



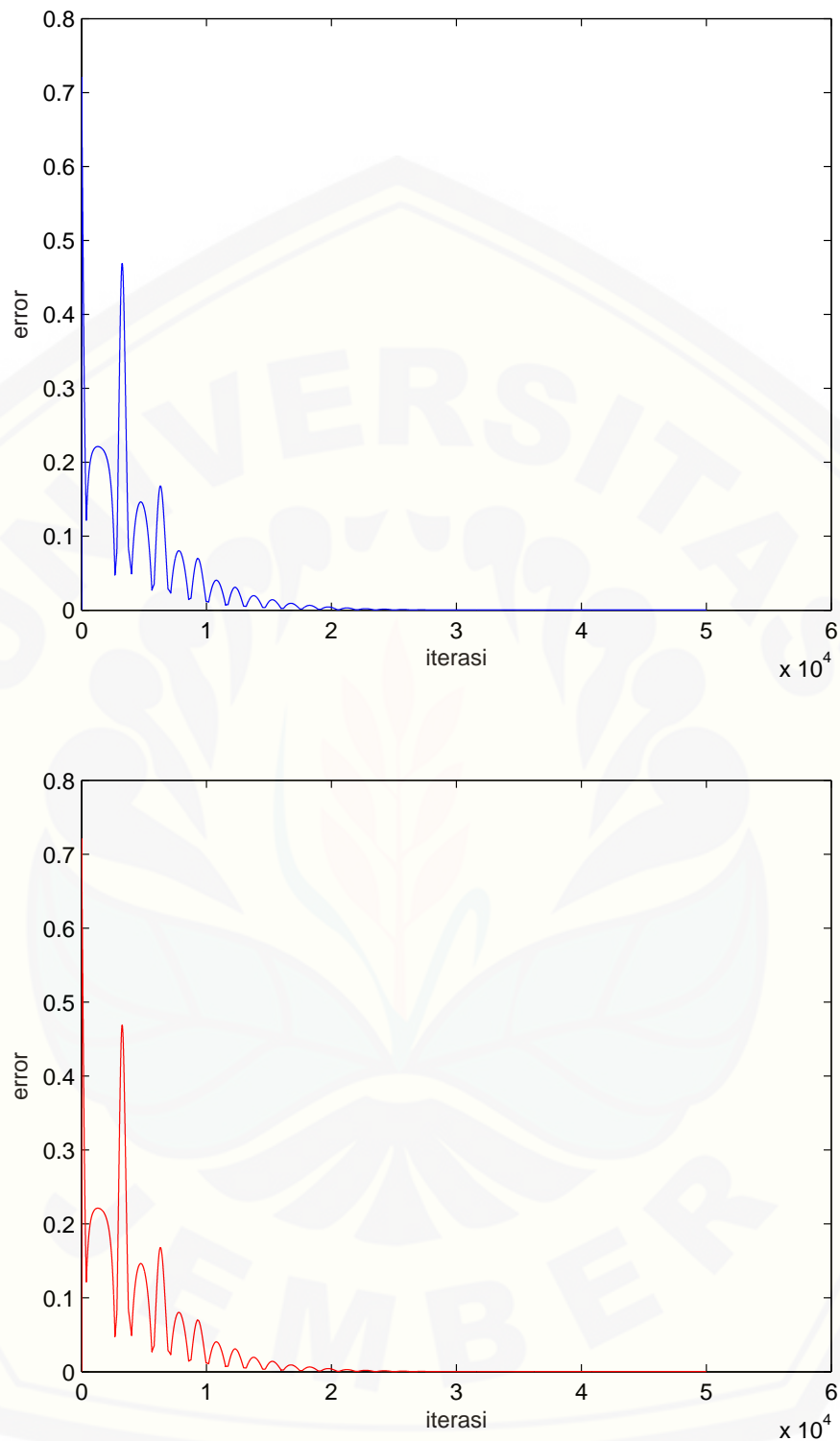
Gambar 4.14: Grafik konvergensi metode RK10A dan RK10B dengan iterasi 500 dan $h=0.01$ pada populasi manusia dan unggas



Gambar 4.15: Grafik konvergensi metode RK10A dan RK10B dengan iterasi 2.500 dan $h=0.01$ pada populasi manusia dan unggas



Gambar 4.16: Grafik konvergensi metode RK10A dan RK10B dengan iterasi 25.000 dan $h=0.01$ pada populasi manusia dan unggas



Gambar 4.17: Grafik konvergensi metode RK10A dan RK10B dengan iterasi 50.000 dan $h=0.01$ pada populasi manusia dan unggas

galami penurunan dari iterasi 0 sampai dengan iterasi 2.500 dan setelah itu mengalami kenaikan lagi pada iterasi 2.500 sampai dengan sekitar 2.510, pada iterasi 2510 sampai dengan iterasi 50.000 mengalami penurunan eror yang bersifat fluktuatif pada setiap iterasi, sehingga pada iterasi 50.000 nilai eror hampir mendekati nol yaitu sebesar 0,0003726058601358773 sedangkan pada RK10B di iterasi 50.000 tidak jauh berbeda pada iterasi 25.000 dan mempunyai nilai eror sebesar 0,0003726093146525500.

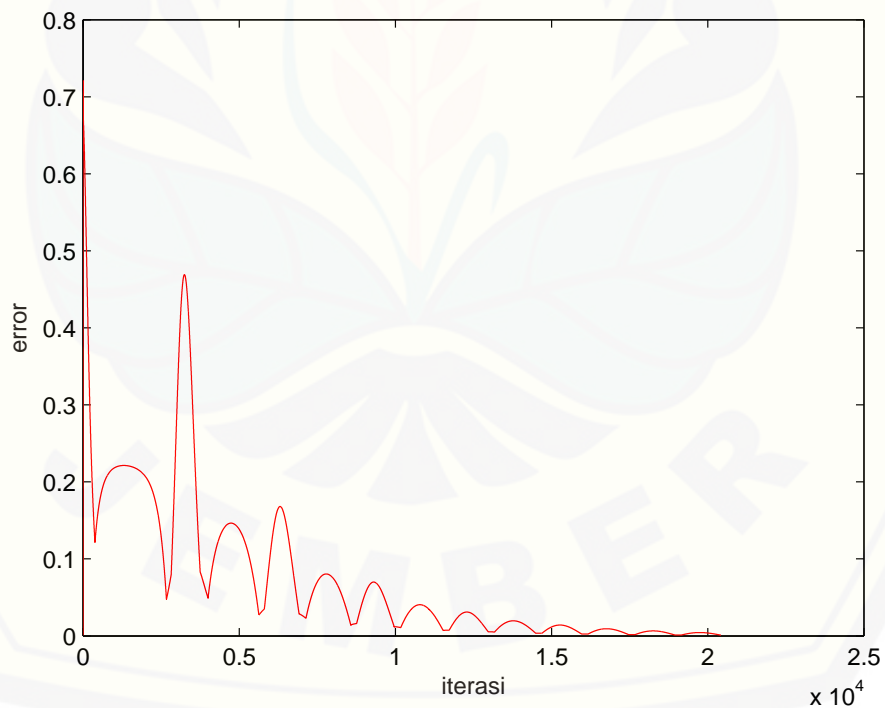
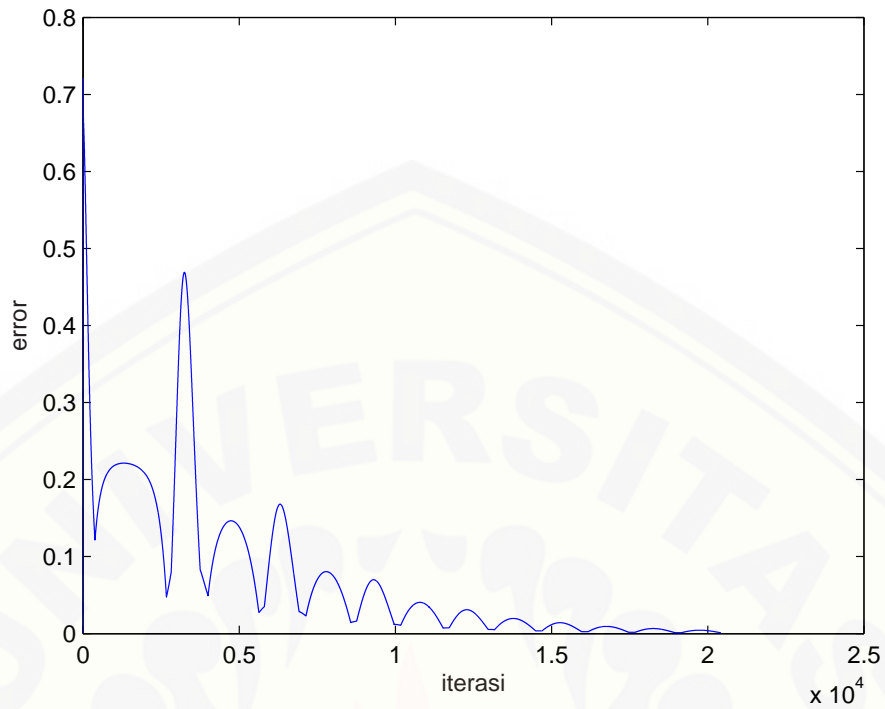
Di sisi lain pada nilai toleransi (ϵ) 10^{-3} , dan 10^{-4} ditetapkan untuk mengetahui efisiensi metode Runge-Kutta orde sepuluh pada RK10A dan RK10B. Pengambilan nilai toleransi tersebut sudah cukup menggambarkan akurasi yang baik karena kesalahan (*error*) yang terjadi sudah cukup kecil. Grafik hasil eksekusi metode Runge-Kutta orde sepuluh memiliki bentuk yang hampir sama pada tingkat toleransi yang sama pada RK10A dan RK10B sebagaimana terlihat pada gambar (4.18) sampai dengan gambar (4.19).

Pada gambar 4.18 bahwa RK10A pada tol 10^{-3} dan $h=0.01$ terlihat bahwa nilai eror mengalami fluktuasi menurun dari iterasi 0 sampai dengan iterasi sekitar 20.000 dan mengalami konvergen pada iterasi 20246. Sedangkan pada RK10B juga sama, terlihat bahwa nilai eror mengalami fluktuasi menurun dari iterasi 0 sampai dengan iterasi 20.000 dan mengalami konvergen pada iterasi 20246.

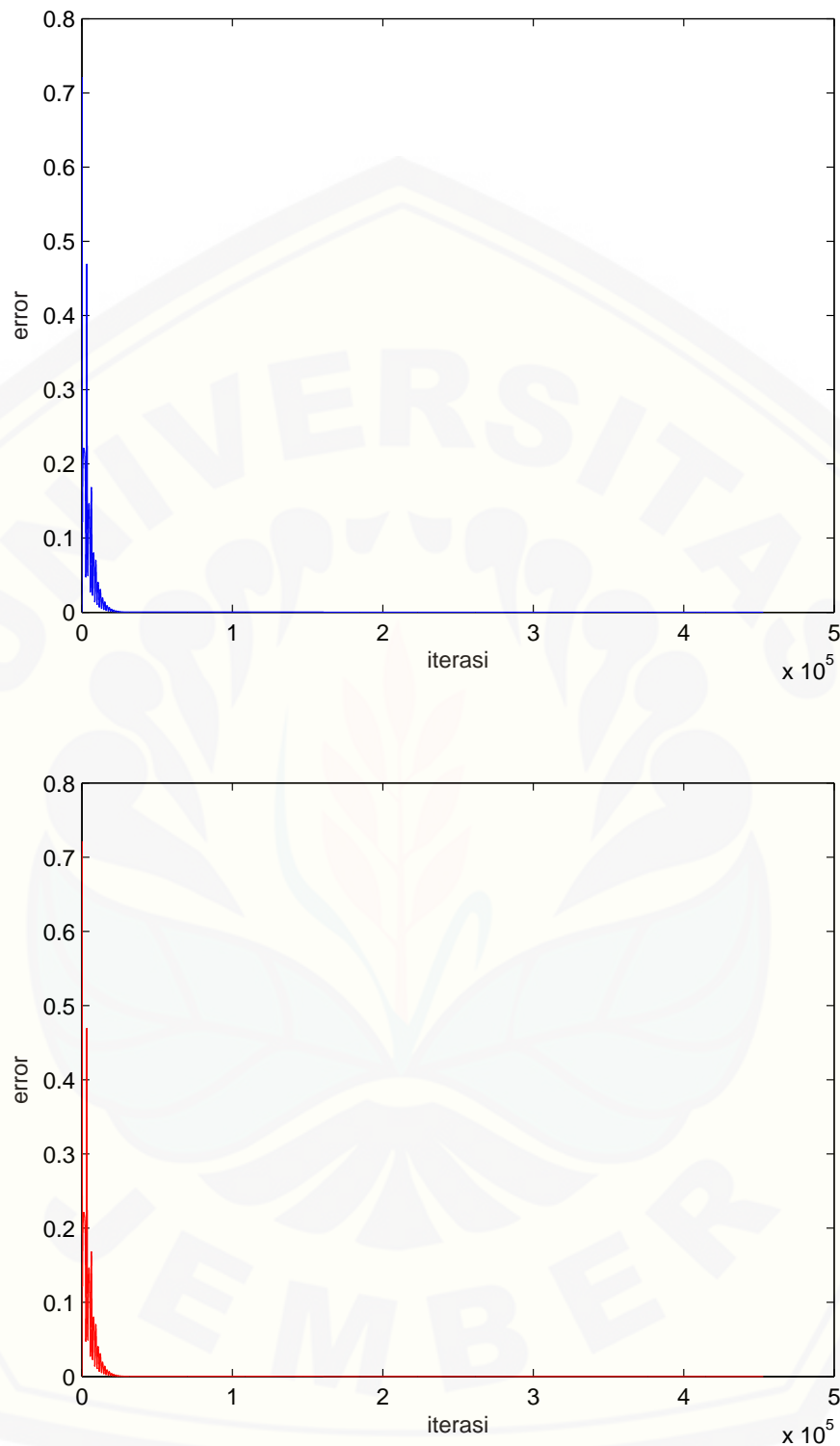
Hal yang lebih akurat lagi Pada gambar 4.19 bahwa RK10A pada tol 10^{-4} dan $h=0.01$ terlihat bahwa nilai eror mengalami fluktuasi menurun dari iterasi 0 sampai dengan iterasi 20426 dan mengalami konvergen pada iterasi 452.999. Sedangkan pada RK10B terlihat bahwa nilai eror mengalami fluktuasi menurun dari iterasi 0 sampai dengan iterasi 20426 dan mengalami konvergen pada iterasi 453002.

4.4.3 Analisis Efektivitas Metode Runge-Kutta Orde Sepuluh

Untuk mengetahui efektivitas metode Runge-Kutta orde sepuluh dilakukan penetapan iterasi (*i*) sehingga dihasilkan data berupa *error* atau galat. Semakin kecil galat yang terjadi, maka semakin efektif suatu metode untuk menyelesaikan



Gambar 4.18: Grafik konvergensi metode RK10A dan RK10B dengan tol 10^{-3} dan $h=0.01$ pada populasi manusia dan unggas



Gambar 4.19: Grafik konvergensi metode RK10A dan RK10B dengan tol 10^{-4} dan $h=0.01$ pada populasi manusia dan unggas

suatu permasalahan. Dalam penelitian ini, *error* yang dihasilkan oleh RK10A dan RK10B akan di tampilkan.

Suatu masalah dalam model matematika memiliki nilai parameter dan tetapan yang berbeda-beda. Hal itu akan mempengaruhi tingkat efektivitas suatu metode untuk menyelesaikannya. Masalah yang akan dianalisis dalam penelitian ini adalah penyebaran virus Avian Influenza. Tidak hanya itu, tetapan yang berbeda untuk kasus yang sama akan memiliki penyelesaian yang berbeda pula. Hal itu berarti penentuan parameter dan tetapan akan berpengaruh terhadap hasil pemrograman. Tabel (4.4) berikut adalah data hasil eksekusi metode Runge-Kutta orde sepuluh berupa data *error*.

Tabel 4.4: Data Efektivitas Metode Runge-Kutta Orde Sepuluh

Iterasi	Error pada metode	
	RK10A	RK10B
50	0,618389437602218	0,618389206709367
100	0,521549315967718	0,521549436927813
250	0,252096606172778	0,252096775461730
500	0,166445688963904	0,166445743277336
1000	0,218065587371939	0,218065620839980
2500	0,129929791659833	0,129929940257512
5000	0,138493022733201	0,138493064104296
10000	0,011879273683487	0,011879270189031
25000	0,0004100342862187745	0,0004100350519564699
50000	0,0003726058601358773	0,0003726093146525500
100000	0,0003164918182392285	0,0003164947497538151
200000	0,0002283577423725092	0,0002283598575445467

Dalam penelitian ini efektivitas diukur berdasarkan *error* yang dihasilkan pada setiap iterasi yang telah ditentukan sebelumnya. Adapun iterasi yang digunakan adalah 50, 100, 250, 500, 1.000, 2.500, 5.000, 10.000, 25.000, 50.000, 100.000 dan 200.000. Pemilihan iterasi tersebut sudah cukup mewakili tingkat akurasi di setiap selang karena memiliki rentang yang jauh. Tabel (4.4) di atas menunjukkan bahwa semakin besar iterasi maka semakin kecil *error* yang terjadi. Tidak hanya itu, penetapan konstanta c dan koefisien matriks ternyata dapat mempengaruhi

hasil *programming* MATLAB. RK10A merupakan formula metode Runge-Kutta orde sepuluh dengan koefisien matriks sedikit nol (*non sparse matrix*). Sedangkan RK10B merupakan formula metode Runge-Kutta orde sepuluh dengan koefisien matriks banyak nol (*sparse matrix*).

Berdasarkan tabel (4.4) diketahui bahwa metode RK10A memiliki *error* yang lebih kecil bila dibandingkan dengan metode RK10B pada iterasi 100, 250, 500, 2.500, 5.000, 10.000, 25.000, 50.000, 100.000 dan 200.000. Sedangkan pada iterasi 50 dan 1.000 kedua metode tersebut memiliki *error* lebih kecil RK10B dari pada RK10A.

Gambar (4.13) sampai dengan (4.17) merupakan grafik konvergensi metode Runge-Kutta orde sepuluh dengan iterasi $(i)=100, 500, 2500, 25000$ dan 50000 . dengan nilai $h=0.01$. Gambar itu menunjukkan bahwa semakin banyak iterasi maka semakin kecil *error* yang terjadi hingga mendekati nol. Hal tersebut membuktikan bahwa selain teruji konvergen secara teoritis, metode Runge-Kutta orde sepuluh juga telah teruji konvergen secara *programming*. Di sisi lain, grafik konvergensi yang dihasilkan oleh metode Runge-Kutta orde sepuluh RK10A dan RK10B ini memiliki bentuk yang mirip. Artinya, kedua (RK10A dan RK10B) memiliki selisih *error* yang cukup kecil pada setiap iterasi.

Bila gambar (4.16) dan (4.17) diperhatikan dengan lebih detil, maka tampak bahwa *error* metode Runge-Kutta orde sepuluh RK10A dan RK10B semakin lama semakin menurun. Pada gambar (4.16) terlihat bahwa pada RK10A di iterasi 25.000 nilai eror mengalami penurunan dari iterasi 0 sampai dengan iterasi 2.500 dan setelah itu mengalami kenaikan lagi pada iterasi 2.500 sampai dengan sekitar 2.510, pada iterasi 2510 sampai dengan iterasi 25.000 mengalami penurunan eror yang bersifat fluktuatif pada setiap iterasi, sehingga pada iterasi 25.000 nilai eror hampir mendekati nol yaitu sebesar $0,0004100342862187745$ sedangkan pada RK10B tidak jauh berbeda pada iterasi 25.000 dan mempunyai nilai eror sebesar $0,000410035051956$. Pada RK10A di iterasi 50.000 di gambar (4.17) terlihat bahwa nilai eror mengalami penurunan dari iterasi 0 sampai dengan iterasi 2.500 dan setelah itu mengalami kenaikan lagi pada iterasi 2.500 sampai dengan seki-

tar 2.510, pada iterasi 2510 sampai dengan iterasi 50.000 mengalami penurunan eror yang bersifat fluktuatif pada setiap iterasi, sehingga pada iterasi 50.000 nilai eror hampir mendekati nol yaitu sebesar 0,0003726058601358773, sedangkan pada RK10B di iterasi 50.000 tidak jauh berbeda pada iterasi 25.000 dan mempunyai nilai eror sebesar 0,0003726093146525500. Secara keseluruhan dapat dikatakan bahwa metode Runge-Kutta orde sepuluh RK10A lebih efektif bila dibandingkan dengan RK10B.

Adapun hal yang mempengaruhi naik turunnya *error* itu adalah penetapan nilai awal populasi burung (unggas) yang sehat - susceptible birds ($S_B(t)$)= 300, burung (unggas) yang terinfeksi - infections birds ($I_B(t)$)= 165, manusia yang sehat - susceptible human ($S_H(t)$)= 60, dan manusia yang terinfeksi - infeksi human ($I_H(t)$)=7. Penetapan nilai itu dapat menghasilkan grafik yang konvergen sebagaimana gambar tersebut.

4.4.4 Analisis Efisiensi Metode Runge-Kutta Orde Sepuluh

Untuk mengetahui efisiensi metode Runge-Kutta orde sepuluh maka dilakukan penetapan batas toleransi ϵ sehingga dihasilkan data iterasi dan waktu tempuh seperti pada tabel (4.5). Semakin kecil waktu yang ditempuh oleh suatu metode untuk menyelesaikan permasalahan, maka semakin efisien metode tersebut. Selain menghasilkan data tersebut, hasil eksekusi *programming* ini juga menghasilkan data berupa grafik hubungan iterasi dan *error* (grafik konvergensi).

Tabel (4.5) menunjukkan bahwa jumlah iterasi dari metode Runge-Kutta orde sepuluh RK10A dan RK10B pada setiap toleransi yang telah ditetapkan ada yang sama dan ada pula yang berbeda. Untuk mencapai batas toleransi 10^{-1} , 10^{-2} , 10^{-3} , 10^{-4} , dan 10^{-5} jumlah iterasi yang dibutuhkan kelima metode berturut-turut yaitu 10^{-1} = 2577 iterasi, 10^{-2} = 11491 iterasi, 10^{-3} = 20426 iterasi, 10^{-4} = selisih 2 antara RK10A dan RK10B, 10^{-5} = selisih 2 antara RK10A dan RK10B. Berhubung ada iterasi yang berbeda maka waktu tempuh dari metode Runge-Kutta orde sepuluh RK10A dan RK10B ada yang berbeda, semakin banyak iterasinya maka semakin banyak waktu yang diperlukan.

Tabel 4.5: Data Efisiensi Runge-Kutta Orde Sepuluh

Output	Toleransi (e)	RK10A	RK10B
Iterasi	0,1	2.577	2.577
	0,01	11.491	11.491
	0,001	20.426	20.426
	0,0001	452.999	453.002
	0,00001	1.158.483	1.158.485
Waktu (detik)	0,1	8,626999999999999	7,862000000000002
	0,01	27,081000000000003	8,050000000000001
	0,001	13,323000000000000	13,290999999999999
	0,0001	452,9310000000000	489,7320000000000
	0,00001	299,3957000000000	327,1934000000000

Pada toleransi 10^{-1} metode RK10B memiliki waktu tempuh lebih cepat sekitar 0,765 detik terhadap metode RK10A. Pada toleransi 10^{-2} metode RK10B juga memiliki waktu tempuh lebih cepat sekitar 19,031 detik terhadap metode RK10A. Begitu juga pada toleransi 10^{-3} , metode RK10B memiliki waktu tempuh lebih cepat sekitar 0,032 detik terhadap metode RK10A. Pada toleransi 10^{-4} metode RK10A memiliki waktu tempuh lebih cepat sekitar 36,801 detik terhadap metode RK10B. Pada toleransi 10^{-5} metode RK10A juga memiliki waktu tempuh lebih cepat sekitar 27,7977 detik terhadap metode RK10B.

Pada toleransi 10^{-4} , metode RK10A memiliki waktu tempuh lebih cepat sekitar 36,801 detik terhadap metode RK10B dikarenakan pada toleransi tersebut RK10B memiliki jumlah iterasi yang lebih banyak dari pada RK10A sekitar 3 iterasi. Sedangkan pada toleransi 10^{-5} , metode RK10A memiliki waktu tempuh lebih cepat sekitar 27,7977 detik terhadap metode RK10B dikarenakan pada toleransi tersebut RK10B memiliki jumlah iterasi yang lebih banyak dari pada RK10A sekitar 2 iterasi.

Dengan demikian dapat disimpulkan bahwa metode RK10B merupakan metode yang paling efisien dalam menyelesaikan model penyebaran virus Avian Influenza karena memiliki waktu tempuh paling sedikit untuk mencapai setiap batas toleransi yang ditentukan. Hal itu dimungkinkan karena metode RK10B memiliki

matrik dengan koefisien banyak nol (*sparse matrix*) sehingga jumlah operasinya (flops) lebih sedikit dibandingkan RK10A sehingga proses eksekusinya lebih cepat. Akan tetapi dalam penelitian ini tidak dibahas tentang flops dikarenakan fungsi tersebut tidak tersedia di MATLAB R2011b. Secara keseluruhan dapat dikatakan bahwa metode Runge-Kutta orde sepuluh RK10B lebih efisien bila dibandingkan dengan RK10A.



BAB 5. KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil dari pembahasan pada bab sebelumnya, dapat disimpulkan bahwa:

1. Metode Runge-Kutta sepuluh tahap yang sekaligus berorde sepuluh mempunyai sifat sebagai berikut:

$$\sum_{i=1}^{10} b_i = 1$$

$$\sum_{i=2}^9 b_i c_i^p = \frac{1}{p+1}; \quad \text{dimana } p = 1, 2, 3, 4, \dots, m-1$$

$$\sum_{i=3}^7 b_i \left(\sum_{j=2}^{i-1} c_j^q a_{ij} \right) = \frac{1}{(q+1)(q+2)}, \quad \text{dimana } q = 1, 3, 4, \dots, m-3$$

2. Salah satu hasil penurunan formula metode Runge-Kutta yaitu:

$$y_{n+1} = y_n + \frac{h}{89600} (2857k_1 + 15741k_2 + 1080k_3 + 19344k_4 + 5778k_5 + 5778k_6 + 19344k_7 + 1080k_8 + 15741k_9 + 2857k_{10})$$

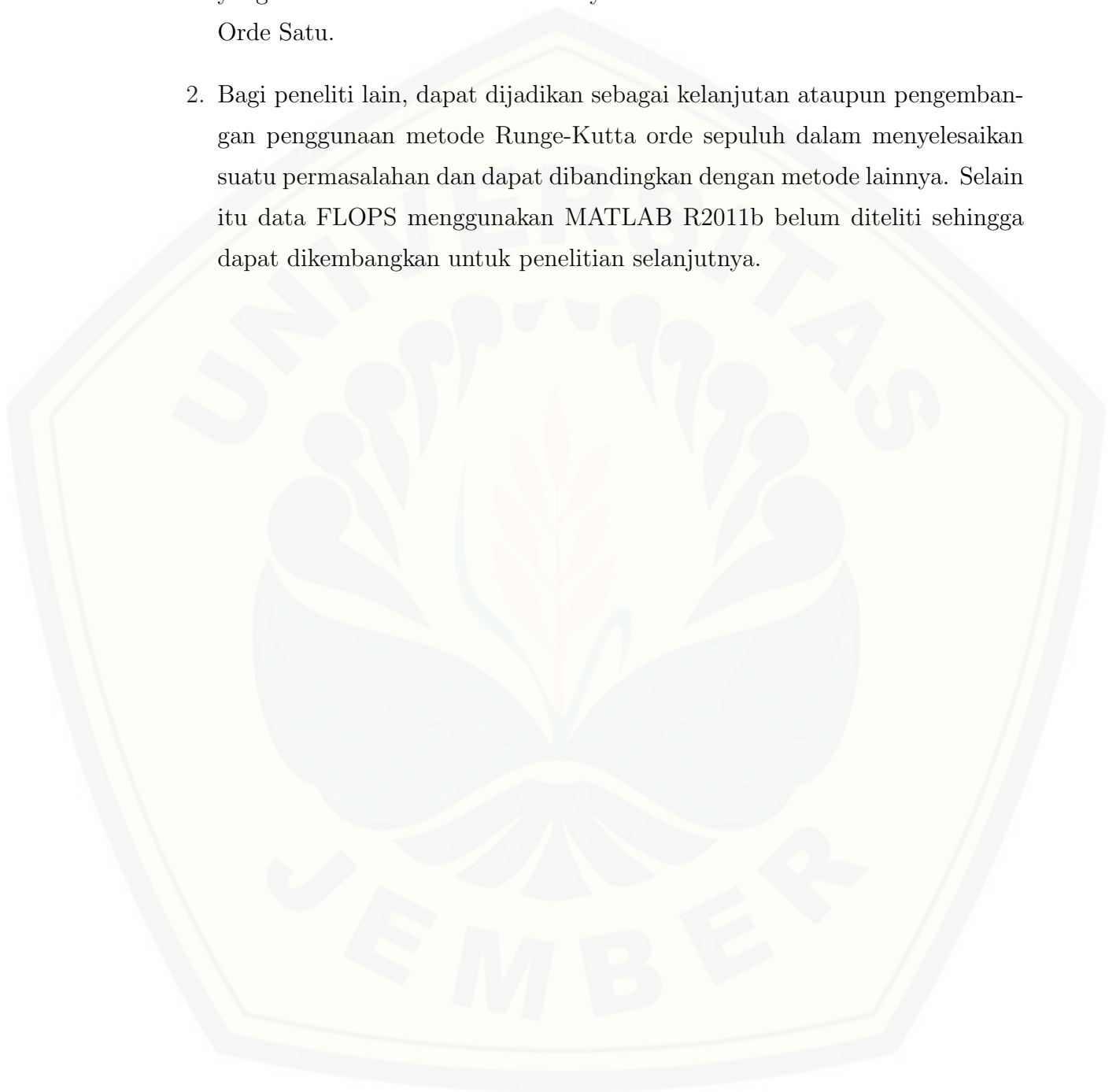
dengan,

$$\begin{aligned}
k_1 &= f(x_n, y_n) \\
k_2 &= f\left(x_n + \frac{h}{9}, y_n + \frac{h}{9}k_1\right) \\
k_3 &= f\left(x_n + \frac{2h}{9}, y_n + \frac{h}{360}(-1617k_1 + 1697k_2)\right) \\
k_4 &= f\left(x_n + \frac{3h}{9}, y_n + \frac{h}{1080}(2501k_1 + 1000k_2 - 3141k_3)\right) \\
k_5 &= f\left(x_n + \frac{4h}{9}, y_n + \frac{h}{8667}(-154197k_1 - 8620k_2 - 8640k_3 + 175309k_4)\right) \\
k_6 &= f\left(x_n + \frac{5h}{9}, y_n + \frac{h}{5778}(99353k_1 - 5700k_2 - 5740k_3 + 5620k_4 - 90323k_5)\right) \\
k_7 &= f\left(x_n + \frac{6h}{9}, y_n + \frac{h}{5580}(-16343k_1 - 520k_2 + 5540k_3 - 5420k_4 - 5560k_5\right. \\
&\quad \left.+ 26023k_6)\right) \\
k_8 &= f\left(x_n + \frac{7h}{9}, y_n + \frac{h}{450}(11546k_1 + 445k_2 - 440k_3 + 420k_4 + 320k_5\right. \\
&\quad \left.+ 420k_6 - 12361k_7)\right) \\
k_9 &= f\left(x_n + \frac{8h}{9}, y_n + \frac{h}{4770}(-1281k_1 + 4720k_2 + 390k_3 + 1280k_4 - 2860k_5\right. \\
&\quad \left.+ 2700k_6 - 4762k_7 + 4053k_8)\right) \\
k_{10} &= f\left(x_n + h, y_n + \frac{h}{1800}(-2500k_1 + 1000k_2 + 120k_3 + 1200k_4 - 600k_5 + 240k_6\right. \\
&\quad \left.+ 1800k_7 + 540k_8)\right)
\end{aligned}$$

3. Metode Runge-Kutta orde sepuluh merupakan metode yang konvergen, dibuktikan pada teorema 4.1.
4. Berdasarkan hasil eksekusi *programming* metode Runge-Kutta orde sepuluh RK10A lebih efektif dibandingkan metode Runge-Kutta orde sepuluh RK10B. Akan tetapi metode Runge-Kutta orde sepuluh RK10B lebih efisien dibandingkan metode Runge-Kutta orde sepuluh RK10A dalam menyelesaikan model.

5.2 Saran

1. Bagi praktisi, hasil analisis efektivitas dan efisiensi metode Runge-Kutta orde sepuluh dapat dijadikan sebagai rujukan untuk memperkirakan metode yang efektif dan efisien dalam menyelesaikan suatu sistem PDB Non Linier Orde Satu.
2. Bagi peneliti lain, dapat dijadikan sebagai kelanjutan ataupun pengembangan penggunaan metode Runge-Kutta orde sepuluh dalam menyelesaikan suatu permasalahan dan dapat dibandingkan dengan metode lainnya. Selain itu data FLOPS menggunakan MATLAB R2011b belum diteliti sehingga dapat dikembangkan untuk penelitian selanjutnya.



DAFTAR PUSTAKA

- Anggraeni,D. 2013.*Efektivitas Metode Runge-Kutta Orde Sembilan untuk Menyelesaikan Model Matematika Pada Sistem Kekebalan Tubuh Terhadap Infeksi Mycobacterium Tuberculosis.*(Tidak dipublikasikan). Artikel. Jember: Fakultas Keguruan dan Ilmu Pendidikan.Universitas Jember.
- Ardhilia,R,M. 2013.*Efektivitas Metode Runge-Kutta Orde Delapan untuk Menyelesaikan Model Matematika Transmisi Penyakit Malaria.*(Tidak dipublikasikan). Artikel. Jember: Fakultas Keguruan dan Ilmu Pendidikan.Universitas Jember.
- Asih, T. 2001. *Efektivitas Metode Runge-Kutta Order 3 Dalam Menyelesaikan Model Gerak Pendulum Nonlinier.* Tidak dipublikasikan. Artikel. Jember: FKIP Universitas Jember.
- Badan Penelitian dan Pengembangan Kesehatan, Depkes RI. 20 Februari 2010. *Flu Burung.* http://www.litbang.depkes.go.id/maskes/072005/flu_burung.pdf.
- Conte, S.D. and Carl de Boor. 1993. *Dasar-Dasar Analisis Numerik Suatu Pendekatan Algoritma.* Tidak dipublikasikan. Artikel. Jakarta: Penerbit Erlangga.
- Dafik. 1998. *Metoda Numerik I.* Jember: Universitas Jember.
- Dafik. 1999a. *Matlab dalam Matematika.* Jember: FKIP Universitas Jember.
- Dafik. 1999b. *Persamaan Differensial Biasa (PDB): Masalah Nilai Awal dan Batas.* Jember: FKIP Universitas Jember.
- Dafik. 1999c. *Metode Numerik Dalam PDB dan MNA (Tidak diterbitkan).* Jember: FKIP Universitas Jember.
- Dafik. 2008a. *Pengantar PDP dan Solusi Analitik.* Jember: FKIP Universitas Jember.
- Dafik. 2008b. *Sistem PDB Nonlinier dan Keseimbangan.* Jember: FKIP Universitas Jember.

- Dafik. 2009. *Metode Numerik dan Aplikasinya*. Jember: FKIP Universitas Jember.
- Dafik. 2010. *Metode Numerik dalam Menyelesaikan Fungsi-Fungsi Aproksimasi..* Jember: FKIP Universitas Jember.
- Dinas Kesehatan Jember.2014.<http://www.antarajatim.com>. [12 Januari 2015]
- Dorn, W. S. dan Mc. Cracken, D. D. 1986. *Studi Kasus Metode Numerik dengan Fortran IV* . Jakarta : Erlangga.
- Faisol, A. 2001. *Penerapan Metode Runge-Kutta Orde Empat Untuk Menyelesaikan Model Penyebaran Virus Dengue Oleh Nyamuk Aedes aegypti*. Tidak dipublikasikan. Artikel. Jember : FKIP Universitas Jember.
- Fausett, L. V. 2008. *Applied Numerical Analysis Using MATLAB Second Edition*. USA : Pearson Education Inc.
- Finizio, N and Ladas, G. 1988. *Persamaan Differensial Biasa dengan Penerapan Modern*. (Penerjemah: Dra. Widiarti Santoso). Jakarta: Erlangga.
- Guswai, Cristian. 2007. *How To Operate Your Store Effectively Yet Efficiently..* Jakarta: PT Elex Media Komputindo.
- Kumala, W. 2005. *Avian influenza : profil dan penularannya pada manusia*. <http://www.univmed.orgwp-content/uploads201102widyasari/> [24 April 2015].
- Lambert, J. D. 1997. *Numerical Method for Ordinary Differential Systems* . New York: John Wiley & Sons.
- Okosun. 2007. *Numerical Simultan of Bird-Flu Epidemics (Medwell Journall)*. Akure: Federal University of Technology.
- Ruhimat, Q.A.2013. *Efektivitas Metode Adam Bashforth Moulton Order Dua Belas Dalam Menganalisa Model Dinamika Penularan Virus Rabies*. (Tidak dipublikasikan).Artikel.Jember: Fakultas Keguruan dan Ilmu Pendidikan. Universitas jember.
- Sahid. 2005. *Pengantar Komputansi Numerik dengan MATLAB*. Yogyakarta:Andi.

Sartono. 2006. *Metode Numerik*. Jakarta: Engine Press.

Shampine, L.F. 1994. *Numerical Solution of Ordinary Differential Equation*. London: Chapman and Hall.

Shodiq, L.J. 2012. *Efektivitas Metode Runge-Kutta Orde Tujuh Terhadap Metode Multistep Adams Orde Enam Pada Model Penyebaran Penyakit Tuberkulosis (TB)*. (Tidak dipublikasikan). Artikel. Jember: Fakultas Keguruan dan Ilmu Pendidikan. Universitas Jember.

Susanti, N.I. 2010. *Efektivitas Metode Multistep Linier Implisit Order Lima untuk Menyelesaikan Model Persamaan Penyebaran bakteri Leptospira*. (Tidak dipublikasikan). Artikel. Jember: Fakultas Keguruan dan Ilmu Pendidikan. Universitas jember.

Suzetta, P. 2005. *Rencana Strategis Nasional Pengendalian Flu Burung (Avian Influenza) dan Kesiapsiagaan Menghadapi Pandemi Influenza*. Jakarta: Menteri Negara PPN/ Kepala Bappenas.

Wibisono, A. 2010. *Efektif dan Efisieni*. <http://aguswibisono.com/2010/efektif-dan-efisieni/> [24 April 2015]

Yanse, N.M.N. 2012. *Efektivitas Metode Adams Bashforth Moulton Order Sembilan dalam Menganalisis Model Penyebaran Penyakit Demam Berdarah Dengue (DBD)*. (Tidak dipublikasikan). Artikel. Jember: Fakultas Keguruan dan Ilmu Pendidikan. Universitas Jember.

Yustica, A. 2010. *Efektivitas Metode Runge-Kutta Order Lima Untuk Menyelesaikan Model Penyebaran Virus Avian Influenza*. (Tidak dipublikasikan). Artikel. Jember: Fakultas Keguruan dan Ilmu Pendidikan. Universitas Jember.

LAMPIRAN

LAMPIRAN A. Format Pemrograman Metode Runge-Kutta Orde Sepuluh

• **Format Programming Efektifitas**

Format *programming* yang dikembangkan dari Metode Runge-Kutta orde sepuluh adalah sebagai berikut:

```

%% Metode Runge Kutta Orde Sepuluh (RK10A)
%% Untuk Solusi Sistem PDB Orde Satu
%% Model Penyebaran Virus Avian Influenza
%% Efektifitas
%% =====
%% y1'=A*a+(1-c)*D-f*y1*(y2/A)-d*y1
%% y2'=f*y1*(y2/A)-(d+m)*y2+c*D
%% y3'=B*b-((g*y3*y2)/A)-e*y3+p*y4
%% y4'=((g*y3*y2)/A)-(e+v+p)*y4
%% =====
%% A = y1 + y2 = Jumlah unggas
%% B = y3 + y4 = Jumlah manusia
%% D = Jumlah burung migrasi
%% y1 = Jumlah suspect unggas
%% y2 = Jumlah unggas yang terinfeksi
%% y3 = Jumlah suspect manusia
%% y4 = Jumlah manusia yang terinfeksi
%% =====
%% a = Rata-rata tingkat kelahiran unggas
%% b = Rata-rata tingkat kelahiran manusia
%% c = Peluang infeksi pada unggas migrasi
%% d = Tingkat kematian alami unggas
%% e = Tingkat kematian alami manusia
%% f = Tingkat transmisi infeksi dari unggas ke unggas
%% g = Tingkat transmisi infeksi dari unggas ke manusia
%% m = Tingkat kematian unggas karena infeksi flu burung
%% v = Tingkat kematian manusia karena infeksi flu burung
%% p = Tingkat kesembuhan manusia (per hari)
%% =====
%% Nilai awal
%% y1(0)= ...
%% y2(0)= ...
%% y3(0)= ...
%% y4(0)= ...

```

```

%% =====

clear pack close clc t0=clock;

n=input('Iterasi (i)='); h=input('Ukuran langkah(h)=');
y10=input('jumlah suspect unggas (y1)='); y20=input('Jumlah unggas
yang terinfeksi (y2)='); y30=input('Jumlah suspect manusia (y3)=');
y40=input('Jumlah manusia yang terinfeksi (y4)=');

A=y10+y20; B=y30+y40; a=0.03; b=0.001; c=0.01; d=1/(365*2);
e=1/(365*75); f=0.9; g=0.1; m=0.99; v=0.009; p=1/7; D=10;

t(:,1)=[0;0;0;0]; y(:,1)=[y10;y20;y30;y40]; error1=norm(y(:,1),inf);
j=1; errvec1=[]; fprintf('\n itn error');
fprintf('\n-----> n '); for j=1:n;
    fprintf('\n%3.0f %9.3e',j,error1);

    k_1(:,j)=[A*a+(1-c)*D-f*y(1,j)*(y(2,j)/A)-d*y(1,j);
              f*y(1,j)*(y(2,j)/A)-(d+m)*y(2,j)+c*D;
              B*b-((g*y(3,j)*y(2,j))/A)-e*y(3,j)+p*y(4,j);
              ((g*y(3,j)*y(2,j))/A)-(e+v+p)*y(4,j)];

    k_2(:,j)=[A*a+(1-c)*D-f*(y(1,j)+h/9*k_1(1,j))*((y(2,j)+h/9*
              k_1(2,j))/A)-d*(y(1,j)+h/9*k_1(1,j));f*(y(1,j)+h/9*k_1(1,j))*
              ((y(2,j)+h/9*k_1(2,j))/A)-(d+m)*(y(2,j)+h/9*k_1(2,j))+c*D;B*b
              -((g*(y(3,j)+h/9*k_1(3,j))*(y(2,j)+h/9*k_1(2,j)))/A)-e*(y(3,j)
              +h/9*k_1(3,j))+p*(y(4,j)+h/9*k_1(4,j));((g*(y(3,j)+h/9*k_1(3,j))
              *(y(2,j)+h/9*k_1(2,j)))/A)-(e+v+p)*(y(4,j)+h/9*k_1(4,j))];

    k_3(:,j)=[A*a+(1-c)*D-f*(y(1,j)+h/360*(-1617*k_1(1,j)+1697*
              k_2(1,j)))*((y(2,j)+h/360*(-1617*k_1(2,j)+1697*k_2(2,j)))
              /A)-d*(y(1,j)+h/360*(-1617*k_1(1,j)+1697*k_2(1,j)));f*
              (y(1,j)+h/360*(-1617*k_1(1,j)+1697*k_2(1,j)))*((y(2,j)+
              h/360*(-1617*k_1(2,j)+1697*k_2(2,j)))/A)-(d+m)*(y(2,j)+
              h/360*(-1617*k_1(2,j)+1697*k_2(2,j)))+c*D;B*b-((g*(y(3,j)
              +h/360*(-1617*k_1(3,j)+1697*k_2(3,j)))*((y(2,j)+h/360*
              (-1617*k_1(2,j)+1697*k_2(2,j))))/A)-e*(y(3,j)+h/360*(-1617*
              k_1(3,j)+1697*k_2(3,j)))+p*(y(4,j)+h/360*(-1617*k_1(4,j)
              +1697*k_2(4,j)));((g*(y(3,j)+h/360*(-1617*k_1(3,j)+1697*
              k_2(3,j)))*((y(2,j)+h/360*(-1617*k_1(2,j)+1697*k_2(2,j))))
              /A)-(e+v+p)*(y(4,j)+h/360*(-1617*k_1(4,j)+1697*k_2(4,j)))]];

```

$$k_4(:, j) = [A*a + (1-c)*D - f*(y(1, j) + h/1080*(2501*k_1(1, j) + 1000*k_2(1, j) - 3141*k_3(1, j))) * ((y(2, j) + h/1080*(2501*k_1(2, j) + 1000*k_2(2, j) - 3141*k_3(2, j)))) / A - d*(y(1, j) + h/1080*(2501*k_1(1, j) + 1000*k_2(1, j) - 3141*k_3(1, j))) ; f*(y(1, j) + h/1080*(2501*k_1(1, j) + 1000*k_2(1, j) - 3141*k_3(1, j))) * ((y(2, j) + h/1080*(2501*k_1(2, j) + 1000*k_2(2, j) - 3141*k_3(2, j)))) / A - (d+m)*(y(2, j) + h/1080*(2501*k_1(2, j) + 1000*k_2(2, j) - 3141*k_3(2, j))) + c*D ; B*b - ((g*(y(3, j) + h/1080*(2501*k_1(3, j) + 1000*k_2(3, j) - 3141*k_3(3, j))) * (y(2, j) + h/1080*(2501*k_1(2, j) + 1000*k_2(2, j) - 3141*k_3(2, j)))) / A - e*(y(3, j) + h/1080*(2501*k_1(3, j) + 1000*k_2(3, j) - 3141*k_3(3, j))) + p*(y(4, j) + h/1080*(2501*k_1(4, j) + 1000*k_2(4, j) - 3141*k_3(4, j))) ; ((g*(y(3, j) + h/1080*(2501*k_1(3, j) + 1000*k_2(3, j) - 3141*k_3(3, j))) * (y(2, j) + h/1080*(2501*k_1(2, j) + 1000*k_2(2, j) - 3141*k_3(2, j)))) / A - (e+v+p)*(y(4, j) + h/1080*(2501*k_1(4, j) + 1000*k_2(4, j) - 3141*k_3(4, j))))] ;$$

$$k_5(:, j) = [A*a + (1-c)*D - f*(y(1, j) + h/8667*(-154197*k_1(1, j) - 8620*k_2(1, j) - 8640*k_3(1, j) + 175309*k_4(1, j))) * ((y(2, j) + h/8667*(-154197*k_1(2, j) - 8620*k_2(2, j) - 8640*k_3(2, j) + 175309*k_4(2, j)))) / A - d*(y(1, j) + h/8667*(-154197*k_1(1, j) - 8620*k_2(1, j) - 8640*k_3(1, j) + 175309*k_4(1, j))) ; f*(y(1, j) + h/8667*(-154197*k_1(1, j) - 8620*k_2(1, j) - 8640*k_3(1, j) + 175309*k_4(1, j))) * ((y(2, j) + h/8667*(-154197*k_1(2, j) - 8620*k_2(2, j) - 8640*k_3(2, j) + 175309*k_4(2, j)))) / A - (d+m)*(y(2, j) + h/8667*(-154197*k_1(2, j) - 8620*k_2(2, j) - 8640*k_3(2, j) + 175309*k_4(2, j))) + c*D ; B*b - ((g*(y(3, j) + h/8667*(-154197*k_1(3, j) - 8620*k_2(3, j) - 8640*k_3(3, j) + 175309*k_4(3, j))) * (y(2, j) + h/8667*(-154197*k_1(2, j) - 8620*k_2(2, j) - 8640*k_3(2, j) + 175309*k_4(2, j)))) / A - e*(y(3, j) + h/8667*(-154197*k_1(3, j) - 8620*k_2(3, j) - 8640*k_3(3, j) + 175309*k_4(3, j))) + p*(y(4, j) + h/8667*(-154197*k_1(4, j) - 8620*k_2(4, j) - 8640*k_3(4, j) + 175309*k_4(4, j))) ; ((g*(y(3, j) + h/8667*(-154197*k_1(3, j) - 8620*k_2(3, j) - 8640*k_3(3, j) + 175309*k_4(3, j))) * (y(2, j) + h/8667*(-154197*k_1(2, j) - 8620*k_2(2, j) - 8640*k_3(2, j) + 175309*k_4(2, j)))) / A - (e+v+p)*(y(4, j) + h/8667*(-154197*k_1(4, j) - 8620*k_2(4, j) - 8640*k_3(4, j) + 175309*k_4(4, j))))] ;$$

$$k_6(:, j) = [A*a + (1-c)*D - f*(y(1, j) + h/5778*(99353*k_1(1, j) - 5700*k_2(1, j) - 5740*k_3(1, j) + 5620*k_4(1, j) - 90323*k_5(1, j))) * ((y(2, j) + h/5778*(99353*k_1(2, j) - 5700*k_2(2, j) - 5740*k_3(2, j) + 5620*k_4(2, j) - 90323*k_5(2, j)))) / A - d*(y(1, j) + h/5778*(99353$$

$$\begin{aligned}
 & *k_1(1, j) - 5700*k_2(1, j) - 5740*k_3(1, j) + 5620*k_4(1, j) - 90323* \\
 & k_5(1, j)); f*(y(1, j) + h/5778*(99353*k_1(1, j) - 5700*k_2(1, j) - \\
 & 5740*k_3(1, j) + 5620*k_4(1, j) - 90323*k_5(1, j))) * ((y(2, j) + h/5778 \\
 & *(99353*k_1(2, j) - 5700*k_2(2, j) - 5740*k_3(2, j) + 5620*k_4(2, j) \\
 & - 90323*k_5(2, j))) / A) - (d+m)*(y(2, j) + h/5778*(99353*k_1(2, j) - \\
 & 5700*k_2(2, j) - 5740*k_3(2, j) + 5620*k_4(2, j) - 90323*k_5(2, j))) \\
 & + c*D; B*b - ((g*(y(3, j) + h/5778*(99353*k_1(3, j) - 5700*k_2(3, j) - \\
 & 5740*k_3(3, j) + 5620*k_4(3, j) - 90323*k_5(3, j))) * (y(2, j) + h/5778 \\
 & *(99353*k_1(2, j) - 5700*k_2(2, j) - 5740*k_3(2, j) + 5620*k_4(2, j) - \\
 & 90323*k_5(2, j)))) / A) - e*(y(3, j) + h/5778*(99353*k_1(3, j) - 5700* \\
 & k_2(3, j) - 5740*k_3(3, j) + 5620*k_4(3, j) - 90323*k_5(3, j))) + p*(y \\
 & (4, j) + h/5778*(99353*k_1(4, j) - 5700*k_2(4, j) - 5740*k_3(4, j) + \\
 & 5620*k_4(4, j) - 90323*k_5(4, j))); ((g*(y(3, j) + h/5778*(99353* \\
 & k_1(3, j) - 5700*k_2(3, j) - 5740*k_3(3, j) + 5620*k_4(3, j) - 90323* \\
 & k_5(3, j))) * (y(2, j) + h/5778*(99353*k_1(2, j) - 5700*k_2(2, j) - \\
 & 5740*k_3(2, j) + 5620*k_4(2, j) - 90323*k_5(2, j)))) / A) - (e+v+p)* \\
 & (y(4, j) + h/5778*(99353*k_1(4, j) - 5700*k_2(4, j) - 5740*k_3(4, j) \\
 & + 5620*k_4(4, j) - 90323*k_5(4, j)))];
 \end{aligned}$$

$$\begin{aligned}
 k_7(:, j) = & [A*a + (1-c)*D - f*(y(1, j) + h/5580*(-16343*k_1(1, j) - 520* \\
 & k_2(1, j) + 5540*k_3(1, j) - 5420*k_4(1, j) - 5560*k_5(1, j) + 26023*k_ \\
 & 6(1, j))) * ((y(2, j) + h/5580*(-16343*k_1(2, j) - 520*k_2(2, j) + 5540* \\
 & k_3(2, j) - 5420*k_4(2, j) - 5560*k_5(2, j) + 26023*k_6(2, j))) / A) - d* \\
 & (y(1, j) + h/5580*(-16343*k_1(1, j) - 520*k_2(1, j) + 5540*k_3(1, j) - \\
 & 5420*k_4(1, j) - 5560*k_5(1, j) + 26023*k_6(1, j))); f*(y(1, j) + h/ \\
 & 5580*(-16343*k_1(1, j) - 520*k_2(1, j) + 5540*k_3(1, j) - 5420*k_4 \\
 & (1, j) - 5560*k_5(1, j) + 26023*k_6(1, j))) * ((y(2, j) + h/5580*(-16343 \\
 & *k_1(2, j) - 520*k_2(2, j) + 5540*k_3(2, j) - 5420*k_4(2, j) - 5560*k_5 \\
 & (2, j) + 26023*k_6(2, j))) / A) - (d+m)*(y(2, j) + h/5580*(-16343*k_1 \\
 & (2, j) - 520*k_2(2, j) + 5540*k_3(2, j) - 5420*k_4(2, j) - 5560*k_5(2, j) \\
 & + 26023*k_6(2, j))) + c*D; B*b - ((g*(y(3, j) + h/5580*(-16343*k_1(3, j) \\
 & - 520*k_2(3, j) + 5540*k_3(3, j) - 5420*k_4(3, j) - 5560*k_5(3, j) + 26023 \\
 & *k_6(3, j))) * (y(2, j) + h/5580*(-16343*k_1(2, j) - 520*k_2(2, j) + 5540 \\
 & *k_3(2, j) - 5420*k_4(2, j) - 5560*k_5(2, j) + 26023*k_6(2, j)))) / A) - e* \\
 & (y(3, j) + h/5580*(-16343*k_1(3, j) - 520*k_2(3, j) + 5540*k_3(3, j) - \\
 & 5420*k_4(3, j) - 5560*k_5(3, j) + 26023*k_6(3, j))) + p*(y(4, j) + h/5580 \\
 & *(-16343*k_1(4, j) - 520*k_2(4, j) + 5540*k_3(4, j) - 5420*k_4(4, j) - \\
 & 5560*k_5(4, j) + 26023*k_6(4, j))); (g*(y(3, j) + h/5580*(-16343* \\
 & k_1(3, j) - 520*k_2(3, j) + 5540*k_3(3, j) - 5420*k_4(3, j) - 5560*k_5 \\
 & (3, j) + 26023*k_6(3, j))) * (y(2, j) + h/5580*(-16343*k_1(2, j) - 520* \\
 & k_2(2, j) + 5540*k_3(2, j) - 5420*k_4(2, j) - 5560*k_5(2, j) + 26023*k_6
 \end{aligned}$$

$$(2, j))) / A) - (e+v+p) * (y(4, j) + h / 5580 * (-16343 * k_1(4, j) - 520 * k_2(4, j) + 5540 * k_3(4, j) - 5420 * k_4(4, j) - 5560 * k_5(4, j) + 26023 * k_6(4, j))))];$$

$$k_8(:, j) = [A * a + (1 - c) * D - f * (y(1, j) + h / 450 * (11546 * k_1(1, j) + 445 * k_2(1, j) - 440 * k_3(1, j) + 420 * k_4(1, j) + 320 * k_5(1, j) + 420 * k_6(1, j) - 12361 * k_7(1, j))) * ((y(2, j) + h / 450 * (11546 * k_1(2, j) + 445 * k_2(2, j) - 440 * k_3(2, j) + 420 * k_4(2, j) + 320 * k_5(2, j) + 420 * k_6(2, j) - 12361 * k_7(2, j)))) / A) - d * (y(1, j) + h / 450 * (11546 * k_1(1, j) + 445 * k_2(1, j) - 440 * k_3(1, j) + 420 * k_4(1, j) + 320 * k_5(1, j) + 420 * k_6(1, j) - 12361 * k_7(1, j)))]; f * (y(1, j) + h / 450 * (11546 * k_1(1, j) + 445 * k_2(1, j) - 440 * k_3(1, j) + 420 * k_4(1, j) + 320 * k_5(1, j) + 420 * k_6(1, j) - 12361 * k_7(1, j))) * ((y(2, j) + h / 450 * (11546 * k_1(2, j) + 445 * k_2(2, j) - 440 * k_3(2, j) + 420 * k_4(2, j) + 320 * k_5(2, j) + 420 * k_6(2, j) - 12361 * k_7(2, j)))) / A) - (d+m) * (y(2, j) + h / 450 * (11546 * k_1(2, j) + 445 * k_2(2, j) - 440 * k_3(2, j) + 420 * k_4(2, j) + 320 * k_5(2, j) + 420 * k_6(2, j) - 12361 * k_7(2, j))) + c * D; B * b - ((g * (y(3, j) + h / 450 * (11546 * k_1(3, j) + 445 * k_2(3, j) - 440 * k_3(3, j) + 420 * k_4(3, j) + 320 * k_5(3, j) + 420 * k_6(3, j) - 12361 * k_7(3, j))) * (y(2, j) + h / 450 * (11546 * k_1(2, j) + 445 * k_2(2, j) - 440 * k_3(2, j) + 420 * k_4(2, j) + 320 * k_5(2, j) + 420 * k_6(2, j) - 12361 * k_7(2, j)))) / A) - e * (y(3, j) + h / 450 * (11546 * k_1(3, j) + 445 * k_2(3, j) - 440 * k_3(3, j) + 420 * k_4(3, j) + 320 * k_5(3, j) + 420 * k_6(3, j) - 12361 * k_7(3, j))) + p * (y(4, j) + h / 450 * (11546 * k_1(4, j) + 445 * k_2(4, j) - 440 * k_3(4, j) + 420 * k_4(4, j) + 320 * k_5(4, j) + 420 * k_6(4, j) - 12361 * k_7(4, j)))]; ((g * (y(3, j) + h / 450 * (11546 * k_1(3, j) + 445 * k_2(3, j) - 440 * k_3(3, j) + 420 * k_4(3, j) + 320 * k_5(3, j) + 420 * k_6(3, j) - 12361 * k_7(3, j))) * (y(2, j) + h / 450 * (11546 * k_1(2, j) + 445 * k_2(2, j) - 440 * k_3(2, j) + 420 * k_4(2, j) + 320 * k_5(2, j) + 420 * k_6(2, j) - 12361 * k_7(2, j)))) / A) - (e+v+p) * (y(4, j) + h / 450 * (11546 * k_1(4, j) + 445 * k_2(4, j) - 440 * k_3(4, j) + 420 * k_4(4, j) + 320 * k_5(4, j) + 420 * k_6(4, j) - 12361 * k_7(4, j))))];$$

$$k_9(:, j) = [A * a + (1 - c) * D - f * (y(1, j) + h / 4770 * (-1281 * k_1(1, j) + 4720 * k_2(1, j) + 390 * k_3(1, j) + 1280 * k_4(1, j) - 2860 * k_5(1, j) + 2700 * k_6(1, j) - 4762 * k_7(1, j) + 4053 * k_8(1, j))) * ((y(2, j) + h / 4770 * (-1281 * k_1(2, j) + 4720 * k_2(2, j) + 390 * k_3(2, j) + 1280 * k_4(2, j) - 2860 * k_5(2, j) + 2700 * k_6(2, j) - 4762 * k_7(2, j) + 4053 * k_8(2, j)))) / A) - d * (y(1, j) + h / 4770 * (-1281 * k_1(1, j) + 4720 * k_2(1, j) + 390 * k_3(1, j) + 1280 * k_4(1, j) - 2860 * k_5(1, j) + 2700 * k_6(1, j) - 4762 * k_7(1, j) + 4053 * k_8(1, j)))]; f * (y(1, j) + h / 4770 * (-1281 * k_1(1, j) + 4720 * k_2(1, j) + 390 * k_3(1, j) + 1280 * k_4(1, j) - 2860 * k_5(1, j) + 2700 * k_6(1, j) - 4762 * k_7(1, j) + 4053 * k_8(1, j)))];$$

$$\begin{aligned}
 & -4762*k_7(1, j)+4053*k_8(1, j)))*((y(2, j)+h/4770*(-1281*k_1(2, j)+4720*k_2(2, j)+390*k_3(2, j)+1280*k_4(2, j)-2860*k_5(2, j)+2700*k_6(2, j)-4762*k_7(2, j)+4053*k_8(2, j)))/A)-(d+m)*(y(2, j)+h/4770*(-1281*k_1(2, j)+4720*k_2(2, j)+390*k_3(2, j)+1280*k_4(2, j)-2860*k_5(2, j)+2700*k_6(2, j)-4762*k_7(2, j)+4053*k_8(2, j)))+c*D;B*b-((g*(y(3, j)+h/4770*(-1281*k_1(3, j)+4720*k_2(3, j)+390*k_3(3, j)+1280*k_4(3, j)-2860*k_5(3, j)+2700*k_6(3, j)-4762*k_7(3, j)+4053*k_8(3, j)))*(y(2, j)+h/4770*(-1281*k_1(2, j)+4720*k_2(2, j)+390*k_3(2, j)+1280*k_4(2, j)-2860*k_5(2, j)+2700*k_6(2, j)-4762*k_7(2, j)+4053*k_8(2, j)))/A)-e*(y(3, j)+h/4770*(-1281*k_1(3, j)+4720*k_2(3, j)+390*k_3(3, j)+1280*k_4(3, j)-2860*k_5(3, j)+2700*k_6(3, j)-4762*k_7(3, j)+4053*k_8(3, j)))+p*(y(4, j)+h/4770*(-1281*k_1(4, j)+4720*k_2(4, j)+390*k_3(4, j)+1280*k_4(4, j)-2860*k_5(4, j)+2700*k_6(4, j)-4762*k_7(4, j)+4053*k_8(4, j))));((g*(y(3, j)+h/4770*(-1281*k_1(3, j)+4720*k_2(3, j)+390*k_3(3, j)+1280*k_4(3, j)-2860*k_5(3, j)+2700*k_6(3, j)-4762*k_7(3, j)+4053*k_8(3, j)))*(y(2, j)+h/4770*(-1281*k_1(2, j)+4720*k_2(2, j)+390*k_3(2, j)+1280*k_4(2, j)-2860*k_5(2, j)+2700*k_6(2, j)-4762*k_7(2, j)+4053*k_8(2, j)))/A)-(e+v+p)*(y(4, j)+h/4770*(-1281*k_1(4, j)+4720*k_2(4, j)+390*k_3(4, j)+1280*k_4(4, j)-2860*k_5(4, j)+2700*k_6(4, j)-4762*k_7(4, j)+4053*k_8(4, j))));
 \end{aligned}$$

$$\begin{aligned}
 k_{10}(:, j) &= [A*a+(1-c)*D-f*(y(1, j)+h/1800*(-2500*k_1(1, j)+1000*k_2(1, j)+120*k_3(1, j)+1200*k_4(1, j)-600*k_5(1, j)+240*k_6(1, j)+1800*k_7(1, j)+540*k_8(1, j)))*((y(2, j)+h/1800*(-2500*k_1(2, j)+1000*k_2(2, j)+120*k_3(2, j)+1200*k_4(2, j)-600*k_5(2, j)+240*k_6(2, j)+1800*k_7(2, j)+540*k_8(2, j)))/A)-d*(y(1, j)+h/1800*(-2500*k_1(1, j)+1000*k_2(1, j)+120*k_3(1, j)+1200*k_4(1, j)-600*k_5(1, j)+240*k_6(1, j)+1800*k_7(1, j)+540*k_8(1, j)))); \\
 & f*(y(1, j)+h/1800*(-2500*k_1(1, j)+1000*k_2(1, j)+120*k_3(1, j)+1200*k_4(1, j)-600*k_5(1, j)+240*k_6(1, j)+1800*k_7(1, j)+540*k_8(1, j)))*((y(2, j)+h/1800*(-2500*k_1(2, j)+1000*k_2(2, j)+120*k_3(2, j)+1200*k_4(2, j)-600*k_5(2, j)+240*k_6(2, j)+1800*k_7(2, j)+540*k_8(2, j)))/A)-(d+m)*(y(2, j)+h/1800*(-2500*k_1(2, j)+1000*k_2(2, j)+120*k_3(2, j)+1200*k_4(2, j)-600*k_5(2, j)+240*k_6(2, j)+1800*k_7(2, j)+540*k_8(2, j)))+c*D;B*b-((g*(y(3, j)+h/1800*(-2500*k_1(3, j)+1000*k_2(3, j)+120*k_3(3, j)+1200*k_4(3, j)-600*k_5(3, j)+240*k_6(3, j)+1800*k_7(3, j)+540*k_8(3, j)))*(y(2, j)+h/1800*(-2500*k_1(2, j)+1000*k_2(2, j)+120*k_3(2, j)+1200*k_4(2, j)-600*k_5(2, j)+240*k_6(2, j)+1800*k_7(2, j)+540*k_8(2, j))
 \end{aligned}$$

```

    ))/A)-e*(y(3,j)+h/1800*(-2500*k_1(3,j)+1000*k_2(3,j)+120*k_3
    (3,j)+1200*k_4(3,j)-600*k_5(3,j)+240*k_6(3,j)+1800*k_7(3,j)
    +540*k_8(3,j)))+p*(y(4,j)+h/1800*(-2500*k_1(4,j)+1000*k_2
    (4,j)+120*k_3(4,j)+1200*k_4(4,j)-600*k_5(4,j)+240*k_6(4,j)
    +1800*k_7(4,j)+540*k_8(4,j)));((g*(y(3,j)+h/1800*(-2500*
    k_1(3,j)+1000*k_2(3,j)+120*k_3(3,j)+1200*k_4(3,j)-600*k_5
    (3,j)+240*k_6(3,j)+1800*k_7(3,j)+540*k_8(3,j)))*(y(2,j)+h/
    1800*(-2500*k_1(2,j)+1000*k_2(2,j)+120*k_3(2,j)+1200*k_4
    (2,j)-600*k_5(2,j)+240*k_6(2,j)+1800*k_7(2,j)+540*k_8(2,j)
    )))/A)-(e+v+p)*(y(4,j)+h/1800*(-2500*k_1(4,j)+1000*k_2(4,j)
    +120*k_3(4,j)+1200*k_4(4,j)-600*k_5(4,j)+240*k_6(4,j)+1800
    *k_7(4,j)+540*k_8(4,j)))]];

y(:,j+1)=[y(:,j)+h/89600*(2857*k_1(:,j)+15741*k_2(:,j)+1080*
k_3(:,j)+19344*k_4(:,j)+5778*k_5(:,j)+5778*k_6(:,j)+19344*k_7
(:,j)+1080*k_8(:,j)+15741*k_9(:,j)+2857*k_10(:,j)))]];

error1=norm(y(:,j+1)-y(:,j),inf);
t(:,j+1)=t(:,1)+h*[j:j:j:j];
j=j+1;
errvec1=[errvec1,error1];
end

subplot(2,2,1),plot(t(1,:),y(1,:), 'b'),xlabel('waktu'),
ylabel('Populasi unggas yang sehat')
subplot(2,2,2),plot(t(2,:),y(2,:), 'b'),xlabel('waktu'),
ylabel('Populasi unggas yang terinfeksi')
subplot(2,2,3),plot(t(3,:),y(3,:), 'b'),xlabel('waktu'),
ylabel('Populasi manusia yang sehat')
subplot(2,2,4),plot(t(4,:),y(4,:), 'b'),xlabel('waktu'),
ylabel('Populasi manusia yang terinfeksi')

figure

plot([1:j],[error1 errvec1], 'b') xlabel('iterasi'),ylabel('error')

format long error=[error1] jumlah_iterasi=j
waktu_dalam_detik=etime(clock,t0)

```

LAMPIRAN B. Format Pemrograman Metode Runge-Kutta Orde Sepuluh

• **Format Programming Efisiensi**

Format *programming* yang dikembangkan dari Metode Runge-Kutta orde sepuluh adalah sebagai berikut:

```

%% Metode Runge Kutta Orde Sepuluh (RK10B)
%% Untuk Solusi Sistem PDB Orde Satu
%% Model Penyebaran Virus Avian Influenza
%% Efisiensi
%% =====
%% y1'=A*a+(1-c)*D-f*y1*(y2/A)-d*y1
%% y2'=f*y1*(y2/A)-(d+m)*y2+c*D
%% y3'=B*b-((g*y3*y2)/A)-e*y3+p*y4
%% y4'=((g*y3*y2)/A)-(e+v+p)*y4
%% =====
%% A = y1 + y2 = Jumlah unggas
%% B = y3 + y4 = Jumlah manusia
%% D = Jumlah burung migrasi
%% y1 = Jumlah suspect unggas
%% y2 = Jumlah unggas yang terinfeksi
%% y3 = Jumlah suspect manusia
%% y4 = Jumlah manusia yang terinfeksi
%% =====
%% a = Rata-rata tingkat kelahiran unggas
%% b = Rata-rata tingkat kelahiran manusia
%% c = Peluang infeksi pada unggas migrasi
%% d = Tingkat kematian alami unggas
%% e = Tingkat kematian alami manusia
%% f = Tingkat transmisi infeksi dari unggas ke unggas
%% g = Tingkat transmisi infeksi dari unggas ke manusia
%% m = Tingkat kematian unggas karena infeksi flu burung
%% v = Tingkat kematian manusia karena infeksi flu burung
%% p = Tingkat kesembuhan manusia (per hari)
%% =====
%% Nilai awal
%% y1(0)= ...
%% y2(0)= ...
%% y3(0)= ...
%% y4(0)= ...
%% =====

```



```

clear pack close clc t0=clock; flops(0);

tol=input('Toleransi (e)='); h=0.01; y10=300; y20=165; y30=60;
y40=7;

A=y10+y20; B=y30+y40; a=0.03; b=0.001; c=0.01; d=1/(365*2);
e=1/(365*75); f=0.9; g=0.1; m=0.99; v=0.009; p=1/7; D=10;

t(:,1)=[0;0;0;0]; y(:,1)=[y10;y20;y30;y40]; error1=norm(y(:,1),inf);
j=1; errvec1=[]; fprintf('\n itn error');
fprintf('\n-----> n '); while error1>=tol
    fprintf('\n%3.0f %9.3e',j,error1);

    k_1(:,j)=[A*a+(1-c)*D-f*y(1,j)*(y(2,j)/A)-d*y(1,j);
              f*y(1,j)*(y(2,j)/A)-(d+m)*y(2,j)+c*D;
              B*b-((g*y(3,j)*y(2,j))/A)-e*y(3,j)+p*y(4,j);
              ((g*y(3,j)*y(2,j))/A)-(e+v+p)*y(4,j)];

    k_2(:,j)=[A*a+(1-c)*D-f*(y(1,j)+h/9*k_1(1,j))*((y(2,j)+h/9
              *k_1(2,j))/A)-d*(y(1,j)+h/9*k_1(1,j));f*(y(1,j)+h/9*
              k_1(1,j))*((y(2,j)+h/9*k_1(2,j))/A)-(d+m)*(y(2,j)+h/9
              *k_1(2,j))+c*D;B*b-((g*(y(3,j)+h/9*k_1(3,j))*(y(2,j)+
              h/9*k_1(2,j)))/A)-e*(y(3,j)+h/9*k_1(3,j))+p*(y(4,j)+h/
              9*k_1(4,j));((g*(y(3,j)+h/9*k_1(3,j))*(y(2,j)+h/9*k_1
              2,j)))/A)-(e+v+p)*(y(4,j)+h/9*k_1(4,j))];

    k_3(:,j)=[A*a+(1-c)*D-f*(y(1,j)+h/45*(-1156*k_1(1,j)+1166*
              k_2(1,j)))*((y(2,j)+h/45*(-1156*k_1(2,j)+1166*k_2(2,j)))
              /A)-d*(y(1,j)+h/45*(-1156*k_1(1,j)+1166*k_2(1,j)));f*
              (y(1,j)+h/45*(-1156*k_1(1,j)+1166*k_2(1,j)))*((y(2,j)+
              h/45*(-1156*k_1(2,j)+1166*k_2(2,j)))/A)-(d+m)*(y(2,j)+
              h/45*(-1156*k_1(2,j)+1166*k_2(2,j)))+c*D;B*b-((g*(y(3,
              j)+h/45*(-1156*k_1(3,j)+1166*k_2(3,j)))*((y(2,j)+h/45*
              (-1156*k_1(2,j)+1166*k_2(2,j))))/A)-e*(y(3,j)+h/45*
              (-1156*k_1(3,j)+1166*k_2(3,j)))+p*(y(4,j)+h/45*(-1156
              *k_1(4,j)+1166*k_2(4,j)));((g*(y(3,j)+h/45*(-1156*k_1
              (3,j)+1166*k_2(3,j)))*((y(2,j)+h/45*(-1156*k_1(2,j)+1166
              *k_2(2,j))))/A)-(e+v+p)*(y(4,j)+h/45*(-1156*k_1(4,j)+
              1166*k_2(4,j)))]];

```

$$\begin{aligned}
 k_4(:, j) = & [A*a+(1-c)*D-f*(y(1, j)+h/4836*(13645*k_1(1, j)-12033 \\
 & *k_3(1, j)))*((y(2, j)+h/4836*(13645*k_1(2, j)-12033*k_3(2, \\
 & j)))/A)-d*(y(1, j)+h/4836*(13645*k_1(1, j)-12033*k_3(1, j)))] ; \\
 & f*(y(1, j)+h/4836*(13645*k_1(1, j)-12033*k_3(1, j)))*((y(2, j) \\
 & +h/4836*(13645*k_1(2, j)-12033*k_3(2, j)))/A)-(d+m)*(y(2, j)+ \\
 & h/4836*(13645*k_1(2, j)-12033*k_3(2, j)))+c*D;B*b-((g*(y(3, j) \\
 & +h/4836*(13645*k_1(3, j)-12033*k_3(3, j)))*y(2, j))/A)-e*(y(3 \\
 & , j)+h/4836*(13645*k_1(3, j)-12033*k_3(3, j)))+p*(y(4, j)+h/ \\
 & 4836*(13645*k_1(4, j)-12033*k_3(4, j)))] ; ((g*(y(3, j)+h/4836* \\
 & (13645*k_1(3, j)-12033*k_3(3, j)))*y(2, j)+h/4836*(13645*k_1 \\
 & (2, j)-12033*k_3(2, j)))/A)-(e+v+p)*(y(4, j)+h/4836*(13645* \\
 & k_1(4, j)-12033*k_3(4, j)))] ;
 \end{aligned}$$

$$\begin{aligned}
 k_5(:, j) = & [A*a+(1-c)*D-f*(y(1, j)+h/2889*(-54136*k_1(1, j)+55420* \\
 & k_4(1, j)))*((y(2, j)+h/2889*(-54136*k_1(2, j)+55420*k_4(2, j)) \\
 &)/A)-d*(y(1, j)+h/2889*(-54136*k_1(1, j)+55420*k_4(1, j)))] ; f* \\
 & (y(1, j)+h/2889*(-54136*k_1(1, j)+55420*k_4(1, j)))*((y(2, j)+ \\
 & h/2889*(-54136*k_1(2, j)+55420*k_4(2, j)))/A)-(d+m)*(y(2, j)+ \\
 & h/2889*(-54136*k_1(2, j)+55420*k_4(2, j)))+c*D;B*b-((g*(y(3, \\
 & j)+h/2889*(-54136*k_1(3, j)+55420*k_4(3, j)))*y(2, j)+h/2889 \\
 & *(-54136*k_1(2, j)+55420*k_4(2, j)))/A)-e*(y(3, j)+h/2889* \\
 & (-54136*k_1(3, j)+55420*k_4(3, j)))+p*(y(4, j)+h/2889*(-54136 \\
 & *k_1(4, j)+55420*k_4(4, j)))] ; ((g*(y(3, j)+h/2889*(-54136*k_1 \\
 & (3, j)+55420*k_4(3, j)))*y(2, j)+h/2889*(-54136*k_1(2, j)+55420 \\
 & *k_4(2, j)))/A)-(e+v+p)*(y(4, j)+h/2889*(-54136*k_1(4, j)+ \\
 & 55420*k_4(4, j)))] ;
 \end{aligned}$$

$$\begin{aligned}
 k_6(:, j) = & [A*a+(1-c)*D-f*(y(1, j)+h/963*(20405*k_1(1, j)-19870* \\
 & k_5(1, j)))*((y(2, j)+h/963*(20405*k_1(2, j)-19870*k_5(2, j)) \\
 &)/A)-d*(y(1, j)+h/963*(20405*k_1(1, j)-19870*k_5(1, j)))] ; f*(y \\
 & (1, j)+h/963*(20405*k_1(1, j)-19870*k_5(1, j)))*((y(2, j)+h/ \\
 & 963*(20405*k_1(2, j)-19870*k_5(2, j)))/A)-(d+m)*(y(2, j)+h/ \\
 & 963*(20405*k_1(2, j)-19870*k_5(2, j)))+c*D;B*b-((g*(y(3, j)+ \\
 & h/963*(20405*k_1(3, j)-19870*k_5(3, j)))*y(2, j)+h/963* \\
 & (20405*k_1(2, j)-19870*k_5(2, j)))/A)-e*(y(3, j)+h/963* \\
 & (20405*k_1(3, j)-19870*k_5(3, j)))+p*(y(4, j)+h/963*(20405 \\
 & *k_1(4, j)-19870*k_5(4, j)))] ; ((g*(y(3, j)+h/963*(20405*k_1 \\
 & (3, j)-19870*k_5(3, j)))*y(2, j)+h/963*(20405*k_1(2, j)- \\
 & 19870*k_5(2, j)))/A)-(e+v+p)*(y(4, j)+h/963*(20405*k_1 \\
 & (4, j)-19870*k_5(4, j)))] ;
 \end{aligned}$$

$$\begin{aligned}
k_7(:, j) = & [A*a+(1-c)*D-f*(y(1, j)+h/1209*(-5476*k_1(1, j)+6282* \\
& k_6(1, j)))*((y(2, j)+h/1209*(-5476*k_1(2, j)+6282*k_6(2, j))) \\
& /A)-d*(y(1, j)+h/1209*(-5476*k_1(1, j)+6282*k_6(1, j)));f*(y \\
& (1, j)+h/1209*(-5476*k_1(1, j)+6282*k_6(1, j)))*((y(2, j)+h/ \\
& 1209*(-5476*k_1(2, j)+6282*k_6(2, j)))/A)-(d+m)*(y(2, j)+h/ \\
& 1209*(-5476*k_1(2, j)+6282*k_6(2, j)))+c*D;B*b-((g*(y(3, j) \\
& +h/1209*(-5476*k_1(3, j)+6282*k_6(3, j)))*((y(2, j)+h/1209* \\
& (-5476*k_1(2, j)+6282*k_6(2, j))))/A)-e*(y(3, j)+h/1209* \\
& (-5476*k_1(3, j)+6282*k_6(3, j)))+p*(y(4, j)+h/1209*(-5476 \\
& *k_1(4, j)+6282*k_6(4, j)));((g*(y(3, j)+h/1209*(-5476* \\
& k_1(3, j)+6282*k_6(3, j)))*((y(2, j)+h/1209*(-5476*k_1(2, j) \\
& +6282*k_6(2, j))))/A)-(e+v+p)*(y(4, j)+h/1209*(-5476*k_1 \\
& (4, j)+6282*k_6(4, j)))]];
\end{aligned}$$

$$\begin{aligned}
k_8(:, j) = & [A*a+(1-c)*D-f*(y(1, j)+h/270*(10841*k_1(1, j)-10631* \\
& k_7(1, j)))*((y(2, j)+h/270*(10841*k_1(2, j)-10631*k_7(2, j))) \\
&)/A)-d*(y(1, j)+h/270*(10841*k_1(1, j)-10631*k_7(1, j)));f*(\\
& y(1, j)+h/270*(10841*k_1(1, j)-10631*k_7(1, j)))*((y(2, j)+h/ \\
& 270*(10841*k_1(2, j)-10631*k_7(2, j)))/A)-(d+m)*(y(2, j)+h/ \\
& 270*(10841*k_1(2, j)-10631*k_7(2, j)))+c*D;B*b-((g*(y(3, j)+ \\
& h/270*(10841*k_1(3, j)-10631*k_7(3, j)))*((y(2, j)+h/270* \\
& (10841*k_1(2, j)-10631*k_7(2, j))))/A)-e*(y(3, j)+h/270* \\
& (10841*k_1(3, j)-10631*k_7(3, j)))+p*(y(4, j)+h/270*(10841 \\
& *k_1(4, j)-10631*k_7(4, j)));((g*(y(3, j)+h/270*(10841*k_1 \\
& (3, j)-10631*k_7(3, j)))*((y(2, j)+h/270*(10841*k_1(2, j)- \\
& 10631*k_7(2, j))))/A)-(e+v+p)*(y(4, j)+h/270*(10841*k_1 \\
& (4, j)-10631*k_7(4, j)))]];
\end{aligned}$$

$$\begin{aligned}
k_9(:, j) = & [A*a+(1-c)*D-f*(y(1, j)+h/5247*(-80*k_1(1, j)+4744* \\
& k_8(1, j)))*((y(2, j)+h/5247*(-80*k_1(2, j)+4744*k_8(2, j))) \\
&)/A)-d*(y(1, j)+h/5247*(-80*k_1(1, j)+4744*k_8(1, j))); \\
& f*(y(1, j)+h/5247*(-80*k_1(1, j)+4744*k_8(1, j)))*((y(2, j) \\
& +h/5247*(-80*k_1(2, j)+4744*k_8(2, j)))/A)-(d+m)*(y(2, j)+ \\
& h/5247*(-80*k_1(2, j)+4744*k_8(2, j)))+c*D;B*b-((g*(y(3, j) \\
& +h/5247*(-80*k_1(3, j)+4744*k_8(3, j)))*((y(2, j)+h/5247*(-80 \\
& *k_1(2, j)+4744*k_8(2, j))))/A)-e*(y(3, j)+h/5247*(-80*k_1 \\
& (3, j)+4744*k_8(3, j)))+p*(y(4, j)+h/5247*(-80*k_1(4, j)+4744 \\
& *k_8(4, j)));((g*(y(3, j)+h/5247*(-80*k_1(3, j)+4744*k_8(3, j) \\
&))*(y(2, j)+h/5247*(-80*k_1(2, j)+4744*k_8(2, j))))/A)-(e+v+p) \\
& *(y(4, j)+h/5247*(-80*k_1(4, j)+4744*k_8(4, j)))]];
\end{aligned}$$

```

k_10(:,j)=[A*a+(1-c)*D-f*(y(1,j)+h*k_1(1,j))*((y(2,j)+h*
k_1(2,j))/A)-d*(y(1,j)+h*k_1(1,j));f*(y(1,j)+h*k_1(1,j)
)*((y(2,j)+h*k_1(2,j))/A)-(d+m)*(y(2,j)+h*k_1(2,j))+c*D;
B*b-((g*(y(3,j)+h*k_1(3,j))*(y(2,j)+h*k_1(2,j)))/A)-e*
(y(3,j)+h*k_1(3,j))+p*(y(4,j)+h*k_1(4,j));((g*(y(3,j)+h*
k_1(3,j))*(y(2,j)+h*k_1(2,j)))/A)-(e+v+p)*(y(4,j)+h*
k_1(4,j))];

y(:,j+1)=[y(:,j)+h/89600*(2857*k_1(:,j)+15741*k_2(:,j)+1080*
k_3(:,j)+19344*k_4(:,j)+5778*k_5(:,j)+5778*k_6(:,j)+19344*
k_7(:,j)+1080*k_8(:,j)+15741*k_9(:,j)+2857*k_10(:,j))];
error1=norm(y(:,j+1)-y(:,j),inf);
t(:,j+1)=t(:,1)+h*[j:j:j];
j=j+1;
errvec1=[errvec1,error1];
end

subplot(2,2,1),plot(t(1,:),y(1,:), 'r'),xlabel('waktu'),
ylabel('Populasi unggas yang sehat')
subplot(2,2,2),plot(t(2,:),y(2,:), 'r'),xlabel('waktu'),
ylabel('Populasi unggas yang terinfeksi')
subplot(2,2,3),plot(t(3,:),y(3,:), 'r'),xlabel('waktu'),
ylabel('Populasi manusia yang sehat')
subplot(2,2,4),plot(t(4,:),y(4,:), 'r'),xlabel('waktu'),
ylabel('Populasi manusia yang terinfeksi') figure

plot([1:j],[error1 errvec1], 'r') xlabel('iterasi'),ylabel('error')

format long error=[error1] jumlah_iterasi=j jumlah_operasi=flops
waktu_dalam_detik=etime(clock,t0)

```