

**SOLUSI NUMERIK MODEL *PREDATOR-PREY*
MENGUNAKAN METODE JARINGAN FUNGSI RADIAL BASIS**

SKRIPSI

**OLEH
MUHAMMAD THAHIRUDDIN
NIM. 12610096**



**JURUSAN MATEMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2017**

**SOLUSI NUMERIK MODEL *PREDATOR-PREY*
MENGUNAKAN METODE JARINGAN FUNGSI RADIAL BASIS**

SKRIPSI

**Diajukan Kepada
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang
untuk Memenuhi Salah Satu Persyaratan dalam
Memperoleh Gelar Sarjana Sains (S.Si)**

**Oleh
Muhammad Thahiruddin
NIM. 12610096**

**JURUSAN MATEMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2017**

**SOLUSI NUMERIK MODEL PREDATOR-*PREY*
MENGUNAKAN METODE JARINGAN FUNGSI RADIAL BASIS**

SKRIPSI

Oleh
Muhammad Thahiruddin
NIM. 12610096

Telah Diperiksa dan Disetujui untuk Diuji
Tanggal 31 Mei 2016

Pembimbing I,



Mohammad Jamhuri, M.Si
NIP. 19810502 200501 1 004

Pembimbing II,



Ach. Nashiehuudin, M.A
NIP. 19730705 200003 1 001

Mengetahui,

Ketua Jurusan Matematika



Dr. Abdussakir, M.Pd
NIP. 19751006 200312 1 001

**SOLUSI NUMERIK MODEL PREDATOR-PREY
MENGUNAKAN METODE JARINGAN FUNGSI RADIAL BASIS**


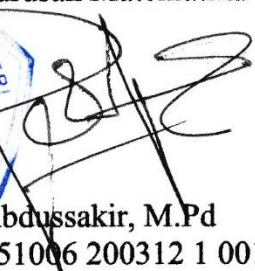
SKRIPSI

Oleh
Muhammad Thahiruddin
NIM. 12610096

Telah Dipertahankan di Depan Dewan Penguji Skripsi
dan Dinyatakan Diterima sebagai Salah Satu Persyaratan
untuk Memperoleh Gelar Sarjana Sains (S.Si)
Tanggal 8 Juni 2016

Penguji Utama : Dr. Usman Pagalay, M.Si
Ketua Penguji : Ari Kusumastuti, M.Pd, M.Si
Sekretaris Penguji : Mohammad Jamhuri, M.Si
Anggota Penguji : Ach. Nashichuddin, M.A

Mengetahui,
Ketua Jurusan Matematika



Dr. Abdussakir, M.Pd
NIP. 19751006 200312 1 001

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini:

Nama : Muhammad Thahiruddin
NIM : 12610096
Jurusan : Matematika
Fakultas : Sains dan Teknologi
Judul Skripsi : Solusi Numerik Model *Predator-Prey* Menggunakan
Metode Jaringan Fungsi Radial Basis

menyatakan dengan sebenarnya bahwa skripsi yang saya tulis ini benar-benar merupakan hasil karya sendiri, bukan merupakan pengambilan data, tulisan, atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar rujukan. Apabila di kemudian hari terbukti atau dapat dibuktikan skripsi ini hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 31 Mei 2016
Yang membuat pernyataan,



Muhammad Thahiruddin
NIM. 12610096

MOTO

دَكَاةٌ بِلاَ اِحْتِشَامٍ اِنْحِطَاطٌ عَظِيْمٌ

“Kecerdasan tanpa kerendahan hati adalah kemunduran besar dalam akhlaq”

(RKH. Ahmad Mahfudz Zayyadi)



PERSEMBAHAN

Skripsi ini penulis persembahkan untuk:

Bapak dan Ibu

yang telah memberikan *wejangan*, kasih sayang, tauladan, doa, serta biaya pendidikan bagi penulis dari lahir sampai umur dewasa ini.

Keluarga besar penulis yang ada di Kadur, Larangan, Kertagena Laok, dan Rombasan yang selalu memberi doa buat penulis.



KATA PENGANTAR

Assalamu'alaikum Warahmatullahi Wabarakatuh

Alhamdulillah, segala puji syukur bagi Allah Swt. atas limpahan rahmat, taufik, hidayah, dan karunia-Nya, sehingga penulis dapat menyelesaikan dengan baik penyusunan skripsi yang berjudul “Solusi Numerik Model *Predator-Prey* Menggunakan Metode Jaringan Fungsi Radial Basis”.

Shalawat serta salam semoga tetap terlimpahkan kepada nabi Muhammad Saw., yang telah menuntun umatnya dari zaman yang gelap ke zaman yang terang benderang yakni agama Islam.

Skripsi ini disusun sebagai salah satu syarat untuk memperoleh gelar sarjana dalam bidang matematika di Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang. Dalam proses penyusunannya tidak mungkin dapat diselesaikan dengan baik tanpa bantuan, bimbingan, serta arahan dari berbagai pihak. Untuk itu ucapan terima kasih penulis sampaikan kepada:

1. Prof. Dr. H. Mudjia Rahardjo, M.Si, selaku rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang.
2. Dr. drh. Bayyinatul Muchtaromah, M.Si, selaku dekan Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang.
3. Dr. Abdussakir, M.Pd, selaku ketua Jurusan Matematika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang.
4. Mohammad Jamhuri, M.Si, selaku dosen pembimbing I yang senantiasa memberikan doa, arahan, nasihat, motivasi dalam melakukan penelitian, serta pengalaman yang berharga kepada penulis.

5. Ach. Nasichuddin, M.A, selaku dosen pembimbing II yang telah memberikan bimbingan, arahan, dan berbagai ilmunya kepada penulis.
6. Segenap sivitas akademika Jurusan Matematika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang terutama seluruh dosen, terima kasih atas segala ilmu dan bimbingannya.
7. Ayah dan Ibu yang selalu memberikan doa, semangat, serta motivasi kepada penulis.
8. Seluruh teman-teman di Jurusan Matematika angkatan 2012 khususnya Matematika C, terima kasih atas kenangan-kenangan indah yang dirajut bersama dalam menggapai cita-cita.
9. Semua pihak yang secara langsung atau tidak langsung telah ikut memberikan bantuan dalam menyelesaikan skripsi ini.

Akhirnya penulis hanya dapat berharap, di balik skripsi ini dapat ditemukan sesuatu yang dapat memberikan manfaat dan wawasan yang lebih luas atau bahkan hikmah bagi penulis, pembaca, dan bagi seluruh mahasiswa.

Wassalamu 'alaikum Warahmatullahi Wabarakatuh

Malang, Mei 2016

Penulis

DAFTAR ISI

HALAMAN JUDUL	
HALAMAN PENGAJUAN	
HALAMAN PERSETUJUAN	
HALAMAN PENGESAHAN	
HALAMAN PERNYATAAN KEASLIAN TULISAN	
HALAMAN MOTO	
HALAMAN PERSEMBAHAN	
KATA PENGANTAR	viii
DAFTAR ISI	x
DAFTAR GAMBAR	xii
DAFTAR LAMPIRAN	xiii
ABSTRAK	xiv
ABSTRACT	xv
ملخص	xvi
BAB I PENDAHULUAN	
1.1 Latar Belakang	1
1.2 Rumusan Masalah	4
1.3 Tujuan Penelitian	4
1.4 Batasan Masalah	5
1.5 Manfaat Penelitian	5
1.6 Metode Penelitian	5
1.7 Sistematika Penulisan	6
BAB II KAJIAN PUSTAKA	
2.1 Jaringan Fungsi Radial Basis	8
2.1.1 Turunan-turunan Fungsi dengan Jaringan Fungsi Radial Basis	11
2.1.2 Fungsi <i>Multiquadrics</i> dan Penentuan <i>Shape-Parameter</i> α^2	12
2.2 Model <i>Predator-Prey</i>	13
2.3 Analisis Galat	16
2.4 Konsep Estimasi dan Aproksimasi dalam Al-Quran	17

BAB III PEMBAHASAN

3.1 Melinierkan Model <i>Predator-Prey</i> dengan Deret Taylor	20
3.2 Mengubah ke Bentuk Jaringan Fungsi Radial Basis	22
3.3 Mengubah ke Bentuk Persamaan Matriks dan Memasukkan Nilai-nilai (t_k) yang Didiskritkan	26
3.4 Menghitung Nilai Bobot w_i dan v_i	28
3.5 Menghitung Solusi Model <i>Predator-Prey</i>	29
3.6 Simulasi	30
3.7 Analisis Galat	39
3.8 Solusi Numerik dalam Pandangan Islam	48

BAB IV PENUTUP

4.1 Kesimpulan	52
4.2 Saran	53

DAFTAR RUJUKAN	54
-----------------------------	----

LAMPIRAN-LAMPIRAN

RIWAYAT HIDUP

DAFTAR GAMBAR

Gambar 3.1 Solusi Numerik Model <i>Predator-Prey</i> dengan $\Delta t = 0,01$ (Simulasi Pertama)	37
Gambar 3.2 Solusi Numerik Model <i>Predator-Prey</i> dengan $\Delta t = 0,01$ (Simulasi Kedua)	39
Gambar 3.3 Galat Solusi Numerik Model <i>Predator-Prey</i> dengan $\Delta t = 0,01$ (Simulasi Pertama)	41
Gambar 3.4 Galat Solusi Numerik Model <i>Predator-Prey</i> dengan $\Delta t = 0,01$ (Simulasi Kedua)	42
Gambar 3.5 Solusi Numerik Model <i>Predator-Prey</i> dengan $\Delta t = 0,02$ (Simulasi Pertama)	43
Gambar 3.6 Solusi Numerik Model <i>Predator-Prey</i> dengan $\Delta t = 0,02$ (Simulasi Kedua)	43
Gambar 3.7 Galat Solusi Numerik Model <i>Predator-Prey</i> dengan $\Delta t = 0,02$ (Simulasi Pertama)	44
Gambar 3.8 Galat Solusi Numerik Model <i>Predator-Prey</i> dengan $\Delta t = 0,02$ (Simulasi Kedua)	44
Gambar 3.9 Solusi Numerik Model <i>Predator-Prey</i> dengan $\Delta t = 0,005$ (Simulasi Pertama)	45
Gambar 3.10 Galat Solusi Numerik Model <i>Predator-Prey</i> dengan $\Delta t = 0,005$ (Simulasi Pertama)	46
Gambar 3.11 Solusi Numerik Model <i>Predator-Prey</i> dengan $\Delta t = 0,005$ (Simulasi Kedua)	46
Gambar 3.12 Galat Solusi Numerik Model <i>Predator-Prey</i> dengan $\Delta t = 0,005$ (Simulasi Kedua)	47

DAFTAR LAMPIRAN

Lampiran 1 <i>M-File</i> Matlab Solusi Numerik Model <i>Predator-Prey</i> (Simulasi Pertama)	56
Lampiran 2 <i>M-File</i> Matlab Solusi Numerik Model <i>Predator-Prey</i> (Simulasi Kedua)	59



ABSTRAK

Thahiruddin, Muhammad. 2016. **Solusi Numerik Model *Predator-Prey* Menggunakan Metode Jaringan Fungsi Radial Basis**. Skripsi. Jurusan Matematika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Mohammad Jamhuri, M.Si. (II) Ach. Nashichuddin, MA.

Kata Kunci: solusi numerik, model *predator-prey*, jaringan fungsi radial basis, nonlinier

Model *predator-prey* merupakan sistem persamaan diferensial biasa nonlinier orde satu. Model *predator-prey* dalam skripsi ini menggambarkan interaksi antara populasi pemangsa dengan populasi mangsanya. Model ini merupakan model *unautonomous* karena perubahan populasinya bergantung pada perubahan waktu, yang mana model ini menggambarkan jumlah dari populasi pemangsa dan populasi mangsa dalam kurun waktu tertentu.

Solusi numerik model *predator-prey* dalam skripsi ini diperoleh dengan menggunakan metode jaringan fungsi radial basis. Dengan metode ini, setiap fungsi dan turunannya dapat didekati secara langsung dengan sebuah fungsi basis jenis *multiquadrics*. Metode yang digunakan adalah metode langsung yaitu dengan menurunkan fungsi basis terhadap variabel bebasnya.

Solusi numerik menggunakan metode jaringan fungsi radial basis khususnya metode langsung yang diperoleh dari penelitian ini menunjukkan keakuratan yang tinggi dengan diperolehnya galat yang cukup kecil dengan galat mutlak maksimum 0,0066 untuk simulasi pertama dan 0,022 pada simulasi kedua. Galat ini terbilang cukup kecil karena galat hanya menunjukkan 0,66% dari nilai awal terkecil untuk model pertama dan 0,5% dari nilai awal terkecil model kedua. Ini menunjukkan bahwa metode jaringan fungsi radial basis cukup efektif dalam mendekati solusi model *predator-prey*.

ABSTRACT

Thahiruddin, Muhammad. 2016. **Numerical Solution of Predator-Prey Model Using Radial Basis Function Networks Method**. Thesis. Department of Mathematics, Faculty of Science and Technology, State Islamic University of Maulana Malik Ibrahim Malang. Supervisors: (I) Mohammad Jamhuri, M.Si. (II) Ach. Nashichuddin, MA.

Kata Kunci: numerical solution, *predator-prey* model, radial basis function networks, nonlinear

Predator-prey model is a nonlinear system of first order ordinary differential equations. Predator-prey model in this paper illustrates the interaction between predator and prey population. This model is a model unautonomous since the changes in populations depend on the time change, which it describes the number of population of the predator and prey populations within a certain time.

The Numerical Solution of predator-prey models in this paper is obtained by applying radial basis function network. With this method, each function and its derivatives can be approached directly by a base function of multiquadrics. The used methode is the direct method by deriving the base function of the independent variables.

Numerical solution using radial basis function network, especially the direct method obtained from this study showed high accuracy by obtaining the error is quite small with a maximum absolute error 0,0066 for the first simulation and 0,022 in the second simulation. This error is quite small, the error is 0,66% of the smallest initial value for the first model and 0,5% of the smallest initial value for the second model. This shows that the radial basis function network method is quite effective in approaching predator-prey model solutions.

ملخص

طاهر الدين، محمد. ٢٠١٦. الحل العددي لنماذج المفترس والفريسة باستخدام شبكة دالة شعاعي أساس. البحث الجامعي. شعبة الرياضيات، كلية العلوم والتكنولوجيا، الجامعة الإسلامية الحكومية مولانا مالك إبراهيم مالانج. المشرف: (١) محمد جمهوري الماجستير. (٢) أحمد ناسح الدين الماجستير.

الكلمات المفتاحية: الحل العددي، نموذج المفترس والفريسة، شبكة دالة شعاعي أساس، غير الخطية

نموذج المفترس والفريسة هو نظام المعادلات التفاضلية العادية غير الخطية من الدرجة الأولى. نموذج المفترس والفريسة في هذا البحث يوضح التفاعل بين السكان المفترس يبلغ عدد سكانها فريسة. هذا النموذج هو نموذج *unautonomous* بسبب التغيرات في السكان تعتمد على تغيير الوقت، الذي يصف عدد من السكان من هذا الحيوان المفترس والفريسة السكان خلال فترة زمنية معينة. يتم الحصول الحل العددية لنماذج المفترس والفريسة في هذا البحث من خلال تطبيق شبكة دالة شعاعي أساس. بهذه الطريقة، كل دالة ومشتقاته يمكن تناولها مباشرة من خلال دالة قاعدة نوع *multiquadrics*. الطريقة المستخدمة هي طريقة المباشر للحد من دالة قاعدة المتغيرات المستقلة. أظهر الحل العددي باستخدام شبكة دالة شعاعي أساس، وخاصة الطريقة المباشرة التي تم الحصول عليها من هذه الدراسة دقة عالية عن طريق الحصول على خطأ صغير جدا مع أقصى الخطأ المطلق "0,0066" لمحاكاة الأولى و "0,022" في محاكاة الثانية. هذا الخطأ هو صغير جدا نظرا ليظهر الخطأ "0,66%" فقط من القيمة الأولية من أصغر لنموذج الأول و "0,5%" من القيمة الأولية من أصغر لنموذج الثاني. هذا يدل على أن طريقة شبكة دالة شعاعي الأساس هو فعال جدا في الاقتراب من الحلول النموذجية المفترس والفريسة.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Manusia adalah makhluk yang penuh dengan keterbatasan. Banyak hal di dunia ini yang tidak dapat manusia lakukan dengan sempurna. Salah satunya ialah dalam hal menghitung. Allah Swt. berfirman dalam al-Quran surat an-Nahl ayat 18

وَأِنْ تَعُدُّوا نِعْمَةَ اللَّهِ لَا تُحْصُوهَا إِنَّ اللَّهَ لَغَفُورٌ رَحِيمٌ^{١٨}

Artinya: “Dan jika kamu menghitung-hitung nikmat Allah, niscaya kamu tak dapat menentukan jumlahnya. Sesungguhnya Allah benar-benar Maha Pengampun lagi Maha Penyayang”(QS. an-Nahl/16:18).

Dalam ayat di atas disebutkan bahwa manusia tidak akan mampu menghitung nikmat Allah Swt. secara terperinci. Hal itu disebabkan karena keterbatasan manusia dalam menghitung. Banyak hal di dunia ini yang tidak sepenuhnya dapat dihitung secara tepat oleh manusia. Salah satunya ialah permasalahan-permasalahan dalam pemodelan matematika.

Pemodelan matematika adalah proses transformasi dari permasalahan-permasalahan sains dan teknik ke persamaan-persamaan matematika. Sehingga permasalahan yang ada diharapkan akan menjadi lebih sederhana dan lebih mudah dicari solusinya. Oleh sebab itu, matematika banyak diterapkan di berbagai disiplin ilmu seperti teknik, ekologi, imunologi, dan fluida (Fitriya, 2011).

Dengan adanya keterbatasan manusia, tidak semua model dapat dicari solusi pastinya (eksaknya). Oleh karena itu, dibuatlah suatu cara untuk menghitung solusi sebuah model yang akan mendekati pada solusi eksaknya, dan kemudian metode ini disebut dengan metode numerik. Munculnya metode numerik tidak lain

dilatarbelakangi oleh keterbatasan manusia seperti disebutkan dalam surat an-Nahl ayat 18 tersebut.

Salah satu model hasil penerapan matematika dalam ilmu ekologi yang cukup terkenal ialah model *predator-prey*. Model *predator-prey* diperkenalkan oleh Alfred J. Lotka pada tahun 1925 dan Vito Volterra pada tahun 1926, yang memformulasikan model matematika tersebut dalam sistem persamaan diferensial (Boyce dan DiPrima, 2012). Model ini menggambarkan interaksi kehidupan suatu populasi pemangsa dan mangsanya. Dalam model ini terdapat ketidaklinieran dalam persamaannya, sehingga akan sulit untuk mencari solusi eksaknya. Oleh karena itu dalam penelitian ini dijelaskan sebuah metode untuk mencari solusi dari model *predator-prey* secara numerik.

Dalam penelitian sebelumnya telah banyak dijumpai tentang solusi numerik dari model *predator-prey* dengan menggunakan bermacam-macam metode. Metode-metode yang pernah dipakai untuk menyelesaikan model *predator-prey* ialah metode Runge-Kutta-Fehlberg, metode Dekomposisi Adomian Laplace (Paul dkk, 2014), Runge-Kutta orde 4, dan Runge-Kutta Gill (Putri, 2013). Penelitian ini menjelaskan tentang bagaimana mencari solusi numerik model *predator-prey* dengan menggunakan metode jaringan fungsi radial basis.

Jaringan fungsi radial basis merupakan pemetaan dari suatu vektor *input* dengan dimensi- p ke vektor *output* yang berdimensi 1, secara aljabar disimbolkan dengan $f: R^p \rightarrow R^1$. Fungsi f terdiri dari himpunan koefisien bobot $\{w_i\}_{i=1}^n$ dan himpunan fungsi-fungsi radial basis $\{\phi(x_k, c_i)\}_{i=1}^n$ dengan $\{x_k\}_{k=1}^m$ dengan $n \leq m$. Secara umum, fungsi radial basis yang dipakai dalam penelitian ini dapat ditulis

dengan $\phi(x_k, c_i) = \sqrt{(x_k - c_i)^2 + \alpha^2}$, $\alpha > 0$, x_i adalah vektor *input*, dan c_i adalah titik *center* dari x ke- k (Mai-Duy dan Tran-Cong, 2003).

Kelebihan metode jaringan fungsi radial basis dibandingkan dengan metode numerik lainnya adalah setiap fungsi dan turunannya dapat didekati langsung dengan fungsi radial basis. Metode jaringan fungsi radial basis dalam mendekati sebuah fungsi dibagi menjadi metode langsung dan metode tidak langsung. Metode langsung ialah mendekati suatu fungsi dengan langsung menggunakan fungsi radial basisnya, sedangkan metode tidak langsung ialah mendekati sebuah fungsi dengan mendekati fungsi turunannya terlebih dahulu dengan menggunakan fungsi radial basis, dan kemudian fungsi radial basisnya diintegrasikan (Mai-Duy dan Tran-Cong, 2003). Metode yang digunakan dalam penelitian ini adalah metode langsung.

Metode jaringan fungsi radial basis sudah banyak digunakan sebagai solusi numerik dari persamaan diferensial baik biasa maupun parsial. Dalam penelitian yang dilakukan oleh Mufidah dan Jamhuri (2015) telah ditunjukkan bahwa metode jaringan fungsi radial basis sangat efektif dalam menyelesaikan persamaan Poisson pada koordinat polar. Akan tetapi pada penelitian-penelitian terdahulu model yang diselesaikan sebagian besar merupakan model-model linier, sehingga koefisien-koefisien bobot $\{w_i\}_{i=1}^n$ dapat langsung dicari dengan hanya satu langkah saja, dan masih sedikit penelitian yang menyelesaikan model nonlinier dengan menggunakan jaringan fungsi radial basis. Jamhuri (2011) telah menyelesaikan persamaan Riccati yang merupakan persamaan diferensial biasa nonlinier dengan metode jaringan fungsi radial basis. Dalam penelitiannya, pencarian nilai bobot $\{w_i\}_{i=1}^n$ dilakukan dengan metode-metode numerik tanpa pelinieran terlebih dahulu pada modelnya, sehingga jika domain fungsinya besar akan diperlukan memori yang besar pula

dalam komputasinya. Penelitian ini menjelaskan penyelesaian numerik model nonlinier yaitu model *predator-prey* yang merupakan sistem persamaan diferensial orde satu nonlinier dengan melinierkan terlebih dahulu modelnya sehingga komputasinya akan memerlukan memori yang lebih kecil dari penelitian sebelumnya.

Penelitian ini diharapkan bisa memperoleh solusi numerik dari persamaan *predator-prey* yang lebih akurat dan komputasi yang lebih cepat daripada penelitian yang telah dilakukan sebelumnya. Selain itu penelitian ini adalah sebagai tambahan referensi dalam menyelesaikan model-model nonlinier dengan menggunakan jaringan fungsi radial basis. Penelitian ini dilakukan juga sebagai cara untuk menutupi keterbatasan manusia dalam hal menghitung. Oleh karena itu, penulis melakukan penelitian ini dengan judul “Solusi Numerik Model *Predator-Prey* Menggunakan Metode Jaringan Fungsi Radial Basis”.

1.2 Rumusan Masalah

Berdasarkan latar belakang di atas, maka rumusan masalah dalam penelitian ini adalah bagaimana solusi numerik model *predator-prey* menggunakan metode jaringan fungsi radial basis?

1.3 Tujuan Penelitian

Tujuan penelitian ini adalah untuk mengetahui solusi numerik model *predator-prey* menggunakan jaringan fungsi radial basis.

1.4 Batasan Masalah

Batasan masalah dalam penelitian ini adalah sebagai berikut:

1. Model *predator-prey* yang diselesaikan adalah model *predator-prey* dua populasi yaitu satu populasi *prey* dan satu populasi *predator*nya.
2. Fungsi radial basis yang digunakan adalah jenis fungsi *multiquadrics*.
3. Metode yang digunakan adalah metode langsung yaitu dengan menurunkan fungsi radial basis terhadap variabel bebasnya.

1.5 Manfaat Penelitian

Penelitian ini diharapkan memberikan metode alternatif dalam menyelesaikan model *predator-prey*.

1.6 Metode Penelitian

Metode yang digunakan dalam menyelesaikan model *predator-prey* secara numerik adalah menggunakan pendekatan studi literatur atau *library research*. Adapun langkah-langkah dalam penelitian ini adalah sebagai berikut:

1. Mengambil contoh model *predator-prey* dengan parameter-parameter umum seperti berikut:

$$\frac{dN(t)}{dt} = aN(t) - \alpha N(t)P(t)$$

$$\frac{dP(t)}{dt} = -bP(t) + \beta N(t)P(t)$$

2. Menambahkan subskrip pada variabel-variabel terikat model *predator-prey*.
3. Melinierkan suku nonlinier dari model *predator-prey* menggunakan deret Taylor.

4. Mendiskritisasi model *predator-prey* yang sudah dilinierkan menggunakan jaringan fungsi radial basis serta nilai awalnya.
5. Memasukkan nilai-nilai *input* (t_k) pada model *predator-prey* dalam bentuk persamaan jaringan fungsi radial basis yang telah diperoleh dan mengubahnya ke bentuk persamaan matriks.
6. Menghitung nilai bobot w_i dan v_i dengan cara iterasi.
7. Menghitung solusi model *predator-prey* dengan mengalikan nilai bobot w_i dan v_i yang telah diperoleh dan fungsi *multiquadrics* yang tidak diturunkan.
8. Melakukan simulasi dan menggambar grafik.
9. Menganalisis galat dengan mensubstitusikan solusi yang telah diperoleh beserta turunannya pada model yang disimulasikan.
10. Memberikan kesimpulan atas hasil penelitian yang telah diperoleh serta saran untuk penelitian selanjutnya.

1.7 Sistematika Penulisan

Penulisan penelitian ini menggunakan sistematika penulisan yang terdiri dari empat bab, yang masing-masing bab dibagi dalam subbab dengan sistematika penulisan sebagai berikut:

Bab I Pendahuluan

Bab ini meliputi beberapa subbab yaitu latar belakang, rumusan masalah, tujuan penelitian, batasan masalah, manfaat penelitian, metode penelitian, dan sistematika penulisan.

Bab II Kajian Pustaka

Bab ini berisi tentang ulasan-ulasan yang berkaitan dengan penelitian, baik dari penelitian terdahulu, berbagai literatur (buku), serta dari jurnal ilmiah. Di antaranya ialah *predator-prey*, jaringan fungsi radial basis, metode langsung jaringan fungsi radial basis, analisis galat, dan kajian agama tentang aproksimasi numerik dalam al-Quran.

Bab III Pembahasan

Bab ini menjelaskan langkah-langkah menyelesaikan model *predator-prey* dengan menggunakan jaringan fungsi radial basis beserta simulasi dan analisis galatnya.

Bab IV Penutup

Bab ini berisi kesimpulan yang diperoleh dari hasil penelitian yang dilakukan, dan saran yang disampaikan penulis untuk penelitian selanjutnya.

BAB II

KAJIAN PUSTAKA

2.1 Jaringan Fungsi Radial Basis

Jaringan fungsi radial basis diperkenalkan pertama kali oleh D.S. Broomhead dan David Lowe dalam makalahnya yang berjudul *Radial Basis Functions, Multi-Variable Functional Interpolation and Adaptive Networks* pada tahun 1988. Metode ini merupakan salah satu metode jaringan syaraf tiruan yang menggunakan fungsi aktivasi berupa fungsi radial basis. Jaringan syaraf tiruan itu sendiri dikembangkan pertama kali oleh seorang ahli neurofisiologi Waren McCulloch dan ahli logika Walter Pitts pada tahun 1943 sebagai algoritma pemrosesan informasi yang terinspirasi dari sistem kerja jaringan syaraf biologis manusia (Hajek, 2005).

Jaringan fungsi radial basis berkembang beberapa tahun terakhir dan banyak digunakan dalam penyelesaian masalah pendekatan fungsi. Struktur dari jaringan fungsi radial basis ini terdiri dari 3 bagian, yaitu lapisan *input*, lapisan tersembunyi, dan lapisan *output*. Tiap-tiap unit pada bagian lapisan tersembunyi merupakan representasi dari fungsi aktivasi yaitu fungsi radial basis (Setiawan, 2002).

Diberikan sebuah himpunan yang anggotanya berisi pasangan variabel bebas \mathbf{x} (sebuah vektor \mathbf{x}) dan variabel terikat y . Himpunan tersebut dilambangkan dengan $\{\mathbf{x}_k, y_k\}_{k=1}^m$ dengan m adalah banyak *input*, dan $\mathbf{x} = [x^{(1)}, x^{(2)}, \dots, x^{(p)}]^T$ dengan p adalah jumlah dimensi dan pangkat T adalah operasi transpose. Jaringan fungsi radial basis adalah bentuk fungsi aproksimasi \hat{y} dari variabel terikat y dan juga merupakan bentuk aproksimasi fungsi turunannya. Fungsi aproksimasinya

kemudian dapat ditulis dengan $y(\mathbf{x}_k) \approx \hat{y}(\mathbf{x}_k) = \sum_{i=1}^n w_i g_i$ (Mai-Duy dan Tran-Cong, 2003).

Fungsi $f: R^p \rightarrow R^1$ terdiri dari himpunan bobot $\{w_i\}_{i=1}^n$, dan himpunan fungsi radial basis $\{g_i\}_{i=1}^n$ dan $n \leq m$. Ada bermacam-macam jenis fungsi radial basis yang dapat digunakan sebagai fungsi aktivasi dan secara umum dapat ditulis dalam bentuk matematis $g_i(\mathbf{x}) = \phi(\|\mathbf{x} - c_i\|)$, yang mana $\|\cdot\|$ adalah lambang dari norma Euclid dan $\{c_i\}_{i=1}^n$ adalah himpunan dari *center* yang bisa dipilih dari sekitar data *input* (Mai-Duy dan Tran-Cong, 2003).

Terdapat beberapa jenis fungsi radial basis yang digunakan sebagai fungsi aktivasi dalam jaringan fungsi radial basis. Fungsi radial basis yang paling umum digunakan di antaranya adalah fungsi radial basis *multiquadrics*, *invers multiquadrics*, dan *gaussian*. Fungsi radial basis *multiquadrics* merupakan jenis fungsi radial basis yang memiliki keakuratan lebih baik dari jenis fungsi yang lain. Fungsi *multiquadrics* dikembangkan pertama kali oleh Roland Hardy pada tahun 1971 dalam karya penelitiannya yang berjudul *Multiquadrics Equations of Topography and Other Surfaces*. Fungsi *multiquadrics* ini kemudian dikembangkan oleh matematikawan Richard Franke pada tahun 1979 dan selanjutnya mengalami perkembangan yang signifikan dengan digunakannya fungsi *multiquadrics* dalam penyelesaian persamaan diferensial, melalui penelitian yang dilakukan oleh Edward Kansa pada tahun 1990 (Sarra dan Kansa, 2009).

Bentuk umum fungsi *multiquadrics* dalam mengaproksimasi fungsi yaitu:

1. Untuk fungsi 1 variabel

$$\phi(x, c_i) = \sqrt{(x - c_i)^2 + \alpha^2}, \quad (2.1)$$

untuk $\alpha > 0$

2. Untuk fungsi 2 variabel

$$\phi(x^{(1)}, x^{(2)}, c_i, d_i) = \sqrt{(x^{(1)} - c_i)^2 + (x^{(2)} - d_i)^2 + \beta^2}, \quad (2.2)$$

untuk $\beta > 0$

3. Untuk p variabel

$$\phi(\mathbf{x}, \mathbf{c}_i) = \sqrt{(\mathbf{x} - \mathbf{c}_i)^2 + \alpha^2},$$

Diberikan sebuah fungsi 1 variabel $y(x)$. $\hat{y}(x)$ adalah aproksimasi fungsi tersebut menggunakan jenis fungsi *multiquadrics*, maka:

$$\hat{y}(x) = \sum_{i=1}^n w_i \sqrt{(x - c_i)^2 + \alpha^2} \quad (2.3)$$

Diberikan fungsi 2 variabel $y(x^{(1)}, x^{(2)})$. $\hat{y}(x^{(1)}, x^{(2)})$ adalah aproksimasi fungsi tersebut menggunakan jenis fungsi *multiquadrics*, maka

$$\hat{y}(x^{(1)}, x^{(2)}) = \sum_{i=1}^n w_i \sqrt{(x^{(1)} - c_i)^2 + (x^{(2)} - d_i)^2 + \beta^2} \quad (2.4)$$

Aproksimasi fungsi 3 variabel sampai p variabel, hanya akan mengubah fungsi radial basisnya saja, mengikuti fungsi yang diaproksimasi tersebut.

Aminataei dan Mazarei (2008) menyatakan bahwa pembahasan yang penting dalam jaringan fungsi radial basis adalah menghitung nilai bobot w_i yang belum diketahui. Langkah-langkah dalam mengaproksimasi suatu fungsi menggunakan jaringan fungsi radial basis setelah dipilih sebuah himpunan fungsi ϕ , adalah menghitung nilai-nilai bobot w_i . Langkah selanjutnya, jika telah diperoleh himpunan nilai-nilai bobot w_i maka jaringan fungsi radial basis dapat dibentuk. Jaringan fungsi radial basis dibentuk oleh himpunan nilai bobot w_i dan

himpunan fungsi radial basis ϕ . Solusi numerik atau hampiran fungsi tersebut dapat dihitung dengan:

$$\hat{y}(\mathbf{x}) = \sum_{j=1}^n w_j \phi(\mathbf{x}, \mathbf{c}_j) \quad (2.5)$$

Mai-Duy dan Tran-Cong (2003) menyatakan bahwa hubungan antara nilai bobot w , y , dan fungsi radial basis A adalah:

$$A\mathbf{w} = \mathbf{y} \quad (2.6)$$

$$(A^{-1}A)\mathbf{w} = A^{-1}\mathbf{y} \quad (2.7)$$

$$A = \begin{bmatrix} A_1(x_1, c_1) & A_2(x_1, c_2) & \cdots & A_n(x_1, c_n) \\ A_1(x_2, c_1) & A_2(x_2, c_2) & \cdots & A_n(x_2, c_n) \\ \vdots & \vdots & \ddots & \vdots \\ A_1(x_m, c_1) & A_2(x_m, c_2) & \cdots & A_n(x_m, c_n) \end{bmatrix} \quad (2.8)$$

$$\mathbf{w} = [w_1, w_2, \dots, w_n]^T \quad (2.9)$$

$$\mathbf{y} = [y_1, y_2, \dots, y_m]^T \quad (2.10)$$

A merupakan fungsi radial basis ϕ yang telah diturunkan ataupun diintegrasikan terhadap variabel bebasnya. Pada kasus tertentu, dapat terjadi $m = n$.

2.1.1 Turunan-turunan Fungsi dengan Jaringan Fungsi Radial Basis

Pada sebarang jaringan fungsi radial basis yang mana fungsi basisnya ditentukan, maka turunan-turunan dari fungsinya dapat dihitung dengan jaringan yang terdiri dari kombinasi dari fungsi-fungsi yang dapat ditentukan pula (turunan dari fungsi-fungsi radial basisnya). Turunan dari aproksimasi fungsi satu variabel pada persamaan (2.3) dapat dihitung sebagai berikut:

$$\frac{d^r \hat{y}}{dx^r} = \hat{y}^{(r)}(x) = \sum_{i=1}^n w_i \frac{d^r \phi(x, c_i)}{dx^r}$$

Turunan parsial dari aproksimasi fungsi p variabel dapat dihitung sebagai berikut:

$$\frac{\partial^s \hat{y}}{\partial x_j \cdots \partial x_l} = \hat{y}_{j \cdots l}(\mathbf{x}) = \sum_{i=1}^n w_i \frac{\partial^s g_i}{\partial x_j \cdots \partial x_l}$$

dengan $\frac{\partial^s g_i}{\partial x_j \cdots \partial x_l}$ adalah fungsi basis yang sesuai dengan turunan fungsi $\hat{y}_{j \cdots l}(\mathbf{x})$, yang mana dapat diperoleh dengan menurunkan fungsi basis $g_i(\mathbf{x})$ dan fungsi $g_i(\mathbf{x})$ dapat terus-menerus diturunkan.

2.1.2 Fungsi *Multiquadrics* dan Penentuan *Shape-Parameter* α^2

Salah satu fungsi radial basis yang paling sering dipakai sebagai fungsi aktivasi ialah fungsi *multiquadrics* (MQ) yang diperkenalkan pertama kali oleh Hardy. Bentuk umum fungsi *multiquadrics* adalah

$$\phi(x, c_i) = ((x - c_i)^2 + \alpha^2)^{\frac{1}{2}}, \quad i = 1, 2, \dots, n \quad (2.11)$$

dengan α^2 adalah *input shape-parameter* yang tidak terikat dengan indeks i . Dalam bentuk ini α^2 disebut dengan *constant multiquadrics* (CMQ). Pemilihan α^2 ini sangat mempengaruhi akurasi dari fungsi *multiquadrics* sebagai fungsi aktivasi (Kansa dan Carlson, 1992).

Masalahnya sekarang ialah menemukan metode untuk menentukan nilai yang optimal α^2 sebagai CMQ. Sudah banyak penelitian yang telah dilakukan untuk menentukan *shape-parameter* CMQ yang optimal. Dalam perkembangannya *shape-parameter* yang awalnya berbentuk konstan (CMQ) berubah menjadi *shape-parameter* yang terikat dengan i . Dalam bentuk ini α^2 disebut dengan *variable multiquadrics* (VMQ) sehingga persamaan (2.11) berubah menjadi

$$\phi(x, c_i) = ((x - c_i)^2 + \alpha_i^2)^{\frac{1}{2}}, \quad i = 1, 2, \dots, n \quad (2.12)$$

Beberapa penelitian dalam menentukan VMQ ini ialah penelitian oleh Kansa (1990) yang mengenalkan *exponential shape-parameter* yang mempunyai bentuk

$$\alpha_i = \left[\alpha_{min}^2 \left(\frac{\alpha_{max}^2}{\alpha_{min}^2} \right)^{\frac{i-1}{n-1}} \right]^{\frac{1}{2}} \quad i = 1, 2, \dots, n \quad (2.13)$$

dan *random shape-parameter* yang diperkenalkan oleh Sarra dan Sturgill (2009) yang mempunyai bentuk

$$\alpha_i = \alpha_{min} + (\alpha_{max} - \alpha_{min})rand(1, n)_i \quad i = 1, 2, \dots, n \quad (2.14)$$

rand adalah sebuah perintah di program Matlab yang membangkitkan sebanyak n -buah bilangan acak pada interval $[0, 1]$. Pada masalah satu dimensi $\alpha_{min} = \frac{1}{\sqrt{n}}$ dan $\alpha_{max} = \frac{3}{\sqrt{n}}$.

Dalam penelitian ini akan dipakai VMQ yang diperkenalkan oleh Xiang, dkk (2012) yang memperkenalkan *trigonometric shape-parameter* yang mempunyai bentuk

$$\alpha_i = \alpha_{min} + (\alpha_{max} - \alpha_{min})\sin(i) \quad i = 1, 2, \dots, n \quad (2.15)$$

Telah ditunjukkan dalam artikelnya bahwa *trigonometric shape-parameter* mempunyai akurasi yang jauh lebih baik daripada *exponential shape-parameter* maupun *random shape-parameter*.

2.2 Model Predator-Prey

Model *predator-prey* atau yang dikenal juga dengan model Lotka-Volterra merupakan model yang menggambarkan perubahan populasi mangsa dan pemangsa terhadap waktu akibat adanya interaksi antara kedua populasi. Model

interaksi ini diperkenalkan oleh seorang ahli biofisika Amerika yaitu Alferd J. Lotka (1880-1949) dan ahli matematika terkemuka Italia yaitu Vito Volterra (1860-1940). Keduanya mengembangkan kajian matematis ini secara terpisah, Lotka mengembangkannya pada tahun 1925 sedangkan Volterra pada tahun 1926 (Boyce dan DiPrima, 2012).

Menurut Dwaradi (2011), model *predator-prey* yang juga dikenal dengan model Lotka-Volterra secara umum dibentuk berdasarkan asumsi-asumsi sebagai berikut:

1. Dalam keadaan tanpa *predator*, lingkungan hidup populasi *prey* sangat ideal dan perkembangannya tidak terbatas.
2. Pertumbuhan *predator* juga ideal, kecuali terdapat kendala makanan.
3. Laju *prey* proporsional dengan laju pertemuan antara *prey* dan *predator*.
4. Laju kematian *predator* adalah konstan, tidak terpengaruh terhadap kepadatan dan umur *prey*.
5. Efisiensi *predator* tidak tergantung umur *prey* dan *predator*.
6. Efisiensi penggunaan *prey* sebagai makanan *predator* untuk bereproduksi adalah konstan dan tidak tergantung umur dan kepadatan *predator*.
7. Gerakan dan kontak *prey* dan *predator* bergantung secara acak. Setiap individu *prey* memiliki peluang yang sama untuk dimangsa.
8. Waktu yang digunakan *predator* untuk memangsa diabaikan.
9. Kepadatan *prey* tidak mempengaruhi peluang pemangsaan.
10. Kepadatan *predator* tidak mempengaruhi peluang *predator* untuk memangsa.
11. Keadaan lingkungan adalah homogen.

Menurut Redjeki (2009) pada model *predator-prey*, misalkan $N(t)$ menyatakan banyaknya populasi *prey* pada saat t dan $P(t)$ menyatakan banyaknya populasi *predator* pada saat t . Jika populasi *prey* dan *predator* tidak saling berinteraksi maka model pertumbuhan masing-masing yaitu:

$$\frac{dN(t)}{dt} = aN(t) \quad (2.16)$$

$$\frac{dP(t)}{dt} = -bP(t) \quad (2.17)$$

dengan a merupakan konstanta pertumbuhan dari populasi *prey* dan b konstanta kematian dari populasi *predator*. Jika populasi *predator* dan *prey* saling berinteraksi, maka populasi *prey* akan berkurang karena dimakan oleh populasi *predator*. Adanya interaksi antara populasi *prey* dan *predator* mengakibatkan berkurangnya populasi *prey*, sedangkan populasi *predator* akan bertambah akibat adanya interaksi tersebut, sehingga modelnya menjadi:

$$\frac{dN(t)}{dt} = aN(t) - \alpha N(t)P(t) \quad (2.18)$$

$$\frac{dP(t)}{dt} = -bP(t) + \beta N(t)P(t) \quad (2.19)$$

Menurut Finizio dan Ladas (1988), model *predator-prey* merupakan sistem persamaan nonlinier dan tidak ada cara yang diketahui untuk menyelesaikan secara eksplisit, meskipun demikian masih dimungkinkan dengan menggunakan teori kualitatif mengenai sistem semacam itu.

Menurut Dwaradi (2011), model Lotka-Volterra layak digunakan jika interaksi yang terjadi hanya intraspecies. Intraspecies dapat diartikan interaksi yang terjadi antara spesies satu dengan spesies yang lain. Model ini layak digunakan

dalam kehidupan nyata dengan tidak terbatasnya kapasitas *prey*. Jika model ini terdapat keterbatasan kapasitasnya, maka model ini tidak layak digunakan.

2.3 Analisis Galat

Solusi numerik adalah solusi yang diperoleh dengan hanya mendekati saja, bukan solusi yang sebenarnya. Oleh karena itu, solusi numerik perlu dianalisis untuk mengetahui seberapa dekat solusi itu pada solusi yang sebenarnya (eksaknya). Dalam menganalisis solusi numerik, akan ada selisih antara solusi numerik yang diperoleh dan solusi eksaknya yang disebut dengan galat (*error*). Galat inilah yang dapat dibuat sebagai ukuran kedekatan solusi numerik terhadap solusi eksaknya. Akan tetapi tidak semua masalah mempunyai solusi eksak, seperti halnya model *predator-prey*.

Model *predator-prey* adalah sistem persamaan diferensial biasa nonlinier yang tidak dapat diketahui solusi eksaknya. Oleh karena itu, dibutuhkan cara tertentu untuk mengukur kedekatan solusi numerik model *predator-prey* dengan solusi sebenarnya. Dalam penelitian ini galat dari solusi numerik akan dihitung dengan cara yang akan dijelaskan berikut.

Seperti yang telah diketahui pada subbab sebelumnya, jaringan fungsi radial basis bukan hanya dapat mendekati sebuah fungsi saja, akan tetapi jaringan fungsi radial basis juga dapat mendekati turunan fungsi juga dengan nilai bobot yang sudah diketahui. Dengan modal nilai bobot yang sudah diketahui, nilai $N(t)$, $P(t)$, $\frac{dN(t)}{dt}$, dan $\frac{dP(t)}{dt}$ pada persamaan (2.18) dan (2.19) dapat dihitung. Langkah-langkah analisis galat solusi numerik model *predator-prey* menggunakan metode jaringan fungsi radial basis adalah sebagai berikut.

Pertama, persamaan (2.19) dan (2.20) diubah menjadi seperti berikut:

$$\frac{dN(t)}{dt} - aN(t) + \alpha N(t)P(t) = 0 \quad (2.20)$$

$$\frac{dP(t)}{dt} + bP(t) - \beta N(t)P(t) = 0 \quad (2.21)$$

Kedua, nilai $N(t)$, $P(t)$, $\frac{dN(t)}{dt}$, dan $\frac{dP(t)}{dt}$ pada persamaan (2.20) dan (2.21) dihitung menggunakan jaringan fungsi radial basis. Ketiga, substitusikan nilai $N(t)$, $P(t)$, $\frac{dN(t)}{dt}$, dan $\frac{dP(t)}{dt}$ yang sudah diperoleh pada persamaan (2.20) dan (2.21). Keempat, hasil dari substitusi pada langkah ketiga dibandingkan dengan 0, semakin kecil selisih dengan 0, maka semakin akurat solusinya. Terakhir, ukuran kecil besarnya selisih yang diperoleh pada langkah keempat adalah dengan membandingkan selisih mutlak terbesar dengan 10% dari nilai awal model *predator-prey* terkecil. Langkah-langkah analisis galat ini akan dibahas dengan lebih jelas pada bagian pembahasan.

2.4 Konsep Estimasi dan Aproksimasi dalam Al-Quran

Keindahan dan keteraturan pola bilangan matematika telah lama ditunjukkan oleh Allah Swt. kepada manusia sejak diturunkannya al-Quran. Tanpa disadari, al-Quran yang telah ada sejak zaman Rasulullah Saw. ternyata di dalamnya menginformasikan tentang ilmu pengetahuan yang baru-baru ini berkembang. Konsep tentang ilmu pengetahuan itu telah ada dalam al-Quran sejak 1400 tahun yang lalu.

Keindahan pola bilangan matematika dalam al-Quran menunjukkan bahwa Allah Swt. sebenarnya telah menunjukkan tentang adanya konsep ilmu matematika, yaitu suatu ilmu hitung yang selanjutnya dapat digunakan untuk membantu manusia

memecahkan masalah yang ada dalam kehidupan sehari-hari. Sebagai manusia tentunya harus mempelajari terlebih dahulu ilmu tersebut untuk dapat mengaplikasikannya dalam kehidupan sehari-hari. Konsep matematika yang telah diinformasikan dalam al-Quran salah satunya ialah konsep estimasi dan juga aproksimasi atau pendekatan. Allah Swt. berfirman dalam al-Quran surat ash-Shaffaat ayat 147:

وَأَرْسَلْنَاهُ إِلَىٰ مِائَةِ أَلْفٍ أَوْ يَزِيدُونَ^{١٤٧}

Artinya: “Dan Kami utus dia kepada seratus ribu orang atau lebih”. (QS. ash-Shaffaat/37:147).

Terjemah dari ayat tersebut dapat diketahui bahwa nabi Yunus diutus oleh Allah Swt. kepada umatnya yang berjumlah 100.000 orang atau lebih. Dalam tafsirnya, Ibnu Katsir (1994) menjelaskan bahwa ada beberapa pendapat berbeda dari para ahli tafsir dalam menafsirkan lafaz *أَوْ يَزِيدُونَ*. Ibnu Abbas mengatakan dalam sebuah riwayat darinya, bahwa jumlah umat nabi Yunus lebih dari 100.000, dimana jumlahnya adalah 130.000 orang. Ada riwayat lain dari Ibnu Abbas yang menyebutkan bahwa jumlah kaum nabi Yunus berjumlah 133.000 sampai 139.000. Ada riwayat lain dari Ibnu Abbas juga, yang menyebutkan bahwa jumlahnya sekitar 143.000 sampai 149.000 orang. Jubair mengatakan bahwa jumlah mereka lebih dari 70.000 orang. Mak-hul mengatakan bahwa mereka berjumlah 110.000 orang (Ibnu Katsir, 1994). Perbedaan pendapat ini terjadi karena memang bahasa al-Quran tidak menyebutkan secara pasti jumlah dari umat nabi Yunus, akan tetapi al-Quran hanya menyebutkan dengan taksiran atau pendekatan saja, yaitu 100.000 atau lebih.

Dalam tafsir Al-Kasyaf disebutkan bahwa maksud dari 100.000 atau lebih adalah dalam pandangan orang yang melihat pada umat nabi Yunus. Artinya jika

ada seseorang yang melihat pada kumpulan umat nabi Yunus maka ia akan berkata “jumlahnya sekitar 100.000 atau lebih” (Azzamkhasyari, 1998). Orang yang mengatakan jumlah dari mereka 100.000 orang atau lebih tidaklah tahu pasti akan jumlah mereka, melainkan ia hanya melakukan penaksiran atau pendekatan dalam menghitungnya.

Al-Qurthubi (2006) juga menjelaskan tentang perbedaan para ahli tafsir dalam menafsirkan lafaz *أَوْ يَزِيدُونَ*. Menurut Al-Mubarrad nabi Yunus diutus kepada sekumpulan orang yang andaikan dilihat maka yang melihat akan berkata “jumlah mereka sekitar 100.000 lebih”, pendapat ini sama dengan pendapat dalam kitab tafsir Al-Kasyaf. Sedangkan menurut Al-Akhfasy dan Az-Zujaj menafsirkan dengan “100.00 atau lebih berdasarkan perkiraan saja”. Menurut Hasan dan Robi’ mengatakan jumlahnya sekitar 135.000 orang.

Sementara dalam kitab tafsir *Ruhul Ma’ani* disebutkan bahwa 100.000 adalah jumlah dari orang-orang *mukallaf* saja, dan jika dihitung dengan para pemudanya maka jumlahnya adalah 100.000 orang lebih. Ada sebuah hadits yang menyebutkan bahwa jumlah umat nabi Yunus adalah 150.000 orang (Al-Alusi, 1981) . Jika hadits ini sahih maka tidak akan ada perbedaan dalam hal jumlah umat nabi Yunus. Dari beberapa pendapat ulama tafsir dapat disimpulkan bahwa jumlah umat nabi Yunus tidak dapat dikatakan secara pasti jumlahnya, melainkan hanya dapat ditaksir yaitu sekitar 100.000 orang atau lebih, dan dari sinilah dapat diambil pengertian bahwa di dalam al-Quran juga mengandung konsep estimasi juga aproksimasi.

BAB III

PEMBAHASAN

Pada bab ini dijelaskan tentang langkah-langkah penyelesaian numerik model *predator-prey* menggunakan metode jaringan fungsi radial basis. Model *predator-prey* merupakan sistem persamaan diferensial biasa nonlinier orde satu. Berbeda dengan persamaan linier yang dapat diselesaikan langsung dengan metode jaringan fungsi radial basis, model ini diperlukan cara khusus untuk menyelesaikannya. Karena model ini nonlinier, maka perlu adanya pelinieran terhadap suku nonliniernya. Berikut langkah-langkah penyelesaian numerik model *predator-prey* menggunakan metode jaringan fungsi radial basis.

3.1. Melinierkan Model *Predator-Prey* dengan Deret Taylor

Pada subbab ini akan dijelaskan langkah-langkah untuk melinierkan model *predator-prey*. Diberikan model *predator-prey* serta nilai awalnya sebagai berikut:

$$\frac{dN(t)}{dt} = aN(t) - \alpha N(t)P(t) \quad (3.1)$$

$$\frac{dP(t)}{dt} = -bP(t) + \beta N(t)P(t) \quad (3.2)$$

$$N(t_1) = N_0 \text{ dan } P(t_1) = P_0 \quad (3.3)$$

Kemudian persamaan (3.1) dan (3.2) diubah menjadi sebagai berikut:

$$\frac{dN(t)}{dt} - aN(t) + \alpha N(t)P(t) = 0 \quad (3.4)$$

$$\frac{dP(t)}{dt} + bP(t) - \beta N(t)P(t) = 0 \quad (3.5)$$

Pada persamaan (3.4) dan (3.5) terdapat suku nonlinier yaitu $\alpha N(t)P(t)$ pada persamaan (3.4) dan $\beta N(t)P(t)$ pada persamaan (3.5). Variabel-variabel

terikatnya kemudian ditambah subskrip yang menunjukkan perulangan agar selanjutnya dapat dilinierkan, penambahan subskrip ini merujuk pada artikel yang berjudul *Multiquadric Radial Basis Function Approximation Methods for the Numerical Solution of Partial Differential Equations* (Sarra dan Kansa, 2009) sehingga persamaan (3.4) dan (3.5) menjadi sebagai berikut:

$$\frac{dN(t)^{(l+1)}}{dt} - aN(t)^{(l+1)} + \alpha N(t)^{(l+1)}P(t)^{(l+1)} = 0 \quad (3.6)$$

$$\frac{dP(t)^{(l+1)}}{dt} + bP(t)^{(l+1)} - \beta N(t)^{(l+1)}P(t)^{(l+1)} = 0 \quad (3.7)$$

dengan $l = 1, 2, \dots$ sampai nilai terpenuhi. Untuk mempermudah penulisan maka variabel bebas pada persamaan yaitu t tidak akan ditulis sehingga $N(t)$ akan ditulis N saja dan $P(t)$ akan ditulis sebagai P saja. Suku-suku nonlinier pada persamaan (3.6) dan (3.7) secara berturut-turut kemudian dilinierkan dengan deret Taylor sebagai berikut:

$$N^{(l+1)}P^{(l+1)} = N^{(l)}P^{(l+1)} + \Delta t \left(\frac{dN^{(l)}}{dt} \right) P^{(l+1)} \quad (3.8)$$

$$N^{(l+1)}P^{(l+1)} = P^{(l)}N^{(l+1)} + \Delta t \left(\frac{dP^{(l)}}{dt} \right) N^{(l+1)} \quad (3.9)$$

Perbedaan dalam menderetkan antara persamaan (3.8) dan (3.9) dimaksudkan agar dua persamaan dalam model *predator-prey* tetap mempunyai ikatan. Persamaan (3.8) dan (3.9) kemudian disubstitusikan pada persamaan (3.6) dan (3.7) menjadi sebagai berikut:

$$\frac{dN^{(l+1)}}{dt} - aN^{(l+1)} + \alpha N^{(l)}P^{(l+1)} + \Delta t \alpha \left(\frac{dN^{(l)}}{dt} \right) P^{(l+1)} = 0 \quad (3.10)$$

$$\frac{dP^{(l+1)}}{dt} + bP^{(l+1)} - \beta P^{(l)}N^{(l+1)} + \Delta t \beta \left(\frac{dP^{(l)}}{dt} \right) N^{(l+1)} = 0 \quad (3.11)$$

Persamaan (3.10) dan (3.11) adalah bentuk linier dari model *predator-prey* yang dilinierkan menggunakan deret Taylor. Perbedaan penderetan antara 2 persamaan dalam model *predator-prey* dimaksudkan agar pada tiap persamaan tetap mengandung dua variabel $N^{(l+1)}$ dan $P^{(l+1)}$ sehingga hubungan antara kedua persamaan tetap ada sebagai satu model.

3.2 Mengubah ke Bentuk Jaringan Fungsi Radial Basis

Setelah mendapatkan persamaan linier (3.10) dan (3.11), langkah selanjutnya adalah mengubah ke bentuk jaringan fungsi radial basis. Metode jaringan fungsi radial basis yang digunakan adalah metode langsung yaitu dengan cara menurunkan fungsi radial basis terhadap variabel bebasnya, sehingga diperoleh $N^{(l+1)}$, $N^{(l)}$, $\frac{dN^{(l+1)}}{dt}$, $\frac{dN^{(l)}}{dt}$, $P^{(l+1)}$, $P^{(l)}$, $\frac{dP^{(l+1)}}{dt}$, dan $\frac{dP^{(l)}}{dt}$ sebagai berikut:

$$N^{(l+1)} = \sum_{i=1}^n w_i^{(l+1)} \phi(t, c_i) \quad (3.12)$$

$$\frac{dN^{(l+1)}}{dt} = \sum_{i=1}^n w_i^{(l+1)} \phi_t(t, c_i) \quad (3.13)$$

$$P^{(l+1)} = \sum_{i=1}^n v_i^{(l+1)} \phi(t, c_i) \quad (3.14)$$

$$\frac{dP^{(l+1)}}{dt} = \sum_{i=1}^n v_i^{(l+1)} \phi_t(t, c_i) \quad (3.15)$$

$$N^{(l)} = \sum_{i=1}^n w_i^{(l)} \phi(t, c_i) \quad (3.16)$$

$$\frac{dN^{(l)}}{dt} = \sum_{i=1}^n w_i^{(l)} \phi_t(t, c_i) \quad (3.17)$$

$$P^{(l)} = \sum_{i=1}^n v_i^{(l)} \phi(t, c_i) \quad (3.18)$$

$$\frac{dP^{(l)}}{dt} = \sum_{i=1}^n v_i^{(l)} \phi_t(t, c_i) \quad (3.19)$$

Persamaan (3.12) sampai (3.19) kemudian disubstitusikan pada persamaan (3.10) dan (3.11) menjadi:

$$\begin{aligned} & \sum_{i=1}^n w_i^{(l+1)} \phi_t(t, c_i) - a \sum_{i=1}^n w_i^{(l+1)} \phi(t, c_i) \\ & + \alpha \sum_{i=1}^n v_i^{(l+1)} \phi(t, c_i) \sum_{i=1}^n w_i^{(l)} \phi(t, c_i) \end{aligned} \quad (3.20)$$

$$+ \Delta t \alpha \sum_{i=1}^n v_i^{(l+1)} \phi_t(t, c_i) \sum_{i=1}^n w_i^{(l)} \phi_t(t, c_i) = 0$$

$$\begin{aligned} & \sum_{i=1}^n v_i^{(l+1)} \phi_t(t, c_i) + b \sum_{i=1}^n v_i^{(l+1)} \phi(t, c_i) \\ & - \beta \sum_{i=1}^n w_i^{(l+1)} \phi(t, c_i) \sum_{i=1}^n v_i^{(l)} \phi(t, c_i) \end{aligned} \quad (3.21)$$

$$- \Delta t \beta \sum_{i=1}^n w_i^{(l+1)} \phi_t(t, c_i) \sum_{i=1}^n v_i^{(l)} \phi_t(t, c_i) = 0$$

Pada persamaan (3.20) dan (3.21), indeks $l = 1, 2, \dots$ digunakan untuk mencari nilai bobot w_i dan v_i yang optimal dengan cara perulangan. $N^{(1)}$ dan $P^{(1)}$ boleh diambil dari nilai acak (Sarra dan Kansa, 2009), akan tetapi pada penelitian ini $N^{(1)}$ dan $P^{(1)}$ ditentukan dengan sebuah fungsi yang berhubungan nilai awal.

Oleh karena nilai dari $N^{(l)}$ dan $P^{(l)}$ selalu diketahui maka tidak perlu diubah ke bentuk fungsi radial basis dan hanya akan ditulis sebagai $N^{(l)}$, $\left(\frac{dN^{(l)}}{dt}\right)$, $P^{(l)}$, dan $\left(\frac{dP^{(l)}}{dt}\right)$ saja. Persamaan (3.20) dan (3.21) kemudian disederhanakan menjadi:

$$\sum_{i=1}^n w_i^{(l+1)} (\phi_t(t, c_i) - a\phi(t, c_i)) + \sum_{i=1}^n v_i^{(l+1)} \left(\alpha N^{(l)} \phi(t, c_i) + \Delta t \alpha \left(\frac{dN^{(l)}}{dt} \right) \phi(t, c_i) \right) = 0 \quad (3.22)$$

$$\sum_{i=1}^n v_i^{(l+1)} (\phi_t(t, c_i) + b\phi(t, c_i)) - \sum_{i=1}^n w_i^{(l+1)} \left(P^{(l)} \beta \phi(t, c_i) + \Delta t \beta \left(\frac{dP^{(l)}}{dt} \right) \phi(t, c_i) \right) = 0 \quad (3.23)$$

Persamaan (3.22) dan (3.23) merupakan model *predator-prey* dalam bentuk persamaan jaringan fungsi radial basis. Persamaan tersebut dapat dibentuk jika terlebih dahulu dicari $\phi(t, c_i)$ dan $\phi_t(t, c_i)$. $\phi_t(t, c_i)$ merupakan turunan dari $\phi(t, c_i)$ terhadap t karena menggunakan metode langsung, di mana $\phi(t, c_i)$ adalah:

$$\begin{aligned} \phi(t, c_i) &= \sqrt{(t - c_i)^2 + \alpha_i^2} \\ \phi_t(t, c_i) &= \frac{\partial}{\partial t} \sqrt{(t - c_i)^2 + \alpha_i^2} \\ &= \frac{\partial}{\partial t} ((t - c_i)^2 + \alpha_i^2)^{\frac{1}{2}} \\ &= \frac{1}{2} ((t - c_i)^2 + \alpha_i^2)^{-\frac{1}{2}} \cdot (2t - 2c_i) \\ &= \frac{(2t - 2c_i)}{2} ((t - c_i)^2 + \alpha_i^2)^{-\frac{1}{2}} \\ &= \frac{t - c_i}{\sqrt{((t - c_i)^2 + \alpha_i^2)}} \end{aligned} \quad (3.24)$$

$$\phi_t(t, c_i) = \frac{t - c_i}{\sqrt{((t - c_i)^2 + \alpha_i^2)}} \quad (3.25)$$

dengan $\alpha_i = \alpha_{min} + (\alpha_{max} - \alpha_{min})\sin(i)$. Untuk satu dimensi $\alpha_{min} = \frac{1}{\sqrt{n}}$ dan $\alpha_{max} = \frac{3}{\sqrt{n}}$ (Xiang dkk, 2012). Persamaan (3.24) dan (3.25) disubstitusikan ke persamaan (3.22) dan (3.23) kemudian menjadi:

$$\begin{aligned} & \sum_{i=1}^n w_i^{(l+1)} \left(\frac{t - c_i}{\sqrt{((t - c_i)^2 + \alpha_i^2)}} - a \sqrt{(t - c_i)^2 + \alpha_i^2} \right) \\ & + \sum_{i=1}^n v_i^{(l+1)} \left(\alpha N^{(l)} \sqrt{(t - c_i)^2 + \alpha_i^2} \right. \\ & \left. + \Delta t \alpha \left(\frac{dN^{(l)}}{dt} \right) \sqrt{(t - c_i)^2 + \alpha_i^2} \right) = 0 \end{aligned} \quad (3.26)$$

$$\begin{aligned} & \sum_{i=1}^n v_i^{(l+1)} \left(\frac{t - c_j}{\sqrt{((t - c_i)^2 + \alpha_i^2)}} + b \sqrt{(t - c_i)^2 + \alpha_i^2} \right) \\ & - \sum_{i=1}^n w_i^{(l+1)} \left(\beta P^{(l)} \sqrt{(t - c_i)^2 + \alpha_i^2} \right. \\ & \left. + \Delta t \beta \left(\frac{dP^{(l)}}{dt} \right) \sqrt{(t - c_i)^2 + \alpha_i^2} \right) = 0 \end{aligned} \quad (3.27)$$

Untuk mempersingkat penulisan persamaan (3.26) dan (3.27) maka dibuat pemisalan sebagai berikut:

$$\frac{t - c_i}{\sqrt{((t - c_i)^2 + \alpha_i^2)}} - a \sqrt{(t - c_i)^2 + \alpha_i^2} = h(t, c_i)$$

$$\alpha N^{(l)} \sqrt{(t - c_i)^2 + \alpha_i^2} + \Delta t \alpha \left(\frac{dN^{(l)}}{dt} \right) \sqrt{(t - c_i)^2 + \alpha_i^2} = g(t, c_i)$$

$$\frac{t - c_i}{\sqrt{(t - c_i)^2 + \alpha_i^2}} + b \sqrt{(t - c_i)^2 + \alpha_i^2} = e(t, c_i)$$

$$\beta P^{(l)} \sqrt{(t - c_i)^2 + \alpha_i^2} + \Delta t \beta \left(\frac{dP^{(l)}}{dt} \right) \sqrt{(t - c_i)^2 + \alpha_i^2} = f(t, c_i)$$

Dengan demikian persamaan (3.26) dan (3.27) menjadi

$$\sum_{i=1}^n w_i^{(l+1)} h(t, c_i) + \sum_{i=1}^n v_i^{(l+1)} g(t, c_i) = 0 \quad (3.28)$$

$$\sum_{i=1}^n v_i^{(l+1)} e(t, c_i) - \sum_{i=1}^n w_i^{(l+1)} f(t, c_i) = 0 \quad (3.29)$$

Nilai awal (3.3) juga diubah ke bentuk jaringan fungsi radial basis menjadi

$$\sum_{i=1}^n w_i^{(l+1)} \phi(t_1, c_i) = P_0 \quad \text{dan} \quad \sum_{i=1}^n v_i^{(l+1)} \phi(t_1, c_i) = P_0 \quad (3.30)$$

3.3 Mengubah ke Bentuk Persamaan Matriks dan Memasukkan Nilai-nilai (t_k) yang Didiskritkan

Setelah mendapatkan persamaan *predator-prey* dalam bentuk jaringan fungsi radial basis (3.28) dan (3.29), selanjutnya ialah mengubah ke bentuk persamaan matriks dan memasukkan nilai-nilai *input* t yang didiskritkan pada persamaannya. Misalkan diberikan $t = \{t_1, t_2, \dots, t_m\}$, t sudah didiskritkan maka dengan memasukkan nilai-nilai t pada persamaan (3.28) dan (3.29) diperoleh sistem persamaan (3.31) dan (3.32) sebagai berikut:

$$\begin{aligned} \sum_{i=1}^n w_i^{(l+1)} h(t_1, c_i) + \sum_{i=1}^n v_i^{(l+1)} g(t_1, c_i) &= 0 \\ \sum_{i=1}^n w_i^{(l+1)} h(t_2, c_i) + \sum_{i=1}^n v_i^{(l+1)} g(t_2, c_i) &= 0 \\ &\vdots \\ \sum_{i=1}^n w_i^{(l+1)} h(t_m, c_i) + \sum_{i=1}^n v_i^{(l+1)} g(t_m, c_i) &= 0 \end{aligned} \quad (3.31)$$

$$\begin{aligned}
\sum_{i=1}^n v_i^{(l+1)} e(t_1, c_i) - \sum_{i=1}^n w_i^{(l+1)} f(t_1, c_i) &= 0 \\
\sum_{i=1}^n v_i^{(l+1)} e(t_2, c_i) - \sum_{i=1}^n w_i^{(l+1)} f(t_2, c_i) &= 0 \\
&\vdots \\
\sum_{i=1}^n v_i^{(l+1)} e(t_m, c_i) - \sum_{i=1}^n w_i^{(l+1)} f(t_m, c_i) &= 0
\end{aligned} \tag{3.32}$$

Setelah memasukkan nilai-nilai (t) dan mendapatkan sistem persamaan (3.31) dan (3.32), selanjutnya ialah menggabungkan kedua sistem persamaan menjadi sebuah sistem persamaan. Penggabungan dapat dilakukan karena variabel-variabel yang tidak diketahui antara persamaan (3.31) dan (3.32) sama, yaitu w_i dan v_i . Sistem persamaan (3.31) dan (3.32) setelah digabungkan menjadi sebagai berikut:

$$\begin{aligned}
\sum_{i=1}^n w_i^{(l+1)} h(t_1, c_i) + \sum_{i=1}^n v_i^{(l+1)} g(t_1, c_i) &= 0 \\
\sum_{i=1}^n w_i^{(l+1)} h(t_2, c_i) + \sum_{i=1}^n v_i^{(l+1)} g(t_2, c_i) &= 0 \\
&\vdots \\
\sum_{i=1}^n w_i^{(l+1)} h(t_m, c_i) + \sum_{i=1}^n v_i^{(l+1)} g(t_m, c_i) &= 0 \\
-\sum_{i=1}^n w_i^{(l+1)} f(t_1, c_i) + \sum_{i=1}^n v_i^{(l+1)} e(t_1, c_i) &= 0 \\
-\sum_{i=1}^n w_i^{(l+1)} f(t_2, c_i) + \sum_{i=1}^n v_i^{(l+1)} e(t_2, c_i) &= 0 \\
&\vdots \\
-\sum_{i=1}^n w_i^{(l+1)} f(t_m, c_i) + \sum_{i=1}^n v_i^{(l+1)} e(t_m, c_i) &= 0
\end{aligned} \tag{3.33}$$

Persamaan (3.33) kemudian diubah menjadi persamaan matriks sebagai berikut:

$$\begin{bmatrix}
h(t_1, c_1) & h(t_1, c_2) & \dots & h(t_1, c_n) & g(t_1, c_1) & g(t_1, c_2) & \dots & g(t_1, c_n) \\
h(t_2, c_1) & h(t_2, c_2) & \dots & h(t_2, c_n) & g(t_2, c_1) & g(t_2, c_2) & \dots & g(t_2, c_n) \\
\vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\
h(t_m, c_1) & h(t_m, c_2) & \dots & h(t_m, c_n) & g(t_m, c_1) & g(t_m, c_2) & \dots & g(t_m, c_n) \\
-f(t_1, c_1) & -f(t_1, c_2) & \dots & -f(t_1, c_n) & e(t_1, c_1) & e(t_1, c_2) & \dots & e(t_1, c_n) \\
-f(t_2, c_1) & -f(t_2, c_2) & \dots & -f(t_2, c_n) & e(t_2, c_1) & e(t_2, c_2) & \dots & e(t_2, c_n) \\
\vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\
-f(t_m, c_1) & -f(t_m, c_2) & \dots & -f(t_m, c_n) & e(t_m, c_1) & e(t_m, c_2) & \dots & e(t_m, c_n)
\end{bmatrix}
\begin{bmatrix}
w_1^{(l+1)} \\
w_2^{(l+1)} \\
\vdots \\
w_n^{(l+1)} \\
v_1^{(l+1)} \\
v_2^{(l+1)} \\
\vdots \\
v_n^{(l+1)}
\end{bmatrix}
=
\begin{bmatrix}
0 \\
0 \\
\vdots \\
0
\end{bmatrix} \tag{3.34}$$

Perlu diingat bahwa nilai untuk $h(t_1, c_i)$, $g(t_1, c_i)$, $e(t_1, c_i)$, dan $f(t_1, c_i)$ akan selalu tetap pada setiap perulangan oleh $l = 1, 2, \dots$ dan seterusnya, dikarenakan

fungsi-fungsinya mengandung nilai awal yang nilainya sudah diberikan sebelumnya. Oleh karena itu $h(t_1, c_i)$, $g(t_1, c_i)$, $e(t_1, c_i)$ dan $f(t_1, c_i)$ akan digantikan dengan nilai awal, sehingga persamaan matriks (3.34) menjadi sebagai berikut:

$$\begin{bmatrix} \phi(t_1, c_1) & \phi(t_1, c_2) & \dots & \phi(t_1, c_n) & 0 & 0 & \dots & 0 \\ h(t_2, c_1) & h(t_2, c_2) & \dots & h(t_2, c_n) & g(t_2, c_1) & g(t_2, c_2) & \dots & g(t_2, c_n) \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ h(t_m, c_1) & h(t_m, c_2) & \dots & h(t_m, c_n) & g(t_m, c_1) & g(t_m, c_2) & \dots & g(t_m, c_n) \\ 0 & 0 & \dots & 0 & \phi(t_1, c_1) & \phi(t_1, c_2) & \dots & \phi(t_1, c_n) \\ -f(t_2, c_1) & -f(t_2, c_2) & \dots & -f(t_2, c_n) & e(t_2, c_1) & e(t_2, c_2) & \dots & e(t_2, c_n) \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ -f(t_m, c_1) & -f(t_m, c_2) & \dots & -f(t_m, c_n) & e(t_m, c_1) & e(t_m, c_2) & \dots & e(t_m, c_n) \end{bmatrix} \begin{bmatrix} w_1^{(l+1)} \\ w_2^{(l+1)} \\ \vdots \\ w_n^{(l+1)} \\ v_1^{(l+1)} \\ v_2^{(l+1)} \\ \vdots \\ v_n^{(l+1)} \end{bmatrix} = \begin{bmatrix} N_0 \\ 0 \\ \vdots \\ 0 \\ P_0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad (3.35)$$

Persamaan matriks (3.35) yang sudah diperoleh, kemudian akan digunakan untuk mencari nilai-nilai w_i dan v_i dengan perulangan $l = 1, 2, \dots$ sampai memperoleh nilai bobot yang optimal.

3.4 Menghitung Nilai Bobot w_i dan v_i

Untuk mencari nilai bobot w_i dan v_i pada persamaan (3.35) dilakukan dengan menggunakan perintah *backslash* (`\`) pada Matlab yang digunakan untuk menyelesaikan sistem persamaan linier.

$$\begin{bmatrix} \phi(t_1, c_1) & \phi(t_1, c_2) & \dots & \phi(t_1, c_n) & 0 & 0 & \dots & 0 \\ h(t_2, c_1) & h(t_2, c_2) & \dots & h(t_2, c_n) & g(t_2, c_1) & g(t_2, c_2) & \dots & g(t_2, c_n) \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ h(t_m, c_1) & h(t_m, c_2) & \dots & h(t_m, c_n) & g(t_m, c_1) & g(t_m, c_2) & \dots & g(t_m, c_n) \\ 0 & 0 & \dots & 0 & \phi(t_1, c_1) & \phi(t_1, c_2) & \dots & \phi(t_1, c_n) \\ -f(t_2, c_1) & -f(t_2, c_2) & \dots & -f(t_2, c_n) & e(t_2, c_1) & e(t_2, c_2) & \dots & e(t_2, c_n) \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ -f(t_m, c_1) & -f(t_m, c_2) & \dots & -f(t_m, c_n) & e(t_m, c_1) & e(t_m, c_2) & \dots & e(t_m, c_n) \end{bmatrix} = \mathbf{A}$$

$$[N_0, 0, \dots, 0, P_0, 0, \dots, 0]^T = \mathbf{F}$$

$$[w_1^{(l+1)}, w_2^{(l+1)}, \dots, w_n^{(l+1)}, v_1^{(l+1)}, v_2^{(l+1)}, \dots, v_n^{(l+1)}]^T = \mathbf{W}^{(l+1)}$$

maka nilai \mathbf{W} dapat dihitung dengan cara sebagai berikut:

$$\mathbf{W}^{(l+1)} = \mathbf{A} \setminus \mathbf{F}$$

Nilai-nilai bobot persamaan linier dapat dicari dengan hanya satu kali perintah *backslash*, akan tetapi untuk persamaan nonlinier pencarian nilai-nilai bobotnya perlu dilakukan perulangan hingga mendapat nilai bobot yang akurat. Pencarian nilai bobotnya dimulai dengan mencari nilai bobot $w_i^{(2)}$ dan $v_i^{(2)}$ dengan modal $N^{(1)}$ dan $P^{(1)}$ yang sudah diketahui yang dapat diperoleh dengan menggunakan perintah *rand*() pada Matlab. Akan tetapi, jika menggunakan perintah *rand*() dalam menentukan $N^{(1)}$ dan $P^{(1)}$ kekonvergenan dalam mencari nilai bobot pada setiap simulasi akan selalu berubah, hal ini disebabkan karena setiap kali pemakaian perintah *rand*() di Matlab akan menghasilkan bilangan yang selalu berbeda. Oleh karena itu nilai $N^{(1)}$ dan $P^{(1)}$ akan ditentukan sebagai sebuah fungsi yang berkaitan dengan nilai awal dari model yang akan diselesaikan agar kekonvergenannya selalu tetap pada setiap simulasi.

Nilai-nilai $N^{(1)}$ dan $P^{(1)}$ kemudian dimasukkan pada persamaan (3.26) dan (3.27) sehingga diperoleh nilai bobot $w_i^{(2)}$ dan $v_i^{(2)}$ melalui langkah-langkah yang telah dijelaskan. Kemudian nilai bobot baru digunakan untuk menghitung $N^{(2)}$ dan $P^{(2)}$ yang kemudian dimasukkan pada persamaan (3.26) dan (3.27) sehingga diperoleh nilai bobot baru yaitu $w_i^{(3)}$ dan $v_i^{(3)}$ dan demikian seterusnya. Perulangan pencarian nilai bobot dihentikan ketika $\max|[N^{(l)}, P^{(l)}] - [N^{(l+1)}, P^{(l+1)}]| \leq \epsilon$, dengan ϵ adalah batas toleransi galat yang sudah ditentukan sebelumnya.

3.5 Menghitung Solusi Model *Predator-Prey*

Setelah mendapatkan koefisien bobot w_i dan v_i yang optimal melalui langkah-langkah pada subbab sebelumnya, selanjutnya adalah menghitung solusi

dari model *predator-prey* dengan cara mengalikan koefisien bobot dengan fungsi radial basis yang tidak diturunkan. Untuk populasi *prey* (N) solusinya diperoleh dengan cara mengalikan fungsi radial basis yang tanpa diturunkan dengan nilai-nilai bobotnya sebagai berikut:

$$\begin{bmatrix} \phi(t_1, c_1) & \phi(t_1, c_2) & \cdots & \phi(t_1, c_n) \\ \phi(t_2, c_1) & \phi(t_2, c_2) & \cdots & \phi(t_2, c_n) \\ \vdots & \vdots & & \vdots \\ \phi(t_m, c_1) & \phi(t_m, c_2) & \cdots & \phi(t_m, c_n) \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{bmatrix} = \begin{bmatrix} N_0 \\ N(t_2) \\ \vdots \\ N(t_m) \end{bmatrix} \quad (3.36)$$

Sementara untuk populasi *predator* (P) solusinya didapatkan dengan cara berikut:

$$\begin{bmatrix} \phi(t_1, c_1) & \phi(t_1, c_2) & \cdots & \phi(t_1, c_n) \\ \phi(t_2, c_1) & \phi(t_2, c_2) & \cdots & \phi(t_2, c_n) \\ \vdots & \vdots & & \vdots \\ \phi(t_m, c_1) & \phi(t_m, c_2) & \cdots & \phi(t_m, c_n) \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix} = \begin{bmatrix} P_0 \\ P(t_2) \\ \vdots \\ P(t_m) \end{bmatrix} \quad (3.37)$$

$N(t)$ dan $P(t)$ adalah solusi numerik dari model *predator-prey*.

3.6 Simulasi

Penjelasan mengenai langkah-langkah menyelesaikan model *predator-prey* menggunakan metode jaringan fungsi radial basis telah dibahas pada subbab-subbab sebelumnya. Simulasi penyelesaian numerik model *predator-prey* menggunakan metode jaringan fungsi radial basis serta analisis galatnya akan dibahas pada subbab ini.

Dalam penelitian ini diselesaikan 2 model *predator-prey*. Model pertama adalah model dengan parameter sembarang, dan model kedua adalah model yang parameternya diambil dari artikel yang berjudul '*Numerical Solution of Lotka Volterra Prey-Predator Model by using Runge–Kutta–Fehlberg Method and Laplace Adomian Decomposition Method*' (Paul dan Mondal, 2016). Model *predator-prey* pertama yang akan diselesaikan adalah sebagai berikut:

$$\begin{aligned}\frac{dN(t)}{dt} &= 0,9N(t) - 0,5N(t)P(t) \\ \frac{dP(t)}{dt} &= -0,75P(t) + 0,25N(t)P(t)\end{aligned}\quad 0 \leq t \leq 5 \quad (3.38)$$

Kondisi awal persamaan (3.38) adalah:

$$\begin{aligned}N(0) &= 2 \\ P(0) &= 1\end{aligned}\quad (3.39)$$

Model Persamaan (3.38) dilinierkan dengan menggunakan metode deret Taylor sebagaimana dijelaskan pada subbab sebelumnya sehingga persamaan (3.38) menjadi seperti berikut:

$$\frac{dN^{(l+1)}}{dt} - 0,9N^{(l+1)} + 0,5N^{(l)}P^{(l+1)} + 0,5\Delta t \left(\frac{dN^{(l)}}{dt} \right) P^{(l+1)} = 0 \quad (3.40)$$

$$\frac{dP^{(l+1)}}{dt} + 0,75P^{(l+1)} - 0,25P^{(l)}N^{(l+1)} - 0,25\Delta t \left(\frac{dP^{(l)}}{dt} \right) N^{(l+1)} = 0 \quad (3.41)$$

Langkah selanjutnya yaitu mengubah persamaan model *predator-prey* yang telah dilinierkan beserta nilai awalnya menjadi persamaan dalam bentuk jaringan fungsi radial basis sebagai berikut:

$$\begin{aligned}\sum_{i=1}^n w_i^{(l+1)} \left(\frac{t - c_i}{\sqrt{(t - c_i)^2 + \alpha_i^2}} - 0,9\sqrt{(t - c_i)^2 + \alpha_i^2} \right) \\ + \sum_{i=1}^n v_i^{(l+1)} \left(0,5N^{(l)}\sqrt{(t - c_i)^2 + \alpha_i^2} \right) \\ + 0,5\Delta t \left(\frac{dN^{(l)}}{dt} \right) \sqrt{(t - c_i)^2 + \alpha_i^2} = 0\end{aligned}\quad (3.42)$$

$$\begin{aligned}
& \sum_{i=1}^n v_i^{(l+1)} \left(\frac{t - c_j}{\sqrt{((t - c_i)^2 + \alpha_i^2)}} + 0,75 \sqrt{(t - c_i)^2 + \alpha_i^2} \right) \\
& - \sum_{i=1}^n w_i^{(l+1)} \left(0,25 P^{(l)} \sqrt{(t - c_i)^2 + \alpha_i^2} \right. \\
& \left. + 0,25 \Delta t \left(\frac{dP^{(l)}}{dt} \right) \sqrt{(t - c_i)^2 + \alpha_i^2} \right) = 0
\end{aligned} \tag{3.43}$$

Domain pada persamaan (3.39) kemudian dipartisi menjadi beberapa data diskrit dengan $\Delta t = 0,01$, kemudian didapatkan $t = \{0; 0,01; 0,02; 0,03; \dots; 5\}$ dan jumlah dari t adalah 501 ($n = 501$). Titik-titik *center* diambil $c = t$. Nilai Δt yang dipakai pada simulasi adalah 0,01, 0,02 dan 0,005. Pemilihan Δt dengan nilai tersebut dikarenakan pencarian solusi akan konvergen dengan nilai tersebut dan belum tentu konvergen dengan nilai yang lain.

Langkah selanjutnya yaitu memasukkan nilai t yang telah didiskritkan pada persamaan (3.44) dan (3.45) dan diperoleh sistem persamaan seperti di bawah ini:

$$\begin{aligned}
& \sum_{i=1}^{501} w_i^{(l+1)} h(t_1, c_i) + \sum_{i=1}^{501} v_i^{(l+1)} g(t_1, c_i) = 0 \\
& \sum_{i=1}^{501} w_i^{(l+1)} h(t_2, c_i) + \sum_{i=1}^{501} v_i^{(l+1)} g(t_2, c_i) = 0 \\
& \vdots \\
& \sum_{i=1}^{501} w_i^{(l+1)} h(t_{501}, c_i) + \sum_{i=1}^{501} v_i^{(l+1)} g(t_{501}, c_i) = 0
\end{aligned} \tag{3.44}$$

$$\begin{aligned}
& \sum_{i=1}^{501} v_i^{(l+1)} e(t_1, c_i) - \sum_{i=1}^{501} w_i^{(l+1)} f(t_1, c_i) = 0 \\
& \sum_{i=1}^{501} v_i^{(l+1)} e(t_2, c_i) - \sum_{i=1}^{501} w_i^{(l+1)} f(t_2, c_i) = 0 \\
& \vdots \\
& \sum_{i=1}^{501} v_i^{(l+1)} e(t_{501}, c_i) - \sum_{i=1}^{501} w_i^{(l+1)} f(t_{501}, c_i) = 0
\end{aligned} \tag{3.45}$$

dengan

$$\begin{aligned}
 h(t, c_i) &= \frac{t - c_i}{\sqrt{((t - c_i)^2 + \alpha_i^2)}} - 0,9 \sqrt{(t - c_i)^2 + \alpha_i^2} \\
 g(t, c_i) &= 0,5N^{(l)} \sqrt{(t - c_i)^2 + \alpha_i^2} + 0,5 \left(\frac{dN^{(l)}}{dt} \right) \Delta t \sqrt{(t - c_i)^2 + \alpha_i^2} \\
 e(t, c_i) &= \frac{t - c_i}{\sqrt{((t - c_i)^2 + \alpha_i^2)}} + 0,75 \sqrt{(t - c_i)^2 + \alpha_i^2} \\
 f(t, c_i) &= 0,25 \sqrt{(t - c_i)^2 + \alpha_i^2} + 0,25 \left(\frac{dP^{(l)}}{dt} \right) \Delta t \sqrt{(t - c_i)^2 + \alpha_i^2}
 \end{aligned}$$

$$i = 1, 2, \dots, 501 \text{ dan } \alpha_i = \frac{1}{\sqrt{501}} + \left(\frac{3}{\sqrt{501}} - \frac{1}{\sqrt{501}} \right) \sin(i)$$

Sistem persamaan (3.44) dan (3.45) untuk $l = 1$ kemudian diubah menjadi persamaan matriks menjadi sebagai berikut:

$$\begin{bmatrix}
 \phi(t_1, c_1) & \phi(t_1, c_2) & \dots & \phi(t_1, c_{501}) & 0 & 0 & \dots & 0 \\
 h(t_2, c_1) & h(t_2, c_2) & \dots & h(t_2, c_{501}) & g(t_2, c_1) & g(t_2, c_2) & \dots & g(t_2, c_{501}) \\
 \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\
 h(t_{501}, c_1) & h(t_{501}, c_2) & \dots & h(t_{501}, c_{501}) & g(t_{501}, c_1) & g(t_{501}, c_2) & \dots & g(t_{501}, c_{501}) \\
 0 & 0 & \dots & 0 & \phi(t_1, c_1) & \phi(t_1, c_2) & \dots & \phi(t_1, c_{501}) \\
 -f(t_2, c_1) & -f(t_2, c_2) & \dots & -f(t_2, c_{501}) & e(t_2, c_1) & e(t_2, c_2) & \dots & e(t_2, c_{501}) \\
 \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\
 -f(t_{501}, c_1) & -f(t_{501}, c_2) & \dots & -f(t_{501}, c_{501}) & e(t_{501}, c_1) & e(t_{501}, c_2) & \dots & e(t_{501}, c_{501})
 \end{bmatrix}
 \begin{bmatrix}
 w_1^{(2)} \\
 w_2^{(2)} \\
 \vdots \\
 w_{501}^{(2)} \\
 v_1^{(2)} \\
 v_2^{(2)} \\
 \vdots \\
 v_{501}^{(2)}
 \end{bmatrix}
 =
 \begin{bmatrix}
 2 \\
 0 \\
 \vdots \\
 0 \\
 1 \\
 0 \\
 \vdots \\
 0
 \end{bmatrix}
 \quad (3.46)$$

Setelah mendapatkan persamaan matriks (3.46), nilai-nilai t kemudian dimasukkan sehingga persamaan matriksnya menjadi sebagai berikut:

$$\begin{bmatrix}
 \phi(0,0) & \phi(0,0,01) & \dots & \phi(0,5) & 0 & 0 & \dots & 0 \\
 h(0,01,0) & h(0,01,0,01) & \dots & h(0,01,5) & g(0,01,0) & g(0,01,0,01) & \dots & g(0,01,5) \\
 \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\
 h(5,0) & h(5,0,01) & \dots & h(5,5) & g(5,0) & g(5,0,01) & \dots & g(5,5) \\
 0 & 0 & \dots & 0 & \phi(0,0) & \phi(0,0,01) & \dots & \phi(0,5) \\
 -f(0,01,0) & -f(0,01,0,01) & \dots & -f(0,01,5) & e(0,01,0) & e(0,01,0,01) & \dots & e(0,01,5) \\
 \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\
 -f(5,0) & -f(5,0,01) & \dots & -f(5,5) & e(5,0) & e(5,0,01) & \dots & e(5,5)
 \end{bmatrix}
 \begin{bmatrix}
 w_1^{(2)} \\
 w_2^{(2)} \\
 \vdots \\
 w_{501}^{(2)} \\
 v_1^{(2)} \\
 v_2^{(2)} \\
 \vdots \\
 v_{501}^{(2)}
 \end{bmatrix}
 =
 \begin{bmatrix}
 2 \\
 0 \\
 \vdots \\
 0 \\
 1 \\
 0 \\
 \vdots \\
 0
 \end{bmatrix}
 \quad (3.47)$$

Sebagaimana telah disebutkan pada subbab sebelumnya bahwa nilai-nilai solusi awal $N^{(1)}$ dan $P^{(1)}$ ditentukan dengan sebuah fungsi yang berhubungan dengan nilai awal, dan untuk model ini nilai-nilai solusi awal yang dipilih ialah $N(t) = \sin t + 2$ untuk model *predator* dan $P(t) = \sin t + 1$ untuk model *prey*. Pada simulasi yang dilakukan, setelah dihitung, persamaan matriks (3.47) menjadi sebagai berikut:

$$\begin{bmatrix} 0,1199 & 0,1263 & \dots & 5,0002 & 0 & 0 & \dots & 0 \\ -0,0251 & -0,1133 & \dots & -5,4911 & 0,1213 & 0,1266 & \dots & 5,0101 \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ -3,5016 & -3,4927 & \dots & -0,0399 & 2,6049 & 2,5997 & \dots & 0,0231 \\ 0 & 0 & \dots & 0 & 0,1199 & 0,1263 & \dots & 5,002 \\ -0,0306 & -0,0318 & \dots & -1,2575 & 0,1733 & 0,0944 & \dots & 2,7427 \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ -0,0521 & -0,0520 & \dots & -0,0005 & 4,7508 & 4,7434 & \dots & 0,0333 \end{bmatrix} \begin{bmatrix} w_1^{(2)} \\ w_2^{(2)} \\ \vdots \\ w_{501}^{(2)} \\ v_1^{(2)} \\ v_2^{(2)} \\ \vdots \\ v_{501}^{(2)} \end{bmatrix} = \begin{bmatrix} 2 \\ 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad (3.48)$$

Matriks pada persamaan (3.48) berukuran 1002×1002 sehingga elemen-elemennya tidak mungkin dimunculkan dalam tulisan. Walaupun demikian, setiap elemen pada matriks dapat diketahui nilainya dengan menggunakan program Matlab pada lampiran. Nilai bobot $w_i^{(2)}$ dan $v_i^{(2)}$ kemudian diperoleh dari persamaan matriks berikut:

$$\begin{bmatrix} w_1^{(2)} \\ w_2^{(2)} \\ \vdots \\ w_{501}^{(2)} \\ v_1^{(2)} \\ v_2^{(2)} \\ \vdots \\ v_{501}^{(2)} \end{bmatrix} = \begin{bmatrix} 0,1199 & 0,1263 & \dots & 5,0002 & 0 & 0 & \dots & 0 \\ -0,0251 & -0,1133 & \dots & -5,4911 & 0,1213 & 0,1266 & \dots & 5,0101 \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ -3,5016 & -3,4927 & \dots & -0,0399 & 2,6049 & 2,5997 & \dots & 0,0231 \\ 0 & 0 & \dots & 0 & 0,1199 & 0,1263 & \dots & 5,002 \\ -0,0306 & -0,0318 & \dots & -1,2575 & 0,1733 & 0,0944 & \dots & 2,7427 \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ -0,0521 & -0,0520 & \dots & -0,0005 & 4,7508 & 4,7434 & \dots & 0,0333 \end{bmatrix}^{-1} \begin{bmatrix} 2 \\ 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} = \begin{bmatrix} -376,7079 \\ 294,0093 \\ \vdots \\ -0,2911 \\ -101,276 \\ 127,7081 \\ \vdots \\ 0,0352 \end{bmatrix} \quad (3.49)$$

Nilai bobot $w_i^{(2)}$ dan $v_i^{(2)}$ yang diperoleh dari persamaan matriks (3.49) adalah sebagai berikut:

$$\begin{aligned} w^{(2)} &= \{-376,7079; 294,0093; \dots; -0,2911\} \\ v^{(2)} &= \{-101,276; 127,7091; \dots; 0,0352\} \end{aligned} \quad (3.50)$$

Solusi $N^{(2)}$ dan $P^{(2)}$ kemudian dapat dihitung dengan mengalikan fungsi basis dengan nilai bobot $v_i^{(2)}$ dan $w_i^{(2)}$ pada persamaan (3.50) yang telah diperoleh sebagai berikut:

$$\begin{bmatrix} N(t_1)^{(2)} \\ N(t_2)^{(2)} \\ \vdots \\ N(t_{501})^{(2)} \end{bmatrix} = \begin{bmatrix} \phi(t_1, c_1) & \phi(t_1, c_2) & \dots & \phi(t_1, c_{501}) \\ \phi(t_2, c_1) & \phi(t_2, c_2) & \dots & \phi(t_2, c_{501}) \\ \vdots & \vdots & & \vdots \\ \phi(t_{501}, c_1) & \phi(t_{501}, c_2) & \dots & \phi(t_{501}, c_{501}) \end{bmatrix} \begin{bmatrix} w_1^{(2)} \\ w_2^{(2)} \\ \vdots \\ w_{501}^{(2)} \end{bmatrix} =$$

$$\begin{bmatrix} 0,1199 & 0,1263 & \dots & 5,002 \\ 0,1203 & 0,1259 & \dots & 4,9902 \\ \vdots & \vdots & & \vdots \\ 5,0014 & 4,9916 & \dots & 0,0444 \end{bmatrix} \begin{bmatrix} -376,7079 \\ 294,0093 \\ \vdots \\ -0,2911 \end{bmatrix} = \begin{bmatrix} 2 \\ 2,0080 \\ \vdots \\ 25,3079 \end{bmatrix}$$

$$\begin{bmatrix} P(t_1)^{(2)} \\ P(t_2)^{(2)} \\ \vdots \\ P(t_{501})^{(2)} \end{bmatrix} = \begin{bmatrix} \phi(t_1, c_1) & \phi(t_1, c_2) & \dots & \phi(t_1, c_{501}) \\ \phi(t_2, c_1) & \phi(t_2, c_2) & \dots & \phi(t_2, c_{501}) \\ \vdots & \vdots & & \vdots \\ \phi(t_{501}, c_1) & \phi(t_{501}, c_2) & \dots & \phi(t_{501}, c_{501}) \end{bmatrix} \begin{bmatrix} v_1^{(2)} \\ v_2^{(2)} \\ \vdots \\ v_{501}^{(2)} \end{bmatrix} =$$

$$\begin{bmatrix} 0,1199 & 0,1263 & \dots & 5,002 \\ 0,1203 & 0,1259 & \dots & 4,9902 \\ \vdots & \vdots & & \vdots \\ 5,0014 & 4,9916 & \dots & 0,0444 \end{bmatrix} \begin{bmatrix} -101,2760 \\ 127,7081 \\ \vdots \\ 0,0352 \end{bmatrix} = \begin{bmatrix} 1 \\ 0,9976 \\ \vdots \\ 0,9499 \end{bmatrix}$$

Perulangan dilanjutkan untuk $l = 2$. Nilai-nilai $N^{(2)}$, $P^{(2)}$, $\frac{dN^{(2)}}{dt}$ dan $\frac{dP^{(2)}}{dt}$ sudah diketahui pada langkah sebelumnya saat $l = 1$. Nilai bobot $w_i^{(3)}$ dan $v_i^{(3)}$ dihitung dengan cara yang sama saat $l = 1$ sebagai berikut:

$$\begin{bmatrix} w_1^{(3)} \\ w_2^{(3)} \\ \vdots \\ w_{501}^{(3)} \\ v_1^{(3)} \\ v_2^{(3)} \\ \vdots \\ v_{501}^{(3)} \end{bmatrix} = \begin{bmatrix} 0,1199 & 0,1263 & \dots & 5,0002 & 0 & 0 & \dots & 0 \\ -0,0251 & -0,1133 & \dots & -5,4911 & 0,1211 & 0,1264 & \dots & 5,0062 \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ -3,5016 & -3,4927 & \dots & -0,0399 & 63,3993 & 63,2747 & \dots & 0,5613 \\ 0 & 0 & \dots & 0 & 0,1199 & 0,1263 & \dots & 5,002 \\ -0,0299 & -0,0314 & \dots & -1,2451 & 0,1733 & 0,0944 & \dots & 2,7427 \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ -1,1867 & -1,1867 & \dots & -0,0105 & 4,7508 & 4,7434 & \dots & 0,0333 \end{bmatrix}^{-1} \begin{bmatrix} 2 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} = \begin{bmatrix} 65,7446 \\ -139,8187 \\ \vdots \\ -0,1949 \\ -39,7926 \\ 58,7053 \\ \vdots \\ 0,0404 \end{bmatrix} \quad (3.51)$$

Nilai bobot $w_i^{(3)}$ dan $v_i^{(3)}$ yang diperoleh dari persamaan matriks (3.51) adalah sebagai berikut:

$$w^{(3)} = \{65,7446; -139,8187; \dots; -0,1949\}$$

$$v^{(3)} = \{-39,7962; 58,7053; \dots; 0,0404\}$$

Solusi $N^{(3)}$ dan $P^{(3)}$ kemudian dapat dihitung dengan mengalikan fungsi basis dengan nilai bobot $v_i^{(3)}$ dan $w_i^{(3)}$ yang telah diperoleh sebagai berikut:

$$\begin{bmatrix} N(t_1)^{(3)} \\ N(t_2)^{(3)} \\ \vdots \\ N(t_{501})^{(3)} \end{bmatrix} = \begin{bmatrix} \phi(t_1, c_1) & \phi(t_1, c_2) & \dots & \phi(t_1, c_{501}) \\ \phi(t_2, c_1) & \phi(t_2, c_2) & \dots & \phi(t_2, c_{501}) \\ \vdots & \vdots & & \vdots \\ \phi(t_{501}, c_1) & \phi(t_{501}, c_2) & \dots & \phi(t_{501}, c_{501}) \end{bmatrix} \begin{bmatrix} w_1^{(3)} \\ w_2^{(3)} \\ \vdots \\ w_{501}^{(3)} \end{bmatrix} =$$

$$\begin{bmatrix} 0,1199 & 0,1263 & \dots & 5,002 \\ 0,1203 & 0,1259 & \dots & 4,9902 \\ \vdots & \vdots & & \vdots \\ 5,0014 & 4,9916 & \dots & 0,0444 \end{bmatrix} \begin{bmatrix} 65,7446 \\ -139,8187 \\ \vdots \\ -0,1949 \end{bmatrix} = \begin{bmatrix} 2 \\ 2,0080 \\ \vdots \\ -9,5283 \end{bmatrix}$$

$$\begin{bmatrix} P(t_1)^{(2)} \\ P(t_2)^{(2)} \\ \vdots \\ P(t_{501})^{(2)} \end{bmatrix} = \begin{bmatrix} \phi(t_1, c_1) & \phi(t_1, c_2) & \dots & \phi(t_1, c_{501}) \\ \phi(t_2, c_1) & \phi(t_2, c_2) & \dots & \phi(t_2, c_{501}) \\ \vdots & \vdots & & \vdots \\ \phi(t_{501}, c_1) & \phi(t_{501}, c_2) & \dots & \phi(t_{501}, c_{501}) \end{bmatrix} \begin{bmatrix} v_1^{(2)} \\ v_2^{(2)} \\ \vdots \\ v_{501}^{(2)} \end{bmatrix} =$$

$$\begin{bmatrix} 0,1199 & 0,1263 & \dots & 5,002 \\ 0,1203 & 0,1259 & \dots & 4,9902 \\ \vdots & \vdots & & \vdots \\ 5,0014 & 4,9916 & \dots & 0,0444 \end{bmatrix} \begin{bmatrix} -39,7962 \\ 58,7053 \\ \vdots \\ 0,0404 \end{bmatrix} = \begin{bmatrix} 1 \\ 0,9975 \\ \vdots \\ -0,8648 \end{bmatrix}$$

Perulangan terus dilanjutkan sampai diperoleh selisih $N(t)^{(l+1)} - N(t)^l$ dan $P(t)^{l+1} - P(t)^l$ yang cukup kecil sehingga diperoleh nilai bobot yang optimal. Pada simulasi pertama ini perulangan dihentikan dan nilai bobot optimal ditemukan pada saat $l = 26$.

Setelah diperoleh nilai bobot yang optimal, solusi numerik model *predator-prey* (3.39) dengan kondisi awal (3.40) dapat dihitung dengan mengalikan fungsi basis dengan nilai bobotnya. Dari simulasi yang dilakukan diperoleh nilai bobot optimal sebagai berikut:

$$w = [-54,2741; 83,0544; -0,4345; \dots; 0,0425]^T$$

$$v = [8,4860; 10,1194; 0,4651; \dots; 0,0191]^T$$

Selanjutnya solusi dari model *predator-prey* (3.35) dapat dihitung sebagai berikut:

$$\begin{bmatrix} N_0 \\ N(t_2) \\ \vdots \\ N(t_{501}) \end{bmatrix} = \begin{bmatrix} \phi(t_1, c_1) & \phi(t_1, c_2) & \dots & \phi(t_1, c_{501}) \\ \phi(t_2, c_1) & \phi(t_2, c_2) & \dots & \phi(t_2, c_{501}) \\ \vdots & \vdots & & \vdots \\ \phi(t_{501}, c_1) & \phi(t_{501}, c_2) & \dots & \phi(t_{501}, c_{501}) \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_{501} \end{bmatrix} = \quad (3.52)$$

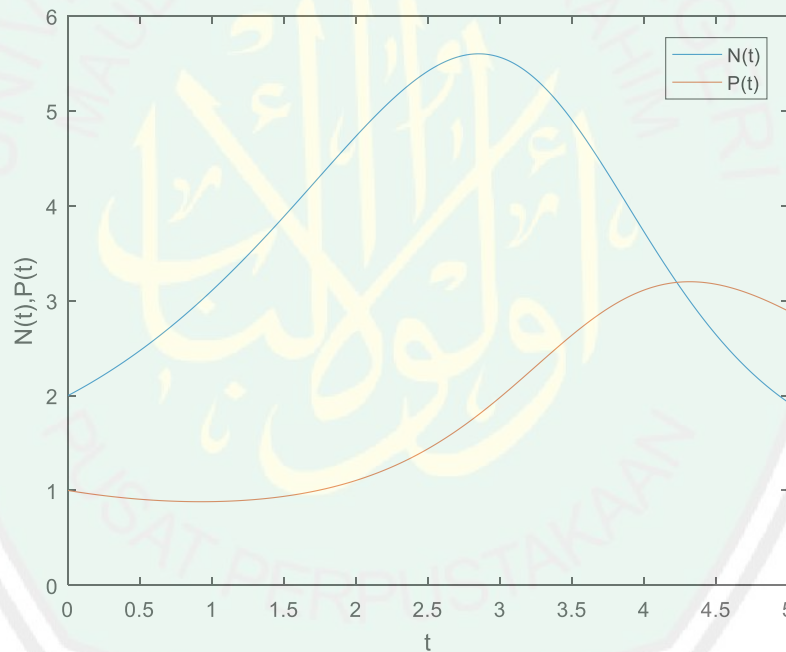
$$\begin{bmatrix} 0,1199 & 0,1263 & \dots & 5,002 \\ 0,1203 & 0,1259 & \dots & 4,9902 \\ \vdots & \vdots & & \vdots \\ 5,0014 & 4,9916 & \dots & 0,0444 \end{bmatrix} \begin{bmatrix} -54,2741 \\ 83,0544 \\ \vdots \\ 0,0425 \end{bmatrix} = \begin{bmatrix} 2 \\ 2,0080 \\ \vdots \\ 1,9261 \end{bmatrix}$$

Sedangkan untuk populasi *predator*, solusinya diperoleh dengan cara berikut:

$$\begin{bmatrix} P_0 \\ P(t_2) \\ \vdots \\ P(t_{501}) \end{bmatrix} = \begin{bmatrix} \phi(t_1, c_1) & \phi(t_1, c_2) & \cdots & \phi(t_1, c_{501}) \\ \phi(t_2, c_1) & \phi(t_2, c_2) & \cdots & \phi(t_2, c_{501}) \\ \vdots & \vdots & \ddots & \vdots \\ \phi(t_{501}, c_1) & \phi(t_{501}, c_2) & \cdots & \phi(t_{501}, c_{501}) \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_{501} \end{bmatrix} = \quad (3.53)$$

$$\begin{bmatrix} 0,1199 & 0,1263 & \cdots & 5,002 \\ 0,1203 & 0,1259 & \cdots & 4,9902 \\ \vdots & \vdots & \ddots & \vdots \\ 5,0014 & 4,9916 & \cdots & 0,0444 \end{bmatrix} \begin{bmatrix} 8,4860 \\ 10,1194 \\ \vdots \\ 0,0191 \end{bmatrix} = \begin{bmatrix} 1 \\ 0,9975 \\ \vdots \\ 2,8890 \end{bmatrix}$$

Hasil simulasi numerik persamaan (3.38) dengan kondisi batas (3.39) dalam bentuk grafik dengan menggunakan program Matlab adalah sebagai berikut:



Gambar 3.1 Solusi Numerik Model *Predator-Prey* (3.38) dengan $\Delta t = 0,01$

Model *predator-prey* merupakan sistem persamaan diferensial biasa nonlinier. Model ini menggambarkan interaksi antara populasi pemangsa dan populasi mangsanya dari waktu ke waktu. Dalam arti lain model ini adalah *time-invariant system* yaitu perubahan populasi mangsa dan pemangsanya akan berubah seiring dengan berubahnya waktu.

Gambar 3.1 mendeskripsikan bahwa ketika populasi *prey* sudah mulai tumbuh maka populasi *predator* juga akan tumbuh. Hal ini terjadi karena ketika populasi mangsa banyak maka persediaan makanan bagi *predator* akan melimpah. Akibatnya populasi mangsa akan tumbuh secara signifikan karena adanya interaksi pemangsaan.

Gambar 3.1 juga mendeskripsikan ketika populasi mangsa sudah mulai menurun karena dimangsa oleh *predator*, populasi *predator* kemudian juga akan menurun. Hal ini disebabkan karena semakin berkurangnya populasi mangsa maka persediaan makanan akan semakin sedikit sehingga *predator* akan mengalami kematian. Naik turunnya populasi *predator* dan *prey* ini akan selalu terjadi ketika waktu t semakin diperpanjang.

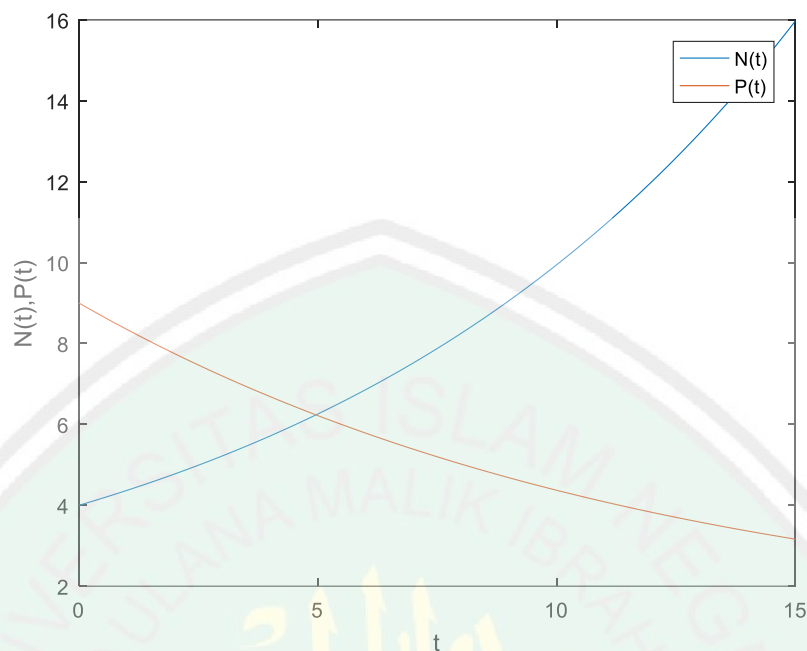
Model *predator-prey* kedua yang akan diselesaikan adalah sebagai berikut:

$$\begin{aligned} \frac{dN(t)}{dt} &= 0,1N(t) - 0,0014N(t)P(t) \\ \frac{dP(t)}{dt} &= -0,08P(t) + 0,0012N(t)P(t) \end{aligned} \quad 0 \leq t \leq 15 \quad (3.54)$$

dengan nilai awal

$$\begin{aligned} N(0) &= 4 \\ P(0) &= 9 \end{aligned} \quad (3.55)$$

Dengan cara yang sama dengan model pertama diperoleh solusi numerik dari model (3.54) dengan nilai awal (3.55) sebagai berikut:



Gambar 3.2 Solusi Numerik Model *Predator-Prey* (3.54) dengan $\Delta t = 0,01$

Selanjutnya untuk mengetahui galat dari solusi numerik yang diperoleh, akan dilakukan analisis galat yang akan diterangkan pada subbab tersendiri.

3.7 Analisis Galat

Untuk mengetahui seberapa dekat solusi yang diperoleh dengan menggunakan jaringan fungsi radial basis maka dilakukanlah analisis galat. Analisis galat dapat dilakukan dengan membandingkan selisih antara solusi numerik dan solusi eksaknya. Akan tetapi model *predator-prey* tidak mempunyai solusi eksak. Maka dilakukanlah langkah-langkah berikut dalam menganalisis galat.

Seperti yang sudah dijelaskan pada bab sebelumnya, jaringan fungsi radial basis selain dapat mendekati sebuah fungsi juga dapat mendekati turunan fungsinya ketika nilai-nilai bobot sudah diketahui. Sehingga pada simulasi subbab sebelumnya, nilai-nilai $N(t_k)$, $P(t_k)$, $\frac{dN(t_k)}{dt}$, dan $\frac{dP(t_k)}{dt}$ sudah dapat diketahui. Galat

dari solusi kemudian dicari dengan mesubstitusikan nilai-nilai dari $N(t_k)$, $P(t_k)$, $\frac{dN(t_k)}{dt}$, dan $\frac{dP(t_k)}{dt}$ pada model *predator-prey* (3.38) dan juga (3.54), galatnya adalah selisih mutlak dari hasil substitusinya dengan 0. Semakin kecil selisihnya maka semakin akurat solusinya. Besar kecilnya galat ditentukan dengan membandingkan nilai galat dengan 10% dari nilai awal. Galat dianggap cukup kecil jika

$$\max \varepsilon < \frac{T_0}{10}$$

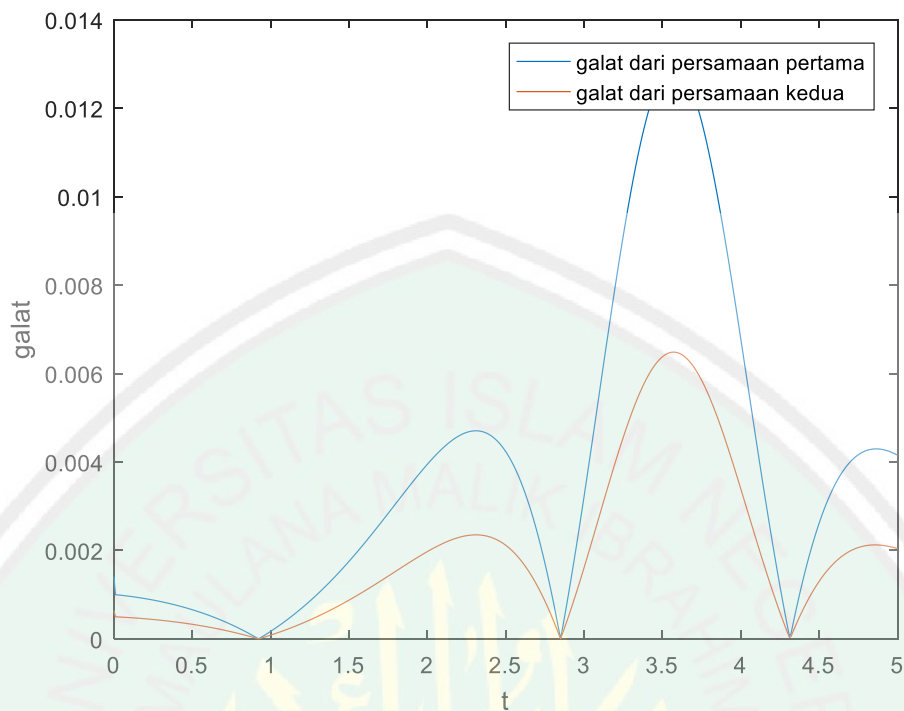
T_0 adalah nilai awal terkecil dari model *predator-prey* antara N_0 dan P_0 , ε adalah nilai galat yang dihitung dengan cara yang akan dijelaskan. $T_0 = P_0 = 1$ untuk model pertama dan $T_0 = P_0 = 4$ untuk model kedua.

Galat solusi numerik model *predator-prey* (3.38) menggunakan metode jaringan fungsi radial basis dihitung dari persamaan berikut:

$$\begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_{501} \end{bmatrix} = \begin{bmatrix} \frac{dN(t_1)}{dt} - 0,9N(t_1) + 0,5N(t_1)P(t_1) \\ \frac{dN(t_2)}{dt} - 0,9N(t_2) + 0,5N(t_2)P(t_2) \\ \vdots \\ \frac{dN(t_{501})}{dt} - 0,9N(t_{501}) + 0,5N(t_{501})P(t_{501}) \end{bmatrix} \quad (3.56)$$

$$\begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_{501} \end{bmatrix} = \begin{bmatrix} \frac{dP(t_1)}{dt} + 0,75P(t_1) - 0,25N(t_1)P(t_1) \\ \frac{dP(t_2)}{dt} + 0,75P(t_2) - 0,25N(t_2)P(t_2) \\ \vdots \\ \frac{dP(t_{501})}{dt} + 0,75P(t_{501}) - 0,25N(t_{501})P(t_{501}) \end{bmatrix} \quad (3.57)$$

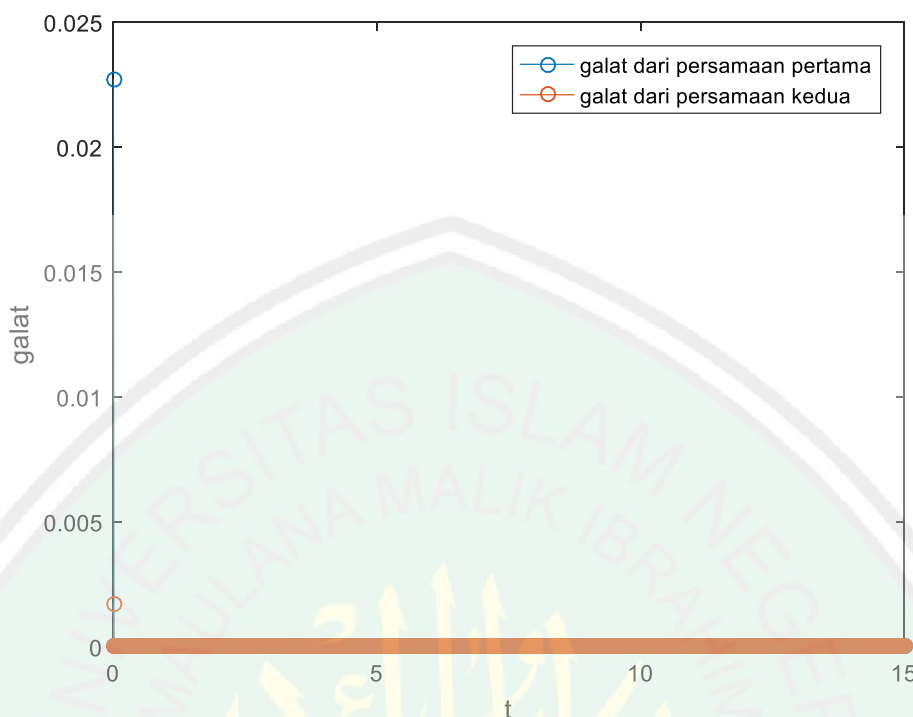
Hasil dari perhitungan galat dapat dilihat pada Gambar 3.3 berikut:



Gambar 3.3 Galat Solusi Numerik (ε) Model *Predator-Prey* (3.38) dengan $\Delta t = 0,01$

Gambar 3.3 menunjukkan galat mutlak dari solusi model *predator-prey* (3.38) dengan metode jaringan fungsi radial basis. Dari simulasi menggunakan program Matlab diperoleh galat mutlak maksimum dari persamaan (3.56) dan (3.57) adalah $\varepsilon = 0,013$. Galat ini sudah cukup kecil karena $0,013 < \frac{1}{10}$. Begitu seterusnya dalam mengukur besar kecilnya galat dalam penelitian ini.

Untuk model (3.54) dengan cara yang sama dalam menganalisis galat model pertama diperoleh galat sebagai berikut:



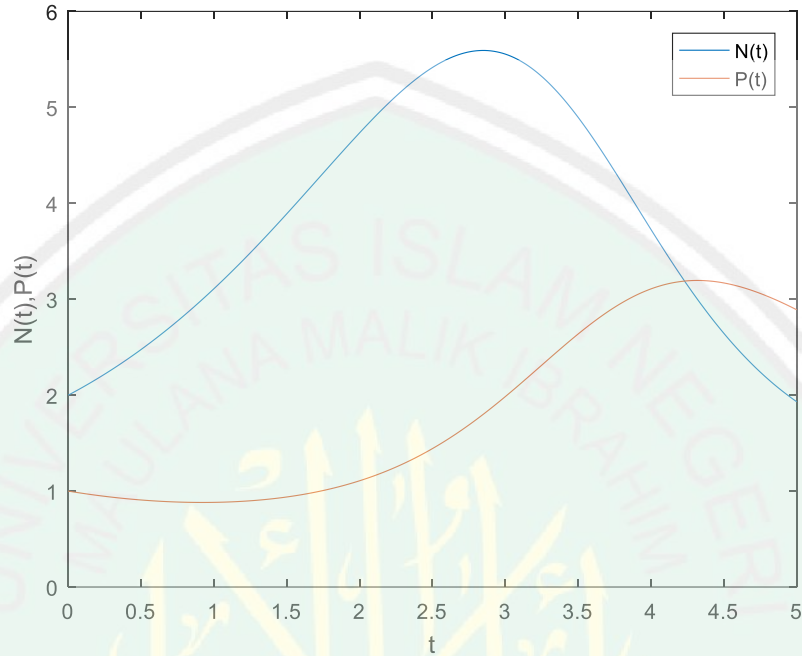
Gambar 3.4 Galat Solusi Numerik (ε) Model *Predator-Prey* (3.54) dengan $\Delta t = 0.01$

Gambar 3.4 menunjukkan galat mutlak ε dari solusi model *predator-prey* (3.54) dengan metode jaringan fungsi radial basis. Dari simulasi menggunakan program Matlab diperoleh galat mutlak maksimum $\varepsilon = 0,022$.

Dari hasil simulasi di atas sudah dapat disimpulkan bahwa solusi yang diperoleh sudah mendekati solusi eksaknya. Ini terbukti dengan diperolehnya galat dari solusi yang sudah sangat kecil. Model *predator-prey* merupakan gambaran dari populasi dari hewan pemangsa dan mangsanya. Sehingga galat sebesar 0,022 sudah dianggap cukup kecil karena sudah lebih kecil dibandingkan dengan 10% dari nilai awal terkecil yaitu 1, dilambangkan $0,022 < \frac{4}{10}$.

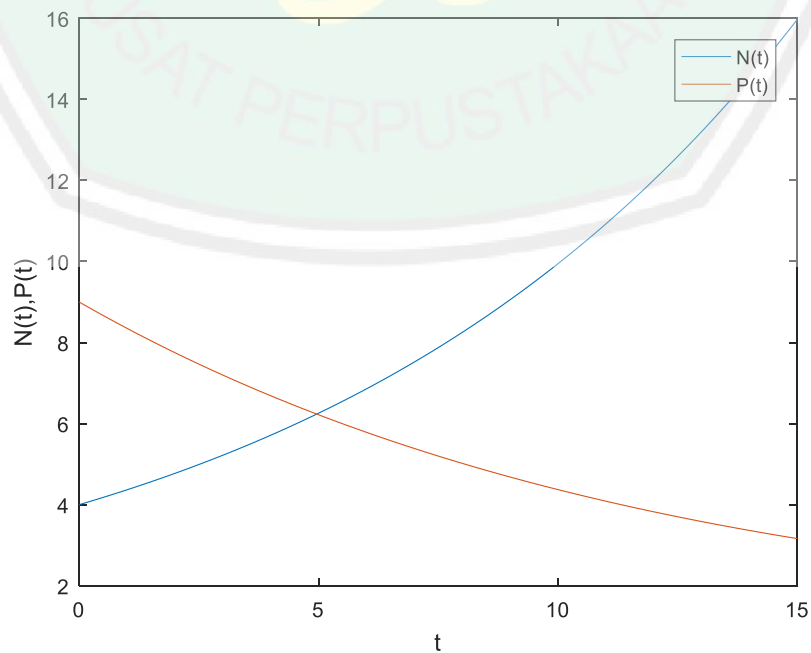
Analisis galat metode jaringan fungsi radial basis pada solusi numerik model *predator-prey* pertama pada persamaan (3.38) dan (3.54) dapat diketahui dari beberapa simulasi yang dilakukan, dengan memperbesar dan memperkecil Δt . Solusi numerik model *predator-prey* pertama (3.38) dengan kondisi awal (3.39)

untuk $\Delta t = 0,02$ dilakukan dengan program Matlab. Gambar solusi numeriknya dapat dilihat pada gambar berikut:



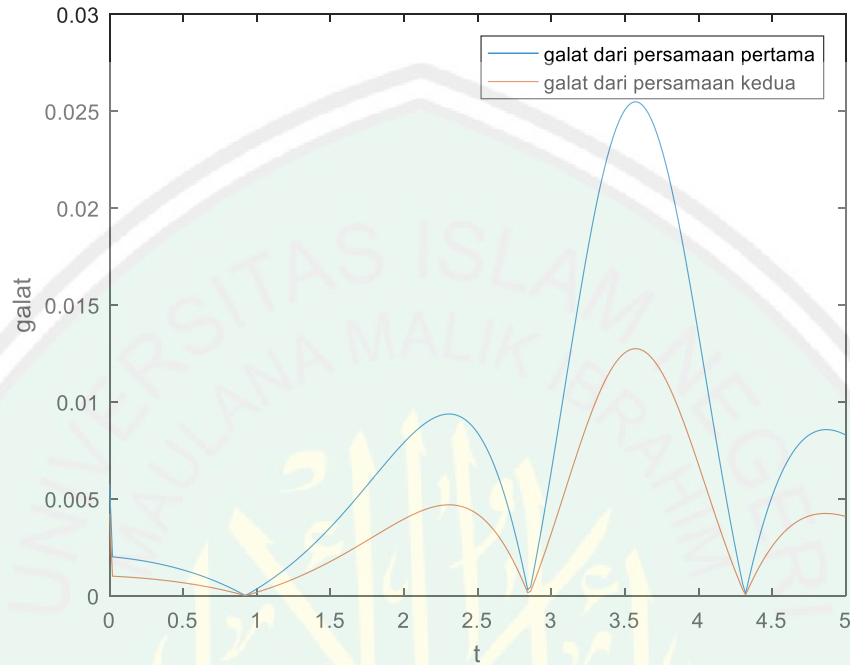
Gambar 3.5 Solusi Numerik Model *Predator-Prey* (3.38) dengan $\Delta t = 0,02$

Untuk model kedua pada persamaan (3.54) dengan kondisi awal pada persamaan (3.55) diperoleh solusi numerik berikut:

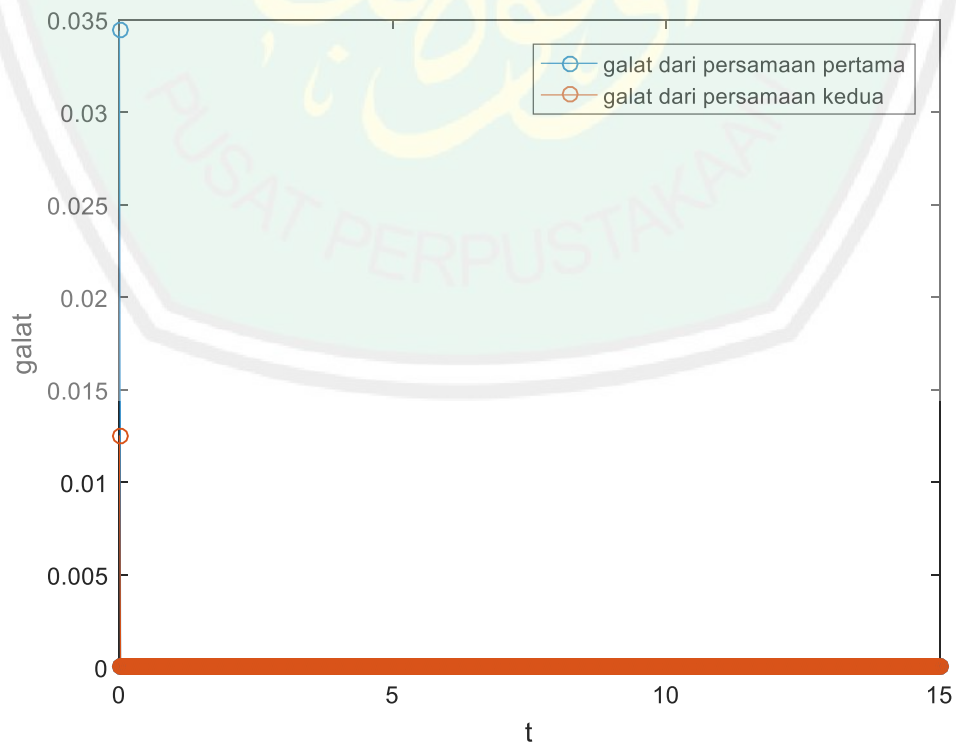


Gambar 3.6 Solusi Numerik Model *Predator-Prey* (3.54) dengan $\Delta t = 0,02$

Galat ε dari model *predator-prey* pada persamaan (3.38) dan (3.54) berturut-turut ditunjukkan oleh gambar-gambar berikut:



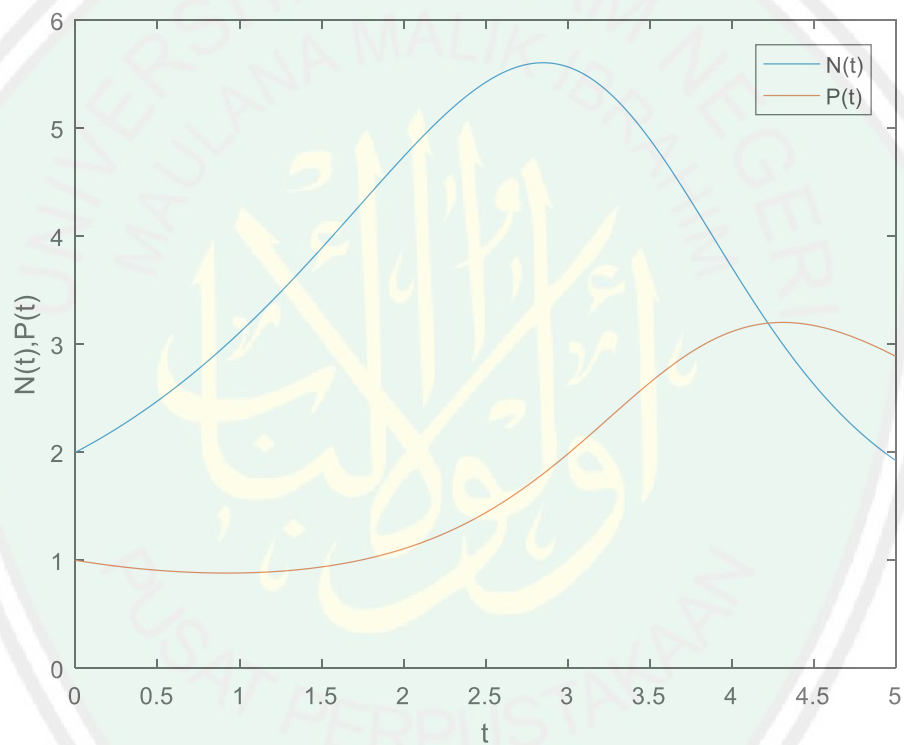
Gambar 3.7 Galat ε Solusi Numerik Model *Predator-Prey* (3.38) dengan $\Delta t = 0,02$



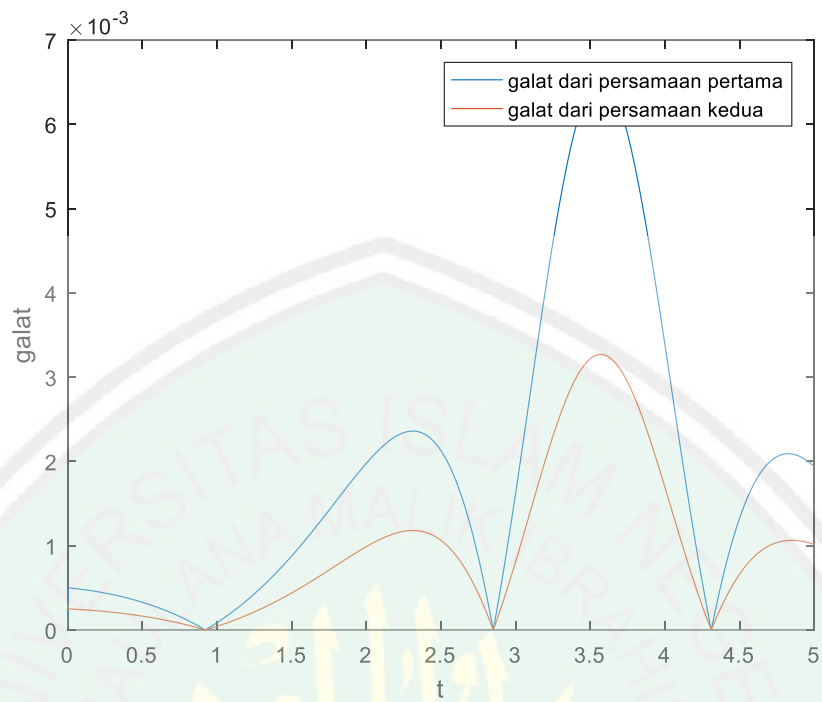
Gambar 3.8 Galat ε Solusi Numerik Model *Predator-Prey* (3.54) dengan $\Delta t = 0,02$

Galat mutlak maksimum ε diketahui dari program Matlab untuk model (3.38) yaitu $\varepsilon = 0,0255$, sedangkan untuk model (3.54) didapat $\varepsilon = 0,0344$ galat tersebut lebih besar daripada ketika $\Delta t = 0,01$.

Simulasi selanjutnya dilakukan dengan cara memperkecil Δt , yaitu dengan $\Delta t = 0,005$. Solusi numerik model pertama pada persamaan (3.38) beserta galatnya dapat dilihat pada gambar-gambar berikut:

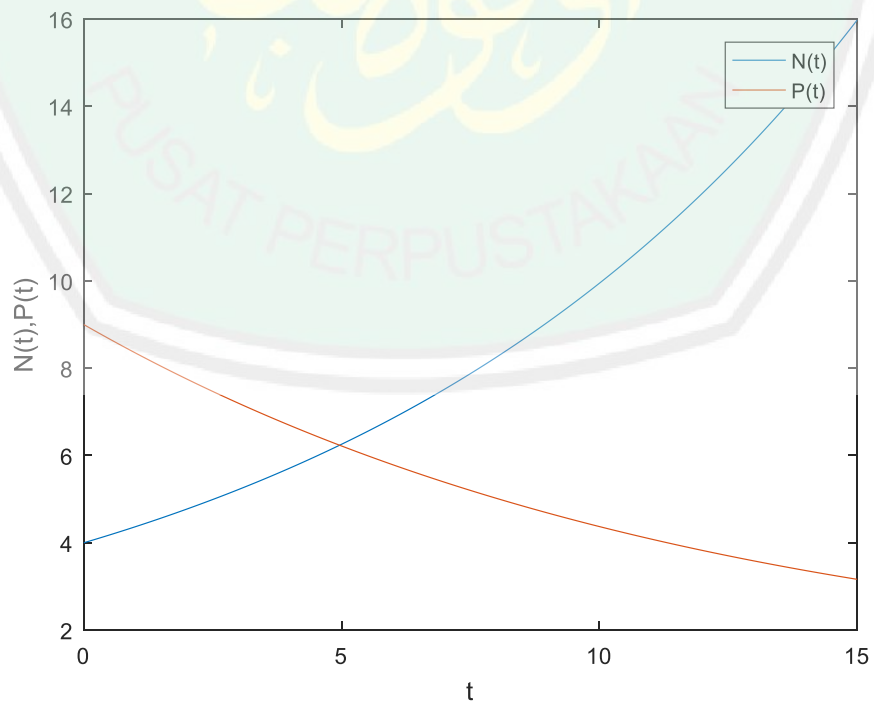


Gambar 3.9 Solusi Numerik Model *Predator-Prey* (3.38) dengan $\Delta t = 0,005$

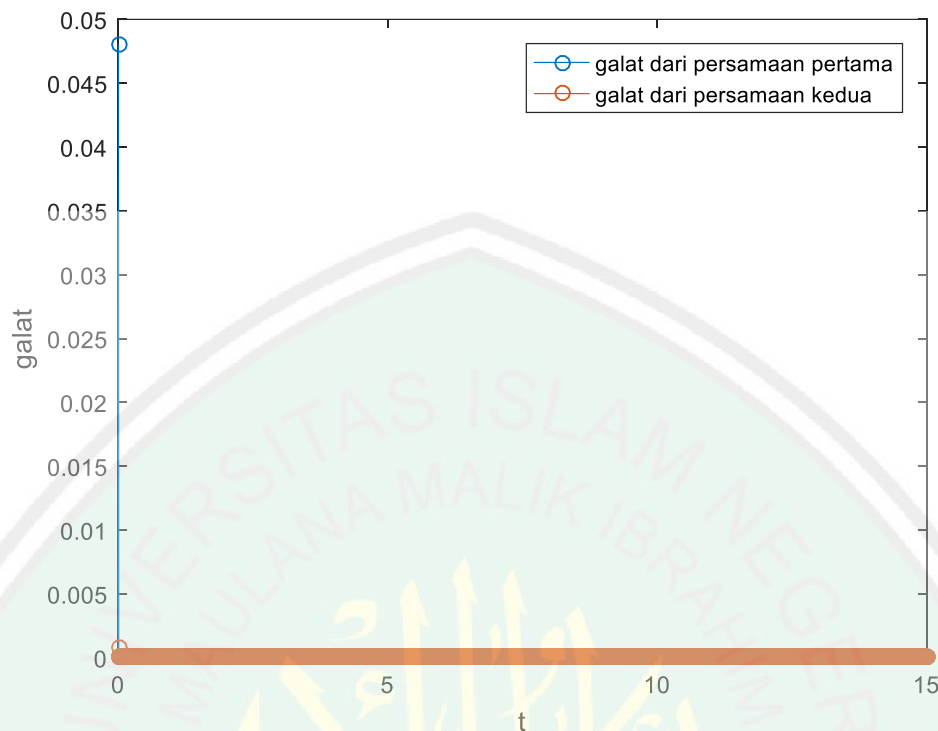


Gambar 3.10 Galat ϵ Solusi Numerik Model *Predator-Prey* (3.38) dengan $\Delta t = 0,005$

Solusi numerik model (3.54) beserta galatnya dapat dilihat pada gambar-gambar berikut:



Gambar 3.11 Solusi Numerik Model *Predator-Prey* (3.54) dengan $\Delta t = 0,005$



Gambar 3.12 Galat ε Solusi Numerik Model *Predator-Prey* (3.54) dengan $\Delta t = 0,005$

Galat mutlak maksimum untuk model pertama pada persamaan (3.38) diketahui dari program Matlab yaitu $\varepsilon = 0,0066$. Sedangkan galat mutlak maksimum untuk model kedua pada persamaan (3.54) diketahui dari program Matlab yaitu dan $\varepsilon = 0,0480$. Galat ε dari model (3.38) lebih kecil daripada ketika $\Delta t = 0,01$ ataupun $\Delta t = 0,02$. Akan tetapi galat dari persamaan (3.54) dengan $\Delta t = 0,005$ lebih besar daripada galat ketika $\Delta t = 0,01$.

Hasil dari beberapa simulasi tersebut dapat memberikan kesimpulan bahwa besarnya galat belum tentu dipengaruhi oleh besar atau kecilnya Δt . Hasil galat terkecil solusi model (3.54) dari beberapa simulasi yang telah dilakukan adalah ketika $\Delta t = 0,01$. penjelasan matematis mengenai analisis galat ini, dapat dilakukan penelitian selanjutnya untuk meneliti Δt yang optimal sehingga diperoleh galat mutlak yang minimal. Meskipun demikian, hasil beberapa simulasi tersebut sudah cukup menunjukkan bahwa metode jaringan fungsi radial basis cukup efektif

dalam mengaproksimasi solusi model *predator-prey* (3.38) maupun (3.54) walaupun terdapat galat pada masing-masing simulasi. Mengingat solusi numerik hanyalah solusi pendekatan saja di mana nilai atau hasil yang diperoleh bukan merupakan solusi analitik atau solusi eksak sehingga terdapat selisih yang tidak lain merupakan galat.

Dalam simulasi yang telah dilakukan masih terdapat kekurangan, yaitu pencarian nilai-nilai bobot tidak selalu konvergen. Pencarian nilai-nilai bobot pada simulasi yang telah dilakukan hanya akan konvergen ketika $t \leq 5$. Ketika $t = 6$ pencarian nilai bobot tidak akan konvergen sampai perulangan ke-1000. Dan juga kekonvergenan nilai bobot dipengaruhi oleh parameter-parameter a , α , b , dan β . Ketika $a = 1$ dan $t = [0, 5]$ pencarian nilai bobot tidak lagi konvergen. Satu lagi yang mempengaruhi kekonvergenan ialah nilai dari Δt . Dalam simulasi yang telah dilakukan dengan mengganti $\Delta t = 0,5$ pencarian nilai bobot menjadi tidak konvergen. Dapat disimpulkan bahwa yang mempengaruhi kekonvergenan dalam mencari nilai bobot ialah nilai-nilai parameter a , α , b , dan β , panjang dari t , dan nilai dari Δt . Untuk menganalisis kekonvergenan pencarian nilai bobot ini, dapat dilakukan penelitian selanjutnya untuk meneliti kombinasi dari faktor-faktor yang mempengaruhi kekonvergenan yang telah disebutkan.

3.8 Solusi Numerik dalam Pandangan Islam

Aproksimasi solusi model *predator-prey* menggunakan jaringan fungsi radial basis adalah menghampiri solusi model *predator-prey* dengan persamaan jaringan fungsi radial basis. Hasil dari aproksimasi solusi model *predator-prey*

disebut dengan solusi numerik, yang pada konsep estimasi menghasilkan nilai taksiran. Allah Swt. berfirman dalam al-Quran surat ash-Shaffaat ayat 147:

وَأَرْسَلْنَاهُ إِلَىٰ مِائَةِ أَلْفٍ أَوْ يَزِيدُونَ^{١٤٧}

Artinya: “Dan Kami utus dia kepada seratus ribu orang atau lebih”(QS. ash-Shaffaat/37:147).

Terjemah dari ayat tersebut dapat diketahui bahwa nabi Yunus diutus oleh Allah Swt. kepada umatnya yang berjumlah 100.000 orang atau lebih. Terjemah dari ayat tersebut jika dibaca dengan seksama, ada rasa atau kesan bahwa terdapat keraguan dalam menyatakan jumlah umat nabi Yunus. Mengapa harus menyatakan 100.000 atau lebih? Mengapa tidak menyatakan jumlah yang sebenarnya? Bukankah Allah Swt. Maha Mengetahui segala sesuatu termasuk jumlah umat nabi Yunus? Jawaban dari pertanyaan tersebut adalah “inilah estimasi (taksiran)” (Abdussakir, 2006).

Melalui ayat tersebut dapat ditangkap bahwa Allah Swt. sebenarnya telah mengajarkan suatu ilmu dalam matematika yang dikenal dengan estimasi (taksiran) dan juga aproksimasi (pendekatan). Keterampilan estimasi dan aproksimasi dalam kehidupan sehari-hari sangat dibutuhkan dan menghemat waktu dalam perhitungan. Dengan adanya konsep estimasi dan aproksimasi maka dapat menjadi alternatif mencari solusi numerik suatu permasalahan matematika tanpa mencari solusi eksaknya (Abdussakir, 2006).

Kata *مِائَةِ أَلْفٍ أَوْ يَزِيدُونَ* yang berarti seratus ribu atau lebih jika dihayati terdapat kesan ketidakpastian akan jumlah tersebut. Bisa jadi jumlahnya memang seratus ribu atau lebih dari seratus ribu. Ayat tersebut menunjukkan bahwa seratus ribu bukanlah jumlah yang sebenarnya, namun jumlah taksiran yang dalam ilmu

matematika biasa disebut dengan solusi numerik karena diperoleh dari proses numerik atau proses pendekatan.

Konsep estimasi dan pendekatan pada ayat tersebut sama dengan konsep solusi numerik model *predator-prey* dalam penelitian ini. Dengan adanya konsep estimasi inilah pencarian solusi masalah-masalah matematika dapat dengan mudah dilakukan. Pencarian solusi secara analitik akan membutuhkan waktu dan pemikiran yang lebih dari pada mengerjakan dengan secara numerik. Hal ini juga menjadi bukti bahwa al-Quran merupakan mukjizat, karena al-Quran sudah menginformasikan tentang konsep estimasi dan aproksimasi 14 abad yang lalu sebelum ilmuan-ilmuan menemukannya.

Ayat pada surat ash-Shaffat ayat 143 mengandung pendekatan numerik dalam mendekati jumlah umat nabi Yunus. Allah Swt. berfirman bahwa jumlah umat nabi Yunus berjumlah 100.000 atau lebih. Dari firman Allah Swt. tersebut dapat dihitung pendekatan jumlah umat nabi Yunus dengan konsep matematika. Misalkan jumlah umat nabi Yunus adalah $x + i$ yang mana x adalah jumlah sebenarnya umat nabi Yunus yang tidak dapat diketahui dengan pasti (eksak) dan i adalah batas galat yang dapat ditoleransi dalam konsep numerik yaitu 10% dari nilai yang diketahui (100.000), maka dapat dibangun asumsi-asumsi sebagai berikut:

1. x dapat lebih atau sama dengan 100.000 atau $x \geq 100.000$
2. i kurang dari 10.000 atau $i < \frac{10}{100} \times 100.000 = 10.000$ dan $i \geq 0$, jadi $0 \leq i < 10.000$

Dari dua asumsi yang sudah dibangun dapat disimpulkan bahwa estimasi jumlah umat nabi Yunus yang dapat dibenarkan oleh konsep matematika hanya ada di antara 100.000 sampai 110.000 saja, jumlah tersebut sudah termasuk dengan

galat yang dapat ditoleransi, atau dapat dituliskan bahwa $100.000 \leq x + i \leq 110.000$. Jumlah yang sudah diperoleh hanya taksiran dengan konsep matematika saja, dan belum dapat menggambarkan jumlah sebenarnya umat nabi Yunus.



BAB IV

PENUTUP

4.1 Kesimpulan

Model *predator-prey* menggambarkan interaksi antara populasi mangsa dan pemangsa. Solusi numerik model *predator-prey* pada skripsi ini diperoleh dengan metode jaringan fungsi radial basis. Model *predator-prey* dapat didekati secara langsung dengan sebuah fungsi basis jenis *multiquadrics* untuk mendapatkan solusi numeriknya.

Langkah-langkah penyelesaian numerik model *predator-prey* menggunakan jaringan fungsi radial basis yaitu: melinierkan model *predator-prey* menggunakan deret Taylor, mendiskritisasi model *predator-prey* menggunakan jaringan fungsi radial basis serta kondisi awalnya, menghitung nilai bobot w_i dan v_i , menghitung solusi numerik model *predator-prey*, melakukan simulasi, dan menggambar grafiknya.

Solusi numerik model *predator-prey* yang diperoleh dalam penelitian ini menunjukkan hasil yang cukup dekat dengan solusi eksaknya. Berdasarkan simulasi yang telah dilakukan, menunjukkan bahwa besarnya galat mutlak maksimum dari masing-masing simulasi tidak dipengaruhi oleh nilai Δt . Semakin kecil nilai Δt belum tentu menghasilkan galat yang lebih kecil. Untuk simulasi pertama, galat mutlak maksimum terkecil yang diperoleh dari simulasi yang dilakukan adalah 0,0066 dan galat ini lebih kecil dari 10% nilai awal terkecil yaitu 1 dilambangkan $0,0066 < \frac{1}{10}$, sedangkan untuk simulasi kedua, galat mutlak maksimum terkecil yang diperoleh dari simulasi yang dilakukan adalah 0,022 dan

galat ini lebih kecil dari 10% nilai awal terkecil yaitu 4 dilambangkan $0,022 < \frac{4}{10}$.

Dapat disimpulkan bahwa metode jaringan fungsi radial basis cukup efektif dalam mengaproksimasi solusi dari model *predator-prey*.

4.2 Saran

Saran untuk penelitian selanjutnya adalah meneliti tentang kekonvergenan solusi numerik model *predator-prey* menggunakan jaringan fungsi radial basis. Juga penelitian ini dapat ditingkatkan dengan menyelesaikan persamaan diferensial parsial nonlinier menggunakan jaringan fungsi radial basis.



DAFTAR RUJUKAN

- Abdussakir. 2006. *Ada Matematika dalam Al-Quran*. Malang: UIN-Maliki Press.
- Al-Alusi. 1981. *Ruuhul Ma'aanii*. Beirut: Atthiba'ah Muniriyah.
- Al-Qurthubi, A. 2006. *Al-Jami' li Ahkaamil Quran*. Beirut: Al-Resalah.
- Aminataei, A. dan Mazarei, M.M. 2008. Numerical Solution of Poisson's Equation Using Radial Basis Function Networks on the Polar Coordinate. *Computers and Mathematics with Applications*, 56: 2887-2895.
- Azzamkhasyari, M. 1998. *Tafsir Al-Kasyaf Juz 5*. Riyadh: Maktabah Al-'Abikan.
- Boyce, W.E. dan DiPrima, R.C. 2012. *Elementary Differential Equations 10th Edition*. New York: John Wiley & Sons, Inc.
- Dwaradi, H. 2011. *Analisis Model Mangsa-Pemangsa Michaelis-Menten dengan Pemanenan pada Populasi Mangsa*. Skripsi Tidak Diterbitkan. Bogor: Institut Pertanian Bogor.
- Finizio, N. dan Ladas, G. 1988. *Persamaan Diferensial Biasa dengan Penerapan Modern*. Jakarta: Erlangga.
- Fitriya, R. 2011. *Penyelesaian Numerik Persamaan Poisson Menggunakan Jaringan Fungsi Radial Basis*. Skripsi Tidak Diterbitkan. Malang: UIN Maulana Malik Ibrahim Malang.
- Hajek, M. 2005. *Neural Networks*. Modul Kuliah Tidak Diterbitkan. California: University of California.
- Jamhuri, M. 2011. Penyelesaian Numerik Persamaan Diferensial Biasa Menggunakan Jaringan Fungsi Radial Basis. *Gamatika*, 2: 162-170.
- Kansa, E.J. dan Carlson R.E. 1992. Improved Accuracy of Multiquadric Interpolation Using Variable Shape Parameters. *Computers Math. Applic*, 24: 99-120.
- Kansa, E.J. 1990. Multiquadrics- A Scattered Data Approximation Scheme with Applications to Computational Fluid-Dynamics-II: Solutions to Parabolic, Hyperbolic and Elliptic Partial Differential Equations. *Computers Math. Applic*, 19: 147-161.
- Katsir, I. 1994. *Tafsir Ibnu Katsir Jilid 7*. Terjemahan Abdullah. Jakarta: Pustaka Imam asy-Syafi'i.

- Mai-Duy, N. dan Tran-Cong, T. 2003. Approximation of Function and Its Derivatives Using Radial Basis Functions Networks. *Applied Mathematical Modelling*, 27: 197-220.
- Mufidah, F. dan Jamhuri, M. 2015. Solusi Numerik Persamaan Poisson Menggunakan Jaringan Fungsi Radial Basis pada Koordinat Polar. *Jurnal Cauchy*, 3 (4): 45-52.
- Paul, S., Mondal, S.P., dan Bhattacharya, P. 2016. Numerical Solution of Lotka Volterra Prey-Predator Model by Using Runge–Kutta–Fehlberg Method and Laplace Adomian Decomposition Method. *Alexandria Engineering Journal*, 55: 613-617.
- Putri, P.P. 2013. *Analisis Solusi Numerik Model Predator-Prey Dengan Metode Runge-Kutta Orde Empat dan Gill*. Skripsi Tidak Diterbitkan. Jember: Universitas Jember.
- Redjeki, P.S. 2009. *Diktat Kuliah MA2271 Metode Matematika*. Bandung: Fakultas MIPA ITB.
- Sarra, S.A. dan Kansa, E. 2009. *Multiquadrics Radial Basis Function Approximation Methods for the Numerical Solution of Partial Differential Equations*. Modul Kuliah Tidak Diterbitkan. California: University of California.
- Setiawan, I. 2002. *Jaringan Syaraf Tiruan Jenis AMN (Associative Memory Networks): CMAC, B-Spline dan RBF untuk Aplikasi Pemodelan dan Pengontrolan*. Modul Kuliah Tidak Diterbitkan. Semarang: Universitas Diponegoro.
- Sarra, S.A. dan Sturgill, D. 2009. A Random Variable Shape Parameter Strategy for Radial Basis Function Approximation Methods. *Engineering Analysis with Boundary Elements*, 33: 1239-1245.
- Xiang, S., Wang, K., Sha, Y., dan Shi, H. 2012. Trigonometric Variable Shape Parameter and Exponent Strategy for Generalized Multiquadric Radial Basis Function Approximation. *Applied Mathematical Modelling*, 36: 1931-1938.

Lampiran 1. M-File Matlab Untuk Solusi Numerik Model Predator-Prey (Simulasi Pertama)

```
clc,clear,clf

dt = 0.01;
t = 0:dt:5;
m = length(t);
c = t;

%parameter
a = 0.9;
b = 0.75;
alpha = 0.5;
beta = 0.25;

%nilai awal
N0 = 2;
P0 = 1;

[T,Z] = ode45(@pp1,t,[N0 P0]);

% w0 = randn(m,1);
% v0 = randn(m,1);

phi = mq(t,c);
phit = mqt(t,c);

% W0 = [w0;v0];

h1 = phit-a*phi;
h = [phi(1,:);h1(2:m,:)];

e1 = phit+b*phi;
e = [phi(1,:);e1(2:m,:)];

pn1 = sin(t)+N0;
pn = zeros(m);
for j = 1:m
    pn(j,:) = phi(j, :)*pn1(j);
end

pnt1 = cos(t);
pnt = zeros(m);
for j = 1:m
    pnt(j,:) = phit(j, :)*pnt1(j);
end

g1 = alpha*pn+alpha*pnt*dt;
g = [zeros(1,m) ; g1(2:m,:)];

np1 = sin(t)+P0;
np = zeros(m);
for j = 1:m
```



```

    np(j,:) = phi(j,:)*np1(j);
end

npt1 = cos(t);
npt = zeros(m);
for j = 1:m
    npt(j,:) = phit(j,:)*npt1(j);
end

f1 = beta*np+beta*npt*dt;
f = [zeros(1,m) ; f1(2:m,:)];

A = [h g;-f e];
B = [N0;zeros(m-1,1);P0;zeros(m-1,1)];

W = A\B;

W0 = W;
w0 = W0(1:m);
v0 = W0(m+1:2*m);

N = phi*w0;
P = phi*v0;

galat_per_iterasi = [];

for i = 1:100
    pn1 = phi*w0;
    pn = zeros(m);
    for j = 1:m
        pn(j,:) = phi(j,:)*pn1(j);
    end

    pnt1 = phit*w0;
    pnt = zeros(m);
    for j = 1:m
        pnt(j,:) = phit(j,:)*pnt1(j);
    end

    g1 = alpha*pn+alpha*pnt*dt;
    g = [zeros(1,m) ; g1(2:m,:)];

    np1 = phi*v0;
    np = zeros(m);
    for j = 1:m
        np(j,:) = phi(j,:)*np1(j);
    end

    npt1 = phit*v0;
    npt = zeros(m);
    for j = 1:m
        npt(j,:) = phit(j,:)*npt1(j);
    end

    f1 = beta*np+beta*npt*dt;
    f = [zeros(1,m) ; f1(2:m,:)];
end

```

```

A = [h g;-f e];
B = [N0;zeros(m-1,1);P0;zeros(m-1,1)];

W = A\B;

W0 = W;
w0 = W0(1:m);
v0 = W0(m+1:2*m);

N = phi*w0;
P = phi*v0;

plot(t,N,t,P)
ylabel('N(t),P(t)')
xlabel('t')
legend('N(t)', 'P(t)')

galatmax1 = max(abs(phit*w0-a*phi*w0+alpha*(phi*w0).*(phi*v0)));
galatmax2 = max(abs(phit*v0+b*phi*v0-beta*(phi*w0).*(phi*v0)));

pause(0.1)
batas = max(abs([pn1;np1]-[N;P]));
if batas<0.0001
    break
end
end

% penghitungan error

eror1 = abs(phit*w0-a*phi*w0+alpha*(phi*w0).*(phi*v0));
eror2 = abs(phit*v0+b*phi*v0-beta*(phi*w0).*(phi*v0));

galat_akhir_max = max([eror1;eror2]);

figure(3)
plot(t,eror1,t,eror2)
legend('galat dari persamaan pertama', 'galat dari persamaan kedua')
ylabel('galat')
xlabel('t')

```

Lampiran 2. M-File Matlab Untuk Solusi Numerik Model Predator-Prey (Simulasi Kedua)

```
clc,clear,clf

dt = 0.01;
t = 0:dt:15;
m = length(t);
c = t;

%parameter
a = 0.1;
b = 0.08;
alpha = 0.0014;
beta = 0.0012;

%nilai awal
N0 = 4;
P0 = 9;

w0 = randn(m,1);
v0 = randn(m,1);

phi = mq(t,c);
phit = mqt(t,c);

W0 = [w0;v0];

h1 = phit-a*phi;
h = [phi(1,:);h1(2:m,:)];

e1 = phit+b*phi;
e = [phi(1,:);e1(2:m,:)];

for i = 1:1000
    pn1 = phi*w0;
    pn = zeros(m);
    for j = 1:m
        pn(j,:) = phi(j, :)*pn1(j);
    end

    pnt1 = phit*w0;
    pnt = zeros(m);
    for j = 1:m
        pnt(j,:) = phit(j, :)*pnt1(j);
    end

    g1 = alpha*pn+alpha*pnt*dt;
    g = [zeros(1,m) ; g1(2:m,:)];

    np1 = phi*v0;
    np = zeros(m);
    for j = 1:m
        np(j,:) = phi(j, :)*np1(j);
    end
end
```

```

npt1 = phit*v0;
npt = zeros(m);
for j = 1:m
    npt(j,:) = phit(j,:)*npt1(j);
end

f1 = beta*np+beta*npt*dt;
f = [zeros(1,m) ; f1(2:m,:)];

A = [h g;-f e];
B = [N0;zeros(m-1,1);P0;zeros(m-1,1)];

W = A\B;

W0 = W;
w0 = W0(1:m);
v0 = W0(m+1:2*m);

N = phi*w0;
P = phi*v0;

figure(1)
plot(t,N,t,P)
ylabel('N(t),P(t)')
xlabel('t')
legend('N(t)', 'P(t)')

galatmax1 = max(abs(phit*w0-a*phi*w0+alpha*(phi*w0).*(phi*v0)));
galatmax2 = max(abs(phit*v0+b*phi*v0-beta*(phi*w0).*(phi*v0)));

pause(0.1)

batas = max(abs([pn1;np1]-[N;P]));
if batas<0.0001
    break
end
end

% penghitungan error

eror1 = abs(phit*w0-a*phi*w0+alpha*(phi*w0).*(phi*v0));
eror2 = abs(phit*v0+b*phi*v0-beta*(phi*w0).*(phi*v0));

galat_akhir_max = max([eror1;eror2]);

figure(3)
plot(t,eror1,'-o',t,eror2,'-o')
legend('galat dari persamaan pertama','galat dari persamaan kedua')
ylabel('galat')
xlabel('t')

```

RIWAYAT HIDUP

Muhammad Thahiruddin, lahir di sebuah klinik di Desa Keppo yang masih masuk daerah Kabupaten Pamekasan pada tanggal 21 Februari 1994, biasa dipanggil Thohir, tinggal di Dusun Kolor, Desa Rombasan, Kecamatan Pragaan, Kabupaten Sumenep. Anak pertama dari dua bersaudara dari bapak Syaikhul Umam dan ibu Atiqoh.

Pendidikan dasarnya ditempuh di MI Al-Ghazali Rombasan sekaligus SDN Kertagena Laok II dan lulus pada tahun 2005, setelah itu dia mondok di Pondok Pesantren Mambaul Ulum Bata-Bata dan melanjutkan pendidikan menengahnya di MTs Mambaul Ulum Bata-Bata. Kemudian dia melanjutkan pendidikan ke MA Mambaul Ulum Bata-Bata dan lulus tahun 2011. Setelah itu dia mengabdikan menjadi guru yang merupakan tugas wajib dari pondok pesantren selama satu tahun di Desa Tellok, Kecamatan Galis, Kabupaten Bangkalan. Selanjutnya, pada tahun 2012 menempuh kuliah di Universitas Islam Negeri Maulana Malik Ibrahim Malang mengambil Jurusan Matematika.

Selama menempuh pendidikan tingkat dasar sampai tingkat perguruan tinggi dia mendapat sedikit prestasi yang tidak begitu gemilang. Prestasi yang pernah dia raih di antaranya, Juara I Olimpiade Matematika tingkat SD se-Kecamatan Kadur pada tahun 2004, Juara I lomba Matematika antar Pondok Pesantren se-Kecamatan Palengaan tahun 2009, juara III *Musabaqoh Fahmil Quran* pada *Musabaqoh Ma'hadiyah* di Pondok Pesantren Mambaul Ulum Bata-Bata tahun 2010, Juara II *Musabaqoh Fahmil Quran* pada ajang *Musabaqoh Tilawatil Quran* (MTQ) Nasional tingkat Kabupaten Pamekasan pada tahun 2010, dan Juara I *Musabaqoh Fahmil Quran* se-Jawa Timur pada Olimpiade al-Quran yang dilaksanakan oleh *Hai'ah Tahfidzul Quran* (HTQ) UIN Maulana Malik Ibrahim Malang tahun 2011.



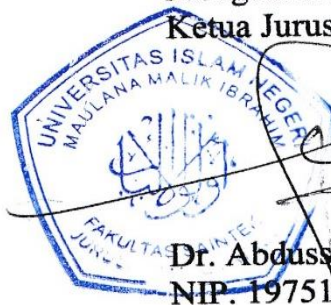
KEMENTERIAN AGAMA RI
UNIVERSITAS ISLAM NEGERI
MAULANA MALIK IBRAHIM MALANG
FAKULTAS SAINS DAN TEKNOLOGI
Jl. Gajayana No. 50 Dinoyo Malang Telp./Fax.(0341)558933

BUKTI KONSULTASI SKRIPSI

Nama : Muhammad Thahiruddin
Nim : 12610096
Fakultas/Jurusan : Sains dan Teknologi/ Matematika
Judul Skripsi : Solusi Nuemerik Model Predator-Prey Menggunakan Metode Jaringan Fungsi Radial basis
Pembimbing I : Mohammad Jamhuri, M.Si
Pembimbing II : Ach. Nashichuddin, MA

	Tanggal	Hal	Tanda Tangan
1.	24 Maret 2016	Konsultasi Bab I & Bab II	1.
2.	05 April 2016	Revisi Bab I & Bab II	2.
3.	12 April 2016	Konsultasi Bab III	3.
4.	12 April 2016	Konsultasi Agama Bab I & II	4.
5.	13 April 2016	Revisi Agama Bab I & II	5.
6.	27 Mei 2016	Revisi Bab III	6.
7.	27 Mei 2016	Konsultasi Agama Bab III	7.
8.	30 Mei 2016	Revisi Agama Bab III	8.
9.	31 Mei 2016	Acc Keseluruhan	9.
10.	31 Mei 2016	Acc Agama Keseluruhan	10.

Malang, 31 Mei 2016
Mengetahui,
Ketua Jurusan Matematika



Dr. Abdussakir, M.Pd
NIP. 19751006 200312 1 001