



**BIFURKASI HOPF PADA MODIFIKASI MODEL
PREDATOR-PREY LESLIE GOWER DENGAN
FUNGSI RESPON HOLLING TIPE II**

skripsi

disajikan sebagai salah satu syarat
untuk memperoleh gelar Sarjana Sains
Program Studi Matematika

oleh

Gesti Essa Waldhani

4111411050

**JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS NEGERI SEMARANG
2015**

PERNYATAAN

Dengan ini saya menyatakan bahwa skripsi yang berjudul “Bifurkasi Hopf pada Modifikasi Model *Predator-Prey* Leslie Gower dengan Fungsi Respon Holling Tipe II” bebas plagiat, dan apabila di kemudian hari terbukti terdapat plagiat dalam skripsi ini, maka saya akan bersedia menerima sanksi sesuai ketentuan perundang-undangan.

Semarang, 20 Februari 2015




Gesti Essa Waldhani

4111411050

PENGESAHAN

Skripsi yang berjudul
Bifurkasi Hopf pada Modifikasi Model *Predator-Prey* Leslie Gower dengan
Fungsi Respon Holling Tipe II
disusun oleh
Gesti Essa Waldhani
4111411050
telah dipertahankan di hadapan sidang Panitia Ujian Skripsi Fakultas Matematika
dan Ilmu Pengetahuan Alam Universitas Negeri Semarang pada tanggal 17
Februari 2015.



Prof. Dr. Wiyanto, M.Si
NIP. 196310121988031001

Ketua Penguji



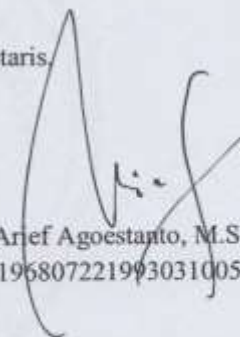
Drs. Supriyono, M.Si
NIP. 195210291980031002

Anggota Penguji/
Pembimbing I



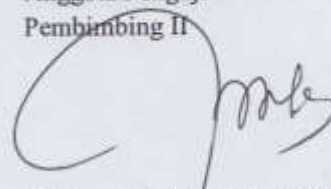
Prof. Dr. St. Budi Waluya, M.Si.
NIP. 196809071993031003

Sekretaris



Drs. Arief Agoestanto, M.Si
NIP. 196807221993031005

Anggota Penguji/
Pembimbing II



Muhammad Kharis, S.Si, M.Sc.
NIP. 198210122005011001

MOTTO DAN PERSEMBAHAN

MOTTO

- ✚ Hanya kepada Engkaulah kami menyembah dan hanya kepada Engkaulah kami mohon pertolongan (Al-Fatihah: 5).
- ✚ “Boleh jadi kamu membenci sesuatu, padahal itu amat baik untukmu, dan boleh jadi (pula) kamu menyukai sesuatu padahal itu amat buruk bagimu.” (Q.S. Al Baqarah: 216)
- ✚ Satu langkah lebih berarti daripada hanya diam.
- ✚ Jadilah pemenang untuk setiap tantangan.
- ✚ Kegagalan merupakan batu loncatan untuk sebuah keberhasilan.

PERSEMBAHAN

- ✚ Untuk Ibu tercinta yang tak pernah berhenti mendoakan keberhasilan putrinya.
- ✚ Untuk Pakde dan Bude yang selalu memberikan kasih sayang, dukungan dan semangat.
- ✚ Untuk Adik perempuanku yang tersayang.
- ✚ Untuk Adik sepupukuku yang tersayang.
- ✚ Untuk Ika, Yanti, Nikmah, Iin, Susanti, Dwi, Ari, Rulli dan Nilam yang selama ini memberikan dukungan dan semangat.
- ✚ Untuk teman-teman matematika angkatan 2011 yang menemaniku dalam berjuang menghadapi semua tantangan dan rintangan.

KATA PENGANTAR

Puji syukur senantiasa terucap ke hadirat Allah atas segala rahmat-Nya dan sholawat selalu tercurah atas Muhammad Rasulullah SAW hingga akhir jaman. Pada kesempatan ini, penulis dengan penuh syukur mempersembahkan skripsi dengan judul “Bifurkasi Hopf pada Modifikasi Model *Predator-Prey* Leslie Gower dengan Fungsi Respon Holling Tipe II”. Penulisan skripsi ini sebagai syarat mutlak yang harus dipenuhi oleh penulis untuk memperoleh gelar Sarjana Sains di Universitas Negeri Semarang.

Penulisan skripsi ini dapat terselesaikan karena adanya bimbingan, bantuan dan dukungan dari berbagai pihak baik secara langsung maupun tidak langsung. Oleh karena itu, penulis mengucapkan terima kasih kepada:

1. Prof. Dr. Fatkhur Rokhman, M.Hum., Rektor Universitas Negeri Semarang.
2. Prof. Dr. Wiyanto, M.Si., Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Negeri Semarang.
3. Drs. Arief Agoestanto, M.Si., Ketua Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Negeri Semarang.
4. Dra. Kristina Wijayanti, M.Si., Ketua Prodi Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Negeri Semarang.
5. Prof. Dr. St. Budi Waluya, M.Si., Dosen Pembimbing I yang telah memberikan bimbingan, semangat dan pengarahan.
6. Muhammad Kharis, S.Si, M.Sc., Dosen Pembimbing II yang telah memberikan masukan dan arahan.

7. Drs. Supriyono, M.Si, Dosen Penguji yang telah memberikan inspirasi, kritik, saran dan motivasi.

Penulis menyadari bahwa skripsi yang disusun dan disampaikan masih memiliki banyak kekurangan dan jauh dari sempurna. Untuk itu penulis menerima segala kritik dan saran yang sifatnya membangun untuk skripsi ini. Semoga skripsi ini dapat memberi manfaat bagi penulis dan para pembaca.

Semarang, 20 Februari 2015

Gesti Essa Waldhani

ABSTRAK

Waldhani, G. E. 2015. *Bifurkasi Hopf pada Modifikasi Model Predator-Prey Leslie Gower dengan Fungsi Respon Holling Tipe II*. Skripsi, Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Negeri Semarang. Pembimbing I Prof. Dr. St. Budi Waluya, M.Si. dan Pembimbing II Muhammad Kharis, S.Si, M.Sc.

Kata kunci: bifurkasi Hopf, *predator-prey*, Leslie Gower, Holling tipe II, *limit cycle*.

Skripsi ini membahas model *predator-prey* Leslie Gower dengan fungsi respon Holling tipe II dan keberadaan dari bifurkasi Hopf.

Permasalahan yang diangkat dalam penelitian ini adalah bagaimana menurunkan model matematika dari modifikasi sistem *predator-prey* Leslie Gower dengan fungsi respon Holling tipe II, bagaimana menentukan titik kesetimbangan serta analisis kestabilan pada modifikasi model *predator-prey* Leslie Gower dengan fungsi respon Holling tipe II, bagaimana pengaruh perubahan parameter terhadap keadaan yang sebenarnya dari modifikasi model *predator-prey* Leslie Gower dengan fungsi respon Holling tipe II dan bagaimana simulasi numerik dari modifikasi model *predator-prey* Leslie Gower dengan fungsi respon Holling tipe II menggunakan *software* Maple.

Metode yang digunakan untuk menganalisis masalah adalah dengan studi pustaka. Langkah-langkah yang digunakan adalah menurunkan model matematika dari modifikasi sistem *predator-prey* Leslie Gower dengan fungsi respon Holling tipe II, menentukan semua titik tetap, menentukan persamaan karakteristik dan nilai eigen dari matriks Jacobian, menentukan nilai parameter terjadinya bifurkasi Hopf, menghitung syarat transversal, membuat simulasi numerik dari modifikasi model *predator-prey* Leslie Gower dengan fungsi respon Holling tipe II menggunakan *software* Maple dan penarikan kesimpulan.

Sebagai hasil penelitian, model yang diperoleh adalah

$$(1) \quad \frac{dx}{dt} = x \left(1 - x - \frac{\alpha y}{m+x} - \frac{h}{c+x} \right),$$

$$(2) \quad \frac{dy}{dt} = \rho y \left(1 - \frac{\beta y}{m+x} \right),$$

dengan $x(0) > 0$ dan $y(0) > 0$.

Dari model tersebut diperoleh enam titik kesetimbangan, yaitu $E_0, E_1, E_H, E_L, E_1^*$ dan E_2^* dengan asumsi terdapat titik ekuilibrium E_H dan E_L jika $(c+1)^2 > 4h, c < 1$ dan $h > c$. Kemudian hanya terdapat titik ekuilibrium E_H jika $(c+1)^2 > 4h, c < 1$ dan $h < c$, serta didefinisikan $E^* = E_1^*$ dengan syarat $\left(\frac{\alpha}{\beta} - c - 1\right)^2 > 4h + c^2$ dan $\frac{\alpha}{\beta} + c < 1$. Asumsi-asumsi tersebut mengakibatkan model mempunyai lima titik kesetimbangan, yaitu E_0, E_1, E_H, E_L dan E^* dengan syarat tersebut. Kemudian dari lima titik kesetimbangan tersebut diperoleh E^*

merupakan titik tetap yang mempunyai sepasang nilai eigen imajiner murni dengan syarat tertentu, sehingga dimungkinkan keberadaan bifurkasi Hopf pada titik E^* . Untuk menganalisis keberadaan bifurkasi Hopf, dinamika populasi *predator-prey* model ini dibagi menjadi 3 kasus dimana setiap kasusnya mengalami kenaikan nilai parameter tingkat pertumbuhan populasi *predator* ($\rho = w$). Dengan memilih nilai parameter yang tepat (w), dapat ditunjukkan keberadaan dari bifurkasi Hopf. Pada kasus ($w = w_0$) terjadi perubahan kestabilan titik tetap E^* dari spiral tidak stabil menjadi spiral stabil dan kemunculan *limit cycle*. Fenomena ini merupakan sifat dari bifurkasi Hopf. Kemudian pada kasus ketiga terjadi kestabilan terhadap kedua populasi yang disebabkan karena tingkat pertumbuhan populasi *predator* semakin tinggi. Secara umum, dapat disimpulkan bahwa kenaikan nilai parameter tingkat pertumbuhan *predator* mempengaruhi kestabilan populasi. Selanjutnya, untuk mengilustrasikan model tersebut maka dilakukan simulasi model dengan menggunakan *software* Maple. Simulasi model yang dilakukan memberikan hasil yang sama dengan hasil analisis.

DAFTAR ISI

	Halaman
HALAMAN JUDUL.....	i
PERNYATAAN.....	ii
PENGESAHAN	iii
MOTTO DAN PERSEMBAHAN	iv
KATA PENGANTAR	v
ABSTRAK	vii
DAFTAR ISI.....	ix
DAFTAR TABEL.....	xii
DAFTAR GAMBAR	xiii
DAFTAR LAMPIRAN.....	xvii
BAB 1 PENDAHULUAN	1
1. 1 Latar Belakang	1
1. 2 Rumusan Masalah.....	6
1. 3 Batasan Masalah	6
1. 4 Tujuan Penelitian	6
1. 5 Manfaat Penelitian	7
BAB 2 LANDASAN TEORI.....	8
2. 1 Sistem Persamaan Diferensial	8
2. 2 Model Pertumbuhan Logistik	9
2. 3 Model Populasi <i>Predator-Prey</i>	12

2. 4 Model Leslie Gower	14
2. 5 Fungsi Respon.....	14
2.5.1 Fungsi Respon Holling Tipe I.....	15
2.5.2 Fungsi Respon Holling Tipe II.....	17
2.5.3 Fungsi Respon Holling Tipe III	22
2. 6 Sistem Dinamik.....	27
2.6.1 Orbit	28
2.6.2 Titik Ekuilibrium.....	28
2.6.3 Solusi Periodik	29
2.6.4 <i>Limit Cycle</i>	29
2.6.5 Potret Fase	29
2. 7 Nilai Eigen dan Vektor Eigen.....	30
2. 8 Sistem Linear	31
2. 9 Kestabilan Sistem Linear	32
2.10 Sistem <i>Nonlinear</i>	33
2.11 Bifurkasi.....	34
2.12 Bifurkasi Hopf.....	35
2.13 Potret Fase dari Sistem Linear.....	36
2.14 Maple.....	41
BAB 3 METODE PENELITIAN.....	42
BAB 4 HASIL DAN PEMBAHASAN.....	44
4.1 Unsur-Unsur yang Berpengaruh Terhadap Model.....	44
4.2 Pembentukan Model <i>Predator-Prey</i> Leslie Gower dan Fungsi Respon Holling Tipe II.....	45

4.2.1 Model Dasar <i>Predator-Prey</i>	46
4.2.2 Fungsi Respon Holling Tipe II.....	47
4.2.3 Fungsi Logistik	49
4.3 Titik Ekuilibrium	53
4.4 Analisis Kestabilan Titik Ekuilibrium	58
4.4.1 Titik Ekuilibrium $E_0(0,0)$	59
4.4.2 Titik Ekuilibrium $E_1\left(0, \frac{m}{\beta}\right)$	61
4.4.3 Titik Ekuilibrium $E_H(x_H, 0)$	64
4.4.4 Titik Ekuilibrium $E_L(x_L, 0)$	73
4.4.5 Titik Ekuilibrium $E^*(x^*, y^*)$	80
4.5 Simulasi Model	89
4.5.1 Simulasi di Titik Ekuilibrium Tidak Terjadinya Bifurkasi Hopf.....	90
4.5.2 Simulasi di Titik Ekuilibrium Terjadinya Bifurkasi Hopf	101
BAB 5 PENUTUP	108
5.1 Simpulan	110
5.2 Saran	113
DAFTAR PUSTAKA	114
LAMPIRAN.....	117

DAFTAR TABEL

	Halaman
Tabel 4.1 Daftar Titik Ekuilibrium, Parameter Batas Keberadaan Titik Ekuilibrium, Nilai-Nilai Parameter dan Kestabilan Titik Ekuilibrium	90
Tabel 4.2 Nilai Parameter untuk Simulasi di Titik Ekuilibrium $E^*(x^*, y^*)$	102

DAFTAR GAMBAR

	Halaman
Gambar 2.1 Bidang Solusi serta Potret Fase Sistem pada Populasi <i>Predator-Prey</i> untuk $T_0(0,0)$ serta untuk $T_1(40; 33,33333333)$	16
Gambar 2.2 Bidang Solusi serta Potret Fase Sistem pada Populasi <i>Predator-Prey</i> untuk $T_0(0,0)$ serta untuk $T_1(1,0)$ dengan Syarat $\mu < \frac{2}{3}$	19
Gambar 2.3 Bidang Solusi serta Potret Fase Sistem pada Populasi <i>Predator-Prey</i> untuk $T_1(1,0)$ Kasus $\mu > \frac{2}{3}$	20
Gambar 2.4 Bidang Solusi serta Potret Fase Sistem pada Populasi <i>Predator-Prey</i> untuk $T_2(0,125; 0,546875)$ Kasus $\mu < \frac{1}{3}$	20
Gambar 2.5 Bidang Solusi serta Potret Fase Sistem pada Populasi <i>Predator-Prey</i> untuk $T_2(0,25; 0,5625)$ Kasus $\mu = \frac{1}{3}$	21
Gambar 2.6 Bidang Solusi serta Potret Fase Sistem pada Populasi <i>Predator-Prey</i> untuk $T_2(0,3333333333; 0,5555555556)$ Kasus $\mu > \frac{1}{3}$	21
Gambar 2.7 Bidang Solusi serta Potret Fase Sistem pada Populasi <i>Predator-Prey</i> untuk $E_0(0,0)$ dengan Syarat $b > \mu + q$ serta untuk $E_1(1,0)$ dengan Syarat $\frac{b}{m^2+1} < \mu + q$ dan $b > \mu + q$	24
Gambar 2.8 Bidang Solusi serta Potret Fase Sistem pada Populasi <i>Predator-Prey</i> untuk $E_1(1,0)$ dengan Syarat $b > \mu + q$ dan $\frac{b}{m^2+1} > \mu + q$	25

Gambar 2.9 Bidang Solusi serta Potret Fase Sistem pada Populasi *Predator-Prey*

untuk $E_2(0,8; 0,4)$ dengan Syarat $b > \mu + q$,

$$m \sqrt{\frac{\mu+q}{b-(\mu+q)}} < 1, p_1 > 0 \text{ dan } p_1^2 - 4p_2 > 0 \dots\dots\dots 26$$

Gambar 2.10 Bidang Solusi serta Potret Fase Sistem pada Populasi *Predator-*

Prey untuk $E_2(0,244948743; 0,5779655447)$ dengan Syarat

$$b > \mu + q, m \sqrt{\frac{\mu+q}{b-(\mu+q)}} < 1, p_1 > 0 \text{ dan } p_1^2 - 4p_2 < 0 \dots\dots\dots 27$$

Gambar 2.11 Trayektori untuk *Node Point*..... 37

Gambar 2.12 Trayektori untuk Titik *Nodal Source* 37

Gambar 2.13 Trayektori untuk *Sadle Point* 38

Gambar 2.14 Trayektori untuk *Star Point*..... 38

Gambar 2.15 Trayektori untuk *Improper Node* dengan $\lambda < 0$ 39

Gambar 2.16 Trayektori untuk *Improper Node* dengan $\lambda > 0$ 39

Gambar 2.17 Trayektori untuk *Stabel Spiral*..... 39

Gambar 2.18 Trayektori untuk *Unstable Spiral*..... 40

Gambar 2.19 Trayektori untuk *Center Point* 40

Gambar 4.1 Bidang Solusi Sistem pada Populasi *Predator-Prey* untuk

Titik Ekuilibrium $E_0(0,0)$ Kasus 1, $E_1\left(0, \frac{m}{\beta}\right)$ Kasus 1,

$E_H(x_H, 0)$ Kasus $h > c$ dengan $c - \sqrt{A} < 0$ serta Titik

$E_L(x_L, 0)$ 94

Gambar 4.2 Potret Fase Sistem pada Populasi *Predator-Prey* untuk

Titik Ekuilibrium $E_0(0,0)$ Kasus 1, $E_1\left(0, \frac{m}{\beta}\right)$ Kasus 1,

$E_H(x_H, 0)$ Kasus $h > c$ dengan $c - \sqrt{A} < 0$ serta Titik	
$E_L(x_L, 0)$	95

Gambar 4.3 Bidang Solusi Sistem pada Populasi *Predator-Prey* untuk

Titik Ekuilibrium $E_0(0,0)$ Kasus 2, $E_1\left(0, \frac{m}{\beta}\right)$ Kasus 2 serta	
$E_H(x_H, 0)$ Kasus $h < c$ dengan $c - \sqrt{A} < 0$	96

Gambar 4.4 Potret Fase Sistem pada Populasi *Predator-Prey* untuk

Titik Ekuilibrium $E_0(0,0)$ Kasus 2, $E_1\left(0, \frac{m}{\beta}\right)$ Kasus 2 serta	
$E_H(x_H, 0)$ Kasus $h < c$ dengan $c - \sqrt{A} < 0$	97

Gambar 4.5 Bidang Solusi Sistem pada Populasi *Predator-Prey* untuk

Titik Ekuilibrium $E_H(x_H, 0)$ Kasus $h < c$ dengan $c - \sqrt{A} > 0$...	98
---	----

Gambar 4.6 Potret Fase Sistem pada Populasi *Predator-Prey* untuk

Titik Ekuilibrium $E_H(x_H, 0)$ Kasus $h < c$ dengan $c - \sqrt{A} > 0$...	99
---	----

Gambar 4.7 Bidang Solusi Sistem pada Populasi *Predator-Prey* untuk

Titik Ekuilibrium $E_H(x_H, 0)$ Kasus $h > c$ dengan $c - \sqrt{A} > 0$...	100
---	-----

Gambar 4.8 Potret Fase Sistem pada Populasi *Predator-Prey* untuk

Titik Ekuilibrium $E_H(x_H, 0)$ Kasus $h > c$ dengan $c - \sqrt{A} > 0$...	101
---	-----

Gambar 4.9 Bidang Solusi Sistem pada Populasi *Predator-Prey* di

Titik Ekuilibrium $E^*(x^*, y^*)$ untuk Kasus $\frac{w_0}{w} > 1$	104
---	-----

Gambar 4.10 Potret Fase Sistem pada Populasi *Predator-Prey* di

Titik Ekuilibrium $E^*(x^*, y^*)$ untuk Kasus $\frac{w_0}{w} > 1$	105
---	-----

Gambar 4.11 Bidang Solusi Sistem pada Populasi *Predator-Prey* di

Titik Ekuilibrium $E^*(x^*, y^*)$ untuk Kasus $\frac{w_0}{w} = 1$ 106

Gambar 4.12 Potret Fase Sistem pada Populasi *Predator-Prey* di titik

Ekuilibrium $E^*(x^*, y^*)$ untuk Kasus $\frac{w_0}{w} = 1$ 107

Gambar 4.13 Bidang Solusi Sistem pada Populasi *Predator-Prey* di

Titik Ekuilibrium $E^*(x^*, y^*)$ untuk Kasus $\frac{w_0}{w} < 1$ 108

Gambar 4.14 Potret Fase Sistem pada Populasi *Predator-Prey* di titik

Ekuilibrium $E^*(x^*, y^*)$ untuk Kasus $\frac{w_0}{w} < 1$ 109

DAFTAR LAMPIRAN

	Halaman
Lampiran 1 Penondimensiolan Model	118
Lampiran 2 <i>Print Out</i> Maple 12 Dinamika Populasi <i>Predator-Prey</i>	120

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Matematika adalah cabang ilmu pengetahuan yang sangat penting dan sangat berperan dalam perkembangan dunia. Matematika dibandingkan dengan disiplin-disiplin ilmu yang lain mempunyai karakteristik tersendiri. Pentingnya matematika tidak lepas dari peranannya dalam segala jenis dimensi kehidupan. Selain itu, matematika juga seringkali dibutuhkan untuk menunjang eksistensi ilmu-ilmu lain seperti fisika, kimia, astronomi, biologi, ekonomi dan lain-lain. Tidak heran mengapa matematika dijuluki "*mathematic is a queen, but also a servant*", matematika sebagai ratu ilmu, tetapi juga sekaligus pelayan untuk ilmu-ilmu lain (Bell, 1952). Matematika dikatakan sebagai ratu ilmu karena matematika dapat tumbuh dan berkembang untuk dirinya sendiri sebagai suatu ilmu tanpa adanya bantuan dari ilmu lain. Selanjutnya matematika dikatakan sebagai pelayan ilmu lain karena ilmu lain tidak dapat tumbuh dan berkembang tanpa adanya bantuan matematika.

Di masa lalu, cabang-cabang matematika yang mempelajari fenomena fisik mendominasi cabang-cabang matematika yang bisa diterapkan pada berbagai fenomena fisik, seperti yang biasa dipelajari dalam fisika dan kimia. Akibatnya, cabang-cabang matematika ini digolongkan dalam kelompok matematika terapan atau matematika fisika. Tetapi sejak berkembangnya ilmu-ilmu komputer,

penerapan cabang-cabang matematika yang mempelajari fenomena-fenomena yang bukan sekadar diskrit, bahkan kontinu, berkembang dengan cepat khususnya berbagai fenomena alam yang teramati agar pola struktur, perubahan ruang dan sifat-sifat fenomena tersebut dapat dinyatakan dalam sebuah bentuk perumusan yang sistematis. Hasil perumusan yang menggambarkan perilaku dari proses fenomena fisik ini disebut model matematika (Widowati & Sutimin, 2007: 1). Fenomena fisik secara alamiah sebagian besar berujung pada hubungan antara kuantitas dan laju perubahan yang dalam matematika hal tersebut berkaitan dengan persamaan diferensial. Dalam mempelajari persamaan diferensial lebih lanjut, kita diperkenalkan dengan yang namanya sistem dinamik.

Sistem dinamik adalah suatu metode yang berhubungan dengan pertanyaan-pertanyaan tentang tendensi-tendensi dinamik sistem-sistem yang kompleks, yaitu pola-pola tingkah laku yang dibangkitkan oleh sistem dengan bertambahnya waktu (Tadu, 2013).

Salah satu contoh sistem yang merupakan pendekatan terhadap suatu fenomena fisik adalah sistem *predator-prey*, dimana *prey* sebagai populasi yang dimangsa dan *predator* sebagai populasi yang memangsa (Du *et al.*, 2007). Sistem *predator-prey* (pemangsa-mangsa) adalah salah satu jenis sistem yang merupakan gabungan atau interaksi dari dua populasi (kelompok individu) yaitu *predator* (pemangsa) dan *prey* (mangsa). Interaksi antar dua populasi ini sangat penting karena kelangsungan hidup makhluk hidup tergantung pada keseimbangan lingkungan disekitarnya. Keseimbangan tersebut dapat tercapai jika jumlah rata-

rata populasi dari populasi *predator* dan *prey* yang sedang berinteraksi sesuai dengan ukuran dan proporsinya.

Penelitian terhadap interaksi *predator-prey* akan dilakukan secara matematis. Penelitian dilakukan dengan menganalisa model matematika dari sistem *predator-prey*. Model tersebut dapat terdiri atas satu atau lebih persamaan atau pertidaksamaan. Persamaan itu melibatkan variabel, atau mungkin melibatkan variabel dan turunan-turunannya (persamaan diferensial), atau nilai-nilai dari variabel untuk waktu-waktu diskrit yang berbeda (Naispospos & Noeniek, 1983: 1). Model matematika dari sistem *predator-prey* ini adalah sistem persamaan diferensial (sistem dinamik)

$$\dot{x} = f(x, a), \tag{1.1}$$

dengan x adalah ruang keadaan ($x \in \mathbb{R}^n$) dan a adalah ruang parameter ($a \in \mathbb{R}^n$).

Model *predator-prey* pertama kali dikenalkan oleh Lotka pada tahun 1925 dan Volterra pada tahun 1926, sehingga model ini juga disebut model Lotka-Volterra (Boyce dan DiPrima, 2009). Model sederhana ini kemudian mengalami banyak modifikasi. Salah satu modifikasi dilakukan oleh Leslie dan Gower dengan memperkenalkan model *predator-prey* dimana *carrying capacity* lingkungan *predator* sebanding dengan banyak *prey* (Leslie, 1948; Leslie & Gower, 1960). Kemudian pada tahun 1950 Holling memperkenalkan fungsi respon. Fungsi respon dalam ekologi adalah banyak makanan yang dimakan oleh *predator* sebagai fungsi kepadatan makanan (Hunsicker *et al*, 2011). Dalam hal ini fungsi respon dibagi atas tiga macam, yaitu fungsi respon tipe I, tipe II dan tipe

III. Fungsi respon tipe I terjadi pada *predator* yang memiliki karakteristik pasif atau lebih suka menunggu *prey*-nya. Fungsi respon tipe II terjadi pada *predator* yang berkarakteristik aktif dalam mencari *prey*. Fungsi respon tipe III terjadi pada *predator* yang cenderung akan mencari populasi *prey* yang lain ketika populasi *prey* yang dimakan mulai berkurang. Pada sistem Lotka-Volterra mengabaikan fungsi respon ini. Pada sistem (1.1) fungsi respon muncul sebagai suatu yang dapat mengurangi jumlah *prey* dan menambah jumlah *predator*.

Secara umum sifat solusi dari sistem (1.1) dapat dilihat berdasarkan nilai eigen atas matriks terkait sistem dinamikanya. Ada suatu teorema yang menyatakan bahwa *solusi dari suatu sistem stabil jika semua nilai eigen mempunyai bagian real negatif dan solusi sistem tidak stabil jika ada nilai eigen yang mempunyai bagian real positif* (Perko, 1991: 56). Teorema tersebut memunculkan suatu pertanyaan, *bagaimana jika ada nilai eigen 0 pada bagian realnya?* Ketika nilai eigen 0 pada bagian realnya merupakan suatu kondisi dimana sistem rentan terhadap gangguan. Sedikit saja sistem terganggu maka nilai eigen dapat berpindah ke daerah negatif (stabil) atau sebaliknya berpindah ke daerah positif (tidak stabil). Hal inilah yang menyebabkan terjadinya bifurkasi, yaitu perbedaan keadaan dinamik seiring dengan perubahan parameter (Wiggins, 1990: 254). Tetapi tidak semua sistem dinamik yang memiliki nilai eigen 0 pada bagian realnya akan mengalami bifurkasi.

Pada penelitian ini akan dibahas tentang analisis kestabilan pada modifikasi model *predator-prey* Leslie Gower dengan fungsi respon Holling tipe II berparameter w (tingkat pertumbuhan populasi *predator*). Dipilihnya fungsi

respon Holling tipe II karena model sistem *predator-prey* dengan fungsi respon Holling tipe II memiliki nilai eigen imajiner murni, sehingga dimungkinkan munculnya bifurkasi. Jenis bifurkasi yang mungkin terjadi untuk sistem dengan nilai eigen imajiner murni adalah bifurkasi Hopf.

Sistem *predator-prey* dengan fungsi respon Holling-tipe II dan laju pemanenan konstan telah dipelajari secara detail oleh Peng *et al* (2009). Zhu dan Lan (2011) mempelajari potret fase di sekitar titik kesetimbangan dalam pada model Leslie Gower dengan laju pemanenan konstan pada *prey*. Song dan Li (2008) mengusulkan dan menganalisa model *predator-prey* periodik dengan modifikasi Leslie-Gower dan fungsi respon Holling-tipe II serta efek impulsif untuk perilaku dinamikanya. Hsu dan Huang (1999) mengusulkan sistem *predator-prey* Leslie dan Holling untuk analisis bifurkasinya. Ma (2012) mempelajari stabilitas global untuk model *predator-prey* yang dimodifikasi oleh Leslie-Gower dengan waktu tunda. Lin dan Ho (2006) mempelajari stabilitas lokal dan global untuk model *predator-prey* yang dimodifikasi oleh Leslie-Gower dan Holling-tipe II dengan waktu tunda.

Berdasarkan uraian latar belakang di atas, maka penulis tertarik untuk menganalisis perilaku sistem *predator-prey* Leslie Gower dengan fungsi respon Holling tipe II ditinjau dari terjadinya bifurkasi satu parameter. Sebuah tulisan dalam bentuk skripsi yang diformulasikan dengan judul “**BIFURKASI HOPF PADA MODIFIKASI MODEL *PREDATOR-PREY* LESLIE GOWER DENGAN FUNGSI RESPON HOLLING TIPE II**”.

1.2 Rumusan Masalah

- (1) Bagaimana menurunkan model matematika dari modifikasi sistem *predator-prey* Leslie Gower dengan fungsi respon Holling tipe II?
- (2) Bagaimana menentukan titik kesetimbangan serta analisis kestabilan pada modifikasi model *predator-prey* Leslie Gower dengan fungsi respon Holling tipe II?
- (3) Bagaimana pengaruh perubahan parameter terhadap keadaan yang sebenarnya dari modifikasi model *predator-prey* Leslie Gower dengan fungsi respon Holling tipe II?
- (4) Bagaimana simulasi numerik dari modifikasi model *predator-prey* Leslie Gower dengan fungsi respon Holling tipe II dengan menggunakan *software* Maple?

1.3 Batasan Masalah

Agar pembahasan dalam penelitian ini tidak meluas, maka penulis perlu memberikan batasan-batasan sebagai berikut.

- (1) Sistem dinamik yang digunakan untuk memodelkan sistem *predator-prey* disini adalah sistem dinamik dimensi 2 dengan 1 parameter.
- (2) Analisis kestabilan terjadinya bifurkasi hanya dilakukan pada satu titik ekuilibrium.

1.4 Tujuan

Tujuan penulisan skripsi ini sebagai berikut.

- (1) Untuk menurunkan model matematika dari modifikasi sistem *predator-prey* Leslie Gower dengan fungsi respon Holling tipe II.

- (2) Untuk menentukan titik kesetimbangan serta analisis kestabilan pada modifikasi model predator-prey Leslie Gower dengan fungsi respon Holling tipe II.
- (3) Untuk menganalisis pengaruh perubahan parameter terhadap keadaan dinamik dari modifikasi model *predator-prey* Leslie Gower dengan fungsi respon Holling tipe II.
- (4) Untuk membuat simulasi numerik dari modifikasi model *predator-prey* Leslie Gower dan fungsi respon Holling tipe II dengan menggunakan *software* Maple.

1.5 Manfaat

Manfaat yang diharapkan dari penyusunan skripsi ini sebagai berikut.

(1) Bagi Peneliti

Memberikan informasi kepada peneliti tentang sifat-sifat atau perilaku dari modifikasi sistem *predator-prey* Leslie Gower dengan fungsi respon Holling tipe II ditinjau dari terjadinya bifurkasi satu parameter.

(2) Bagi Mahasiswa

Menambah wawasan dan kemampuan dalam mengaplikasikan ilmu-ilmu matematika, dalam bidang biologi yaitu tentang sifat-sifat atau perilaku dari modifikasi sistem *predator-prey* Leslie Gower dengan fungsi respon Holling tipe II ditinjau dari terjadinya bifurkasi satu parameter.

(3) Bagi Universitas

Menambah koleksi buku referensi yang ada di Perpustakaan Universitas Negeri Semarang.

BAB 2

LANDASAN TEORI

2.1 Sistem Persamaan Diferensial

Sistem persamaan diferensial adalah suatu sistem yang memuat n buah persamaan diferensial, dengan n buah fungsi yang tidak diketahui, dimana n merupakan bilangan bulat positif lebih besar sama dengan 2 (Finizio & Ladas, 1998). Antara persamaan diferensial yang satu dengan yang lain saling keterkaitan dan konsisten. Sistem persamaan Lotka-Volterra merupakan salah satu contoh sistem persamaan diferensial dalam ekologi. Sistem persamaan tersebut mempunyai bentuk sebagai berikut.

$$\begin{aligned}\frac{dH}{dt} &= aH - bHP, \\ \frac{dP}{dt} &= -dP + cHP,\end{aligned}\tag{2.1}$$

dengan

$H(t)$: kepadatan populasi *prey* pada waktu t ($H(t) > 0$).

$P(t)$: kepadatan populasi *predator* pada waktu t ($P(t) > 0$).

a : koefisien laju pertumbuhan intrinsik populasi *prey* ($a > 0$).

b : angka penurunan kepadatan populasi *prey* karena terjadinya interaksi antara *prey* dan *predator* ($b > 0$).

d : koefisien laju kematian alami *predator* ($d > 0$).

c : angka pertumbuhan kepadatan populasi *predator* karena terjadinya

interaksi antara *prey* dan *predator* ($c > 0$).

(Rahadian, 2009: 27).

2.2 Model Pertumbuhan Logistik

Model ini merupakan penyempurnaan dari model eksponensial. Model eksponensial mempunyai kelemahan yaitu saat nilai $R_1 > 0$, maka populasi tumbuh sampai tak terbatas, dengan R_1 merupakan laju pertumbuhan populasi. Hal ini mustahil, sehingga perlu adanya kajian lagi. Model ini diberikan dengan memberikan asumsi bahwa $R_1 = R(N)$. Nilai $R(N)$ ditentukan oleh kelahiran dan pengaruh kepadatan populasi (keterbatasan luas lingkungan). Nilai $R(N)$ dapat dirumuskan dengan

$$R(N) = a - bN, \quad (2.2)$$

dimana a menyatakan laju pertumbuhan populasi tanpa pengaruh lingkungan dan b menyatakan pengaruh dari penambahan kepadatan populasi (semakin padat populasi maka persaingan antar individu meningkat). Model pertumbuhan logistik dirumuskan sebagai berikut.

$$\frac{dN}{dt} = N(a - bN). \quad (2.3)$$

Titik ekuilibrium dari model tersebut diperoleh dengan menyelesaikan persamaan $N(a - bN) = 0$ diperoleh nilai N yang memenuhi adalah $N = 0$ dan $N = \frac{a}{b}$. Populasi nol pasti menjadi titik ekuilibrium tetapi yang menarik adalah $N = \frac{a}{b}$. Ini adalah populasi terbesar dimana lingkungan masih mendukung populasi tanpa adanya kehilangan individu anggota populasi (individu mati). Nilai

ini disebut *carrying capacity* dari lingkungan (habitat). Teori ini memprediksi bahwa populasi $N = \frac{a}{b}$ berkaitan dengan Z.P.G (*Zero Population Growth*).

Solusi persamaan logistik di atas apabila diketahui nilai awal $N(0) = N_0$ adalah sebagai berikut.

$$\frac{dN}{dt} = N(a - bN) \Leftrightarrow \frac{dN}{N(a-bN)} = dt. \quad (2.4)$$

Dengan metode integral fungsi rasional dalam kalkulus diperoleh bahwa

$$\frac{1}{N(a-bN)} = \frac{A}{N} + \frac{B}{a-bN}$$

$$\Leftrightarrow \frac{1}{N(a-bN)} = \frac{Aa+(B-Ab)N}{N(a-bN)}$$

$$\Leftrightarrow 1 = Aa + (B - Ab)N$$

$$\Leftrightarrow Aa = 1 \text{ dan } (B - Ab)N = 0$$

$$Aa = 1 \Leftrightarrow A = \frac{1}{a} \text{ dan } B - Ab = 0 \Leftrightarrow B - \frac{b}{a} = 0 \Leftrightarrow B = \frac{b}{a}.$$

Dari hasil di atas diperoleh $\frac{1}{N(a-bN)} = \frac{1}{a} + \frac{b}{a-bN}$.

Apabila disubstitusikan ke persamaan diferensial (2.4) diperoleh

$$\frac{dN}{N(a-bN)} = dt \Leftrightarrow \frac{1}{a} dN + \frac{b}{a-bN} dN = dt.$$

Dengan melakukan pengintegralan kedua ruas diperoleh

$$\int \frac{1}{a} dN + \int \frac{b}{a-bN} dN = \int dt$$

$$\Leftrightarrow \frac{1}{a} \int \frac{1}{N} dN + \frac{b}{a} \left(-\frac{1}{b}\right) \int \frac{1}{a-bN} d(a-bN) = \int dt$$

$$\Leftrightarrow \frac{1}{a} \ln|N| - \frac{1}{a} \ln|a-bN| = t + c.$$

Dengan substitusi nilai $N(0) = N_0$ diperoleh

$$\begin{aligned} \frac{1}{a} \ln|N| - \frac{1}{a} \ln|a - bN| &= t + \frac{1}{a} \ln|N_0| - \frac{1}{a} \ln|a - bN_0| \\ \Leftrightarrow \frac{1}{a} \ln|N| - \frac{1}{a} \ln|a - bN| - \frac{1}{a} \ln|N_0| + \frac{1}{a} \ln|a - bN_0| &= t \\ \Leftrightarrow \frac{1}{a} (\ln|N| - \ln|N_0|) + \frac{1}{a} (\ln|a - bN_0| - \ln|a - bN|) &= t. \end{aligned}$$

Karena N dan N_0 keduanya positif maka diperoleh

$$\begin{aligned} \frac{1}{a} \ln \frac{N}{N_0} + \frac{1}{a} \ln \left| \frac{a - bN_0}{a - bN} \right| &= t \\ \Leftrightarrow \frac{1}{a} \left(\ln \frac{N}{N_0} + \ln \left| \frac{a - bN_0}{a - bN} \right| \right) &= t \\ \Leftrightarrow \ln \frac{N}{N_0} + \ln \left| \frac{a - bN_0}{a - bN} \right| &= at \\ \Leftrightarrow \ln \frac{N}{N_0} \left| \frac{a - bN_0}{a - bN} \right| &= at \\ \Leftrightarrow \frac{N}{N_0} \left| \frac{a - bN_0}{a - bN} \right| &= e^{at}. \end{aligned}$$

Karena $a - bN_0$ dan $a - bN$ mempunyai tanda yang sama maka persamaan di atas menjadi

$$\begin{aligned} \frac{N}{N_0} \left(\frac{a - bN_0}{a - bN} \right) &= e^{at} \\ \Leftrightarrow N(a - bN_0) &= (a - bN)N_0 e^{at}. \end{aligned}$$

Apabila dilanjutkan dengan beberapa tahap perhitungan lagi diperoleh

$$\begin{aligned} N(a - bN_0) &= aN_0 e^{at} - bNN_0 e^{at} \\ \Leftrightarrow N(a - bN_0 + bN_0 e^{at}) &= aN_0 e^{at} \\ \Leftrightarrow N &= \frac{aN_0 e^{at}}{a - bN_0 + bN_0 e^{at}} \\ \Leftrightarrow N &= \frac{aN_0 e^{at}}{a - bN_0 + bN_0 e^{at}} \times \frac{\frac{1}{bN_0 e^{at}}}{\frac{1}{bN_0 e^{at}}} \\ \Leftrightarrow N &= \frac{\frac{a}{b}}{\frac{a}{bN_0 e^{at}} - \frac{1}{e^{at}} + 1} \end{aligned}$$

$$\Leftrightarrow N = \frac{\frac{a}{b}}{\frac{a}{bN_0}e^{-at} - e^{-at} + 1}$$

$$\Leftrightarrow N = \frac{\frac{a}{b}}{\left(\frac{a}{bN_0} - 1\right)e^{-at} + 1}$$

$$\Leftrightarrow N = \frac{\frac{a}{b}}{1 + \left(\frac{a - bN_0}{bN_0}\right)e^{-at}}$$

Untuk nilai $t \rightarrow \infty$ diperoleh nilai $N \rightarrow \frac{a}{b}$ yang mempengaruhi *carrying capacity* dari lingkungan (Haberman, 1997).

2.3 Model Populasi *Predator-Prey*

Laju populasi *prey* dengan tidak adanya *predator* tumbuh cepat mendekati eksponensial dan tak terbatas dalam bentuk sebagai berikut.

$$\frac{dN(t)}{dt} = N(t)r, \quad (2.5)$$

dengan

$N(t)$: kepadatan populasi *prey* pada waktu t .

r : koefisien laju pertumbuhan intrinsik populasi *prey*.

Laju populasi *prey* menjadi fungsi logistik karena sumber daya alam yang terbatas, yang kemudian dapat menulisnya sebagaimana persamaan logistik sebelumnya yaitu sebagai berikut.

$$\frac{dN(t)}{dt} = N(t)r \left(1 - \frac{N(t)}{K}\right), \quad (2.6)$$

dengan proporsi dan banyaknya individu dalam populasi yang belum digunakan $\left(1 - \frac{N(t)}{K}\right)$ dan K (*carrying capacity*) adalah jumlah maksimum banyaknya individu dalam suatu populasi. Populasi pada tingkat K kadang juga disebut

tingkat kejenuhan, karena untuk populasi besar lebih banyak kematian daripada kelahiran.

Carrying capacity atau daya dukung adalah jumlah maksimum individu yang dapat didukung atau dilayani oleh sumber daya yang ada didalam suatu ekosistem. Dengan kata lain, *carrying capacity* dapat disebut juga sebagai kemampuan lingkungan (ekosistem) dalam mendukung kehidupan semua makhluk yang ada didalamnya secara berkelanjutan. Dalam hal ini, *carrying capacity* berhubungan erat dengan ketersediaan tanaman sebagai makanan *prey*. Kemudian ditunjukkan suatu persamaan dimana *prey* dan *predator* akan saling berinteraksi yaitu sebagai berikut.

$$\frac{dN(t)}{dt} = -\beta N(t)P(t), \quad (2.7)$$

dengan β adalah laju penangkapan *prey* oleh *predator* dan $P(t)$ adalah populasi *predator*. Dalam hal ini *prey* berinteraksi dengan *predator*. Dari beberapa penjelasan di atas maka dapat dibentuk model dinamika pertumbuhan populasi *prey* adalah sebagai berikut.

$$\frac{dN(t)}{dt} = N(t)r \left(1 - \frac{N(t)}{K}\right) - \beta N(t)P(t). \quad (2.8)$$

Dalam hal ini diasumsikan $K, \beta, r > 0$, yaitu mengingat setiap populasi memiliki potensi untuk berkembang biak.

Pada persamaan di atas bersifat mengurangi jumlah populasi *prey*. Karena dalam hubungannya *prey* akan berinteraksi dengan *predator*. Akan tetapi sebaliknya pada model pertumbuhan *predator* maka respon ini akan bersifat menambah jumlah *predator* (Timuneno *et al.*, 2008).

2.4 Model Leslie Gower

Leslie memperkenalkan model *predator-prey* berikut, dimana “*carrying capacity*” lingkungan *predator* sebanding dengan jumlah *prey* (Leslie, 1948; Leslie & Gower):

$$\begin{aligned}\frac{dH}{dt} &= H(r_1 - a_1H - a_2P), \\ \frac{dP}{dt} &= r_2P \left(1 - \frac{a_3P}{H}\right),\end{aligned}\tag{2.9}$$

dengan

$H(t)$: kepadatan populasi *prey* pada waktu t ($H(t) > 0$).

$P(t)$: kepadatan populasi *predator* pada waktu t ($P(t) > 0$).

r_1 : koefisien laju pertumbuhan intrinsik populasi *prey* ($r_1 > 0$).

r_2 : koefisien laju pertumbuhan populasi *predator* ($r_2 > 0$).

a_1 : angka penurunan kepadatan populasi *prey* karena terjadinya interaksi antar *prey* ($a_1 > 0$).

a_2 : angka penurunan kepadatan populasi *prey* karena terjadinya interaksi antara *prey* dan *predator* ($a_2 > 0$).

a_3 : angka penurunan kepadatan populasi *predator* karena terjadinya interaksi antar *predator* ($a_3 > 0$).

2.5 Fungsi Respon

Fungsi respon dalam ekologi adalah jumlah makanan yang dimakan oleh *predator* sebagai fungsi kepadatan makanan. Dalam hal ini fungsi respon dibagi atas tiga macam, yaitu fungsi respon Holling tipe I, tipe II dan tipe III.

2.5.1 Fungsi Respon Holling tipe I

Fungsi respon Holling tipe I merupakan hubungan antara kepadatan populasi *prey* dan tingkat konsumsi (Altwegg, 2006). Tingkat konsumsi *predator* meningkat linear dengan kepadatan *prey*, tetapi akan konstan ketika *predator* berhenti memangsa. Fungsi respon Holling tipe I terjadi pada *predator* yang memiliki karakteristik pasif, atau lebih suka menunggu *prey*-nya, sebagai contoh *predator*-nya adalah laba-laba. Adapun tingkat pertumbuhan *prey* pada fungsi respon Holling tipe I diberikan sebagai berikut (Tsai & Pao, 2004).

$$F^I(N) = mN, \quad (2.10)$$

dengan

F^I : fungsi respon Holling tipe I.

m : tingkat konsumsi maksimum *predator* terhadap *prey*.

N : kepadatan populasi *prey*.

Contoh 2.1 (Rahadian, 2009: 27)

Diberikan sistem persamaan diferensial sebagai berikut.

$$\begin{aligned} \frac{dH}{dt} &= aH - bHP, \\ \frac{dP}{dt} &= -dP + cHP, \end{aligned} \quad (2.11)$$

dengan

$H(t)$: kepadatan populasi *prey* pada waktu t ($H(t) > 0$).

$P(t)$: kepadatan populasi *predator* pada waktu t ($P(t) > 0$).

a : koefisien laju pertumbuhan intrinsik populasi *prey* ($a > 0$).

b : angka penurunan kepadatan populasi *prey* karena terjadinya interaksi antara *prey* dan *predator* ($b > 0$).

d : koefisien laju kematian alami *predator* ($d > 0$).

c : angka pertumbuhan kepadatan populasi *predator* karena terjadinya interaksi antara *prey* dan *predator* ($c > 0$).

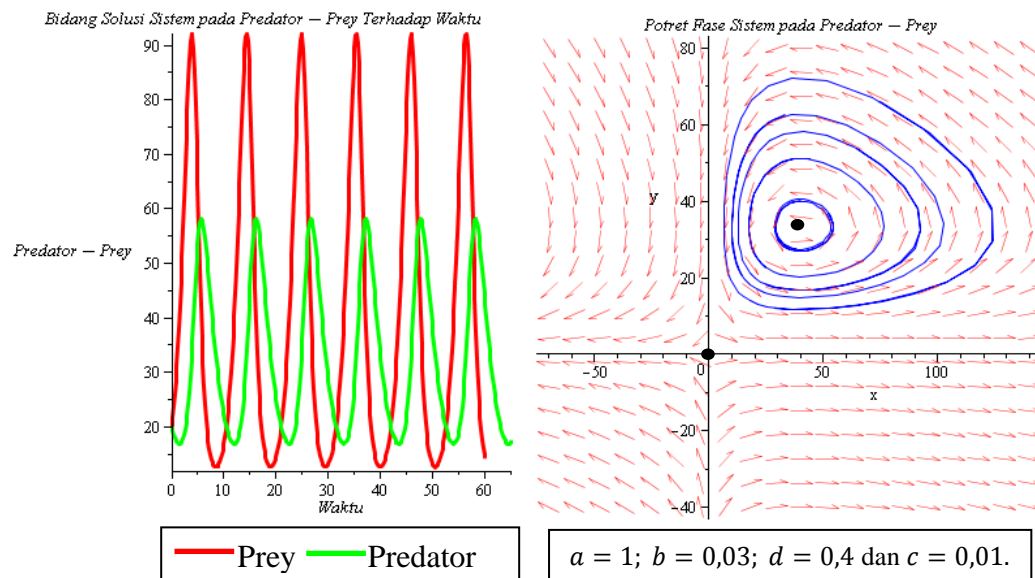
Sistem (2.11) memiliki dua titik ekuilibrium, yaitu $T_0(0,0)$ dan $T_1\left(\frac{d}{c}, \frac{a}{b}\right)$

dengan matriks Jacobian $J = \begin{pmatrix} a - bP & -bH \\ cP & -d + cH \end{pmatrix}$. Dari matriks Jacobian sistem

(2.11) yang dihitung pada setiap titik kesetimbangan menghasilkan nilai eigen

$\lambda_1 = a$ dan $\lambda_2 = -d$ untuk $T_0(0,0)$ serta $\lambda_{1,2} = \pm i\sqrt{ad}$ untuk $T_1\left(\frac{d}{c}, \frac{a}{b}\right)$.

Hasil simulasi dapat dilihat pada Gambar 2.1.



Gambar 2.1 Bidang solusi serta potret fase sistem pada populasi *predator-prey* untuk $T_0(0,0)$ menyebabkan kestabilan $T_0(0,0)$ bersifat pelana tidak stabil serta untuk $T_1(40; 33,33333333)$ menyebabkan *center* pada titik T_1

2.5.2 Fungsi Respon Holling Tipe II

Fungsi respon Holling tipe II menggambarkan rata-rata tingkat konsumsi dari *predator*, ketika *predator* menghabiskan waktu untuk mengkonsumsi *prey*. Fungsi respon tipe II terjadi pada *predator* yang berkarakteristik aktif dalam mencari *prey*, sebagai contoh *predator*-nya adalah serigala. Pada model Holling tipe II, saat kepadatan populasi mangsa rendah, model tersebut menggambarkan tingkat konsumsi sebagai fungsi naik dari populasi *prey*, sampai pada kepadatan *prey* yang tinggi dimana tingkat konsumsi mencapai titik kejenuhan (*half saturation*). Dalam hal ini, tingkat pertumbuhan *prey* pada fungsi respon Holling tipe II diberikan sebagai berikut (Skalski & Gilliam, 2001).

$$F^{(II)}(N) = \frac{mN}{1+bN} \quad (2.12)$$

dengan

$F^{(II)}$: fungsi respon Holling tipe II.

m : tingkat konsumsi maksimum *predator* terhadap *prey*.

b : waktu *predator* mengkonsumsi *prey*.

N : kepadatan populasi *prey*.

Contoh 2.2 (Abadi *et al.*, 2013)

Sebagai contoh, sistem berikut adalah dua persamaan diferensial yang tergantung pada satu parameter.

$$\begin{aligned} x' &= x(1-x) - \frac{2xy}{1+2x} \\ y' &= y\left(\frac{2x}{1+2x} - \mu\right), \end{aligned} \quad (2.13)$$

dengan

x : kepadatan populasi *prey* ($x > 0$).

y : kepadatan populasi *predator* ($y > 0$).

μ : laju kematian alami *predator* ($\mu > 0$).

Sistem (2.13) memiliki tiga titik ekuilibrium, yaitu $T_0(0,0)$, $T_1(1,0)$ dan

$T_2\left(\frac{\mu}{2(1-\mu)}, \frac{2-3\mu}{4(1-\mu)^2}\right)$ untuk semua μ dengan matriks Jacobian

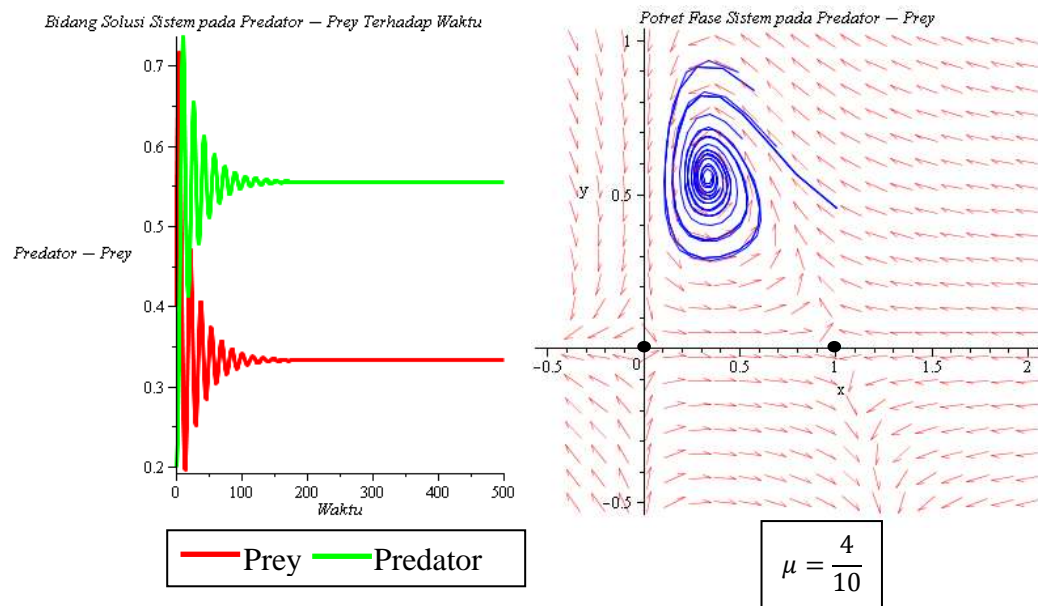
$$J = \begin{pmatrix} 1 - 2x - \frac{2y}{(1+2x)^2} & -\frac{2x}{1+2x} \\ \frac{2y}{(1+2x)^2} & \frac{2x}{1+2x} - \mu \end{pmatrix}. \text{ Dari matriks Jacobian sistem (2.13) yang}$$

dihitung pada setiap titik kesetimbangan menghasilkan nilai eigen $\lambda_1 = 1$ dan

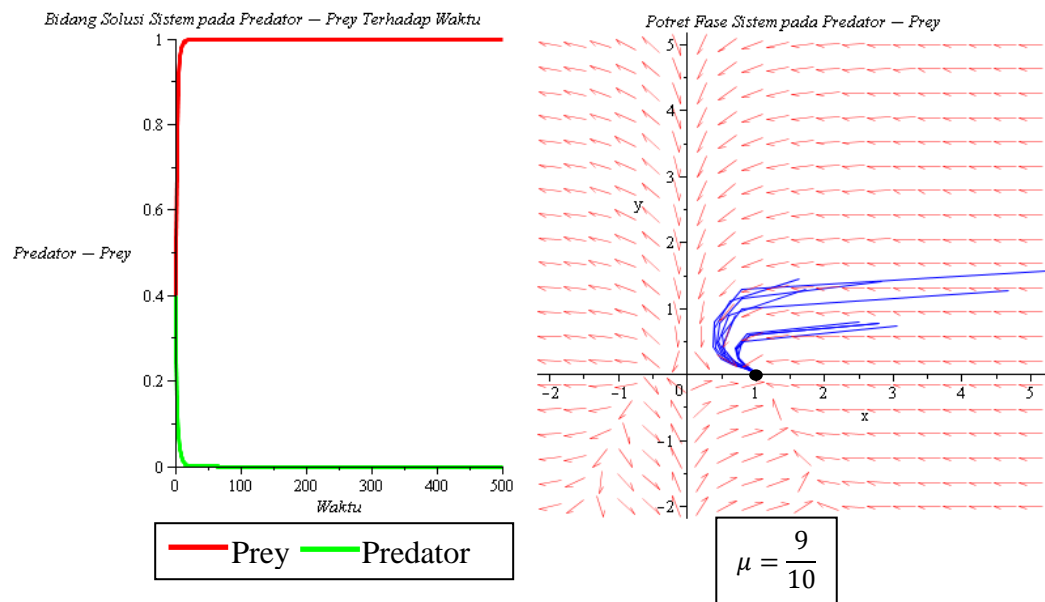
$\lambda_2 = -\mu$ untuk $T_0(0,0)$, $\lambda_1 = -1$ dan $\lambda_2 = \frac{2}{3} - \mu$ untuk $T_1(1,0)$ serta $\lambda_{1,2} =$

$$\frac{\mu(1-3\mu)}{4(1-\mu)} \pm \frac{1}{2} \sqrt{\left(\frac{\mu(1-3\mu)}{2(1-\mu)}\right)^2 - 4\mu \left(\frac{2-3\mu}{2}\right)} \text{ untuk } T_2\left(\frac{\mu}{2(1-\mu)}, \frac{2-3\mu}{4(1-\mu)^2}\right).$$

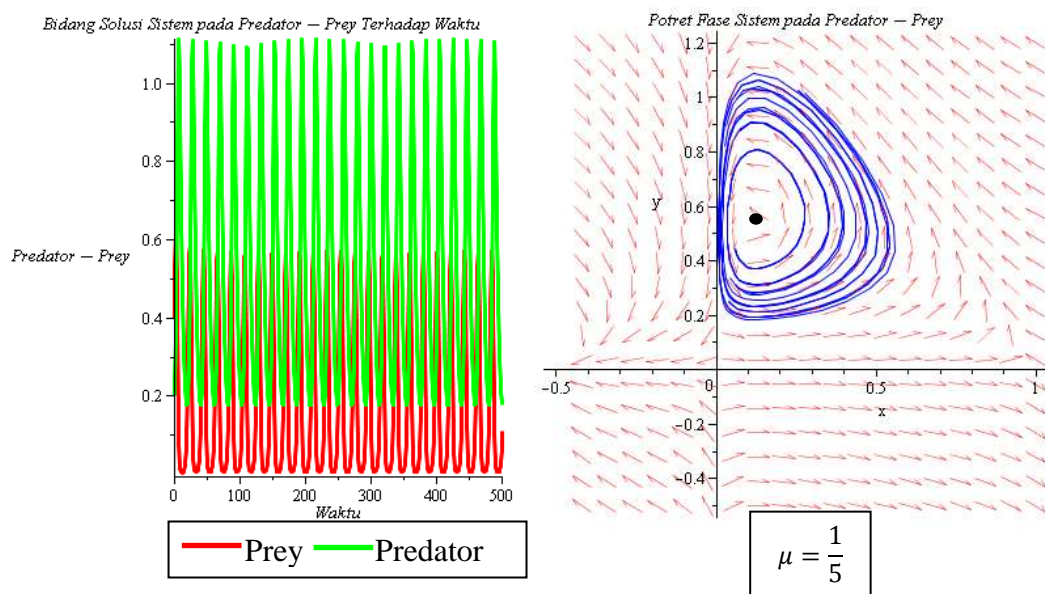
Hasil simulasi dapat dilihat pada Gambar 2.2, 2.3, 2.4, 2.5 dan 2.6.



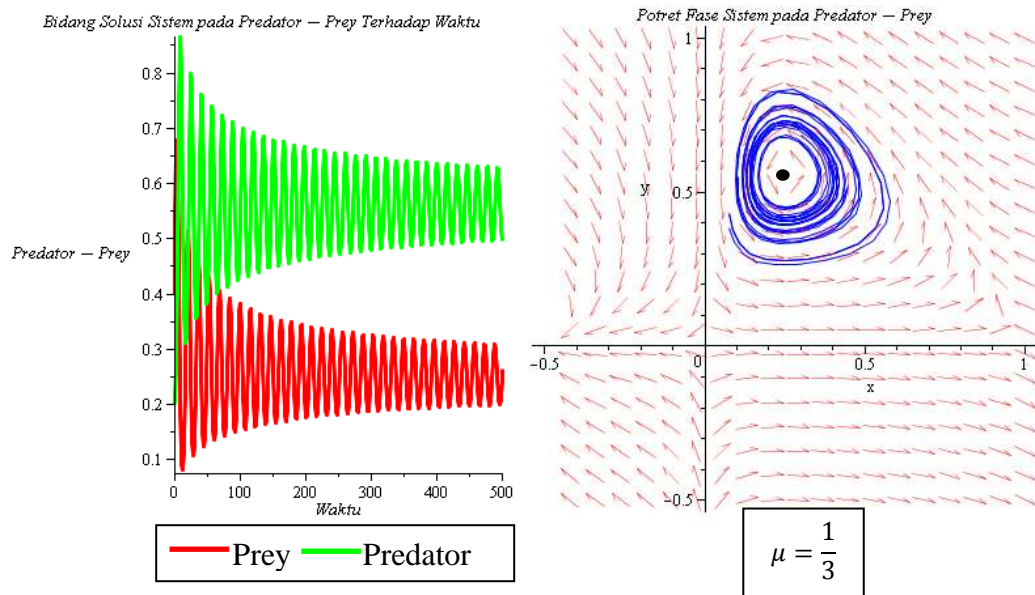
Gambar 2.2 Bidang solusi serta potret fase sistem pada populasi *predator-prey* untuk $T_0(0,0)$ menyebabkan kestabilan $T_0(0,0)$ bersifat pelana tidak stabil serta untuk $T_1(1,0)$ dengan syarat $\mu < \frac{2}{3}$ menyebabkan kestabilan $T_1(1,0)$ bersifat pelana tidak stabil



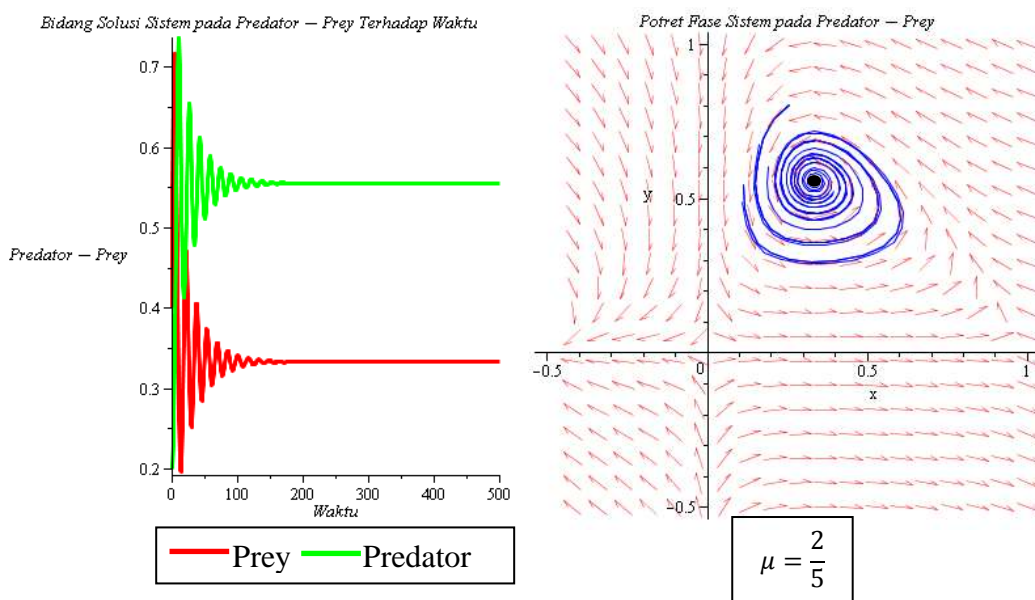
Gambar 2.3 Bidang solusi serta potret fase sistem pada populasi *predator-prey* untuk $T_1(1,0)$ dengan syarat $\mu > \frac{2}{3}$ menyebabkan kestabilan $T_1(1,0)$ bersifat simpul stabil



Gambar 2.4 Bidang solusi serta potret fase sistem pada populasi *predator-prey* untuk $T_2(0,125; 0,546875)$ kasus $\mu < \frac{1}{3}$ menyebabkan kestabilan T_2 bersifat spiral tidak stabil



Gambar 2.5 Bidang solusi serta potret fase sistem pada populasi *predator-prey* untuk $T_2(0,25; 0,5625)$ kasus $\mu = \frac{1}{3}$ menyebabkan munculnya *limit cycle*



Gambar 2.6 Bidang solusi serta potret fase sistem pada populasi *predator-prey* untuk $T_2(0,3333333333; 0,5555555556)$ kasus $\mu > \frac{1}{3}$ menyebabkan kestabilan T_2 bersifat spiral stabil

2.5.3 Fungsi Respon Holling Tipe III

Fungsi respon Holling tipe III juga menggambarkan tingkat pertumbuhan *predator*. Tetapi pada tipe ini dapat terlihat mengenai penurunan tingkat pemangsaan pada saat kepadatan *prey* rendah. Hal tersebut tidak dapat terlihat pada fungsi respon Holling tipe II. Fungsi respon Holling tipe III terjadi pada *predator* yang cenderung akan mencari populasi *prey* yang lain ketika populasi *prey* yang dimakan mulai berkurang. Karena *predator* yang cenderung akan mencari populasi *prey* yang lain, maka tingkat pertemuan antara *predator* dan *prey* adalah dua. Hal inilah yang menyebabkan variabel populasi *prey* menjadi N^2 , sehingga laju populasi menjadi lebih cepat. Adapun tingkat pertumbuhan *prey* pada fungsi respon Holling tipe III diberikan sebagai berikut (Ndam & Kassem, 2009).

$$F^{(III)}(N) = \frac{mN^2}{a^2 + N^2}, \quad (2.14)$$

dengan

$F^{(III)}$: fungsi respon Holling tipe III.

m : tingkat konsumsi maksimum *predator* terhadap *prey*.

a : tingkat kejenuhan pemangsaan.

N : kepadatan populasi *prey*.

Contoh 2.3 (Wijayanti, 2014)

Sebagai contoh, sistem berikut adalah dua persamaan diferensial yang tergantung pada satu parameter.

$$\frac{dx}{dt} = x(1 - x) - \frac{ax^2y}{m^2 + x^2},$$

$$\frac{dy}{dt} = \frac{bx^2y}{m^2+x^2} - (\mu + q)y, \quad (2.15)$$

dengan

x : kepadatan populasi *prey* ($x > 0$).

y : kepadatan populasi *predator* ($y > 0$).

a : angka penurunan kepadatan populasi *prey* karena terjadinya interaksi antara *prey* dan *predator* ($a > 0$).

b : angka pertumbuhan kepadatan populasi *predator* karena terjadinya interaksi antara *prey* dan *predator* ($b > 0$).

m : tingkat kejenuhan pemangsaan ($m > 0$).

q : koefisien laju pemanenan *prey* ($q > 0$).

μ : koefisien laju kematian alami *predator* ($\mu > 0$).

Sistem (2.15) memiliki tiga titik ekuilibrium, yaitu $E_0(0,0)$, $E_1(1,0)$ dan

$$E_2 \left(m \sqrt{\frac{\mu+q}{b-(\mu+q)}}, m \frac{(1-m \sqrt{\frac{\mu+q}{b-(\mu+q)}})(1+\frac{\mu+q}{b-(\mu+q)})}{a \sqrt{\frac{\mu+q}{b-(\mu+q)}}} \right) \text{ untuk semua } \mu \text{ dengan matriks}$$

$$\text{Jacobian } J = \begin{pmatrix} 1 - 2x - \frac{2am^2xy}{(m^2+x^2)^2} & -\frac{ax^2}{m^2+x^2} \\ \frac{2bm^2xy}{(m^2+x^2)^2} & \frac{bx^2}{m^2+x^2} - \mu - q \end{pmatrix}. \text{ Dari matriks Jacobian}$$

sistem (2.15) yang dihitung pada setiap titik kesetimbangan menghasilkan nilai

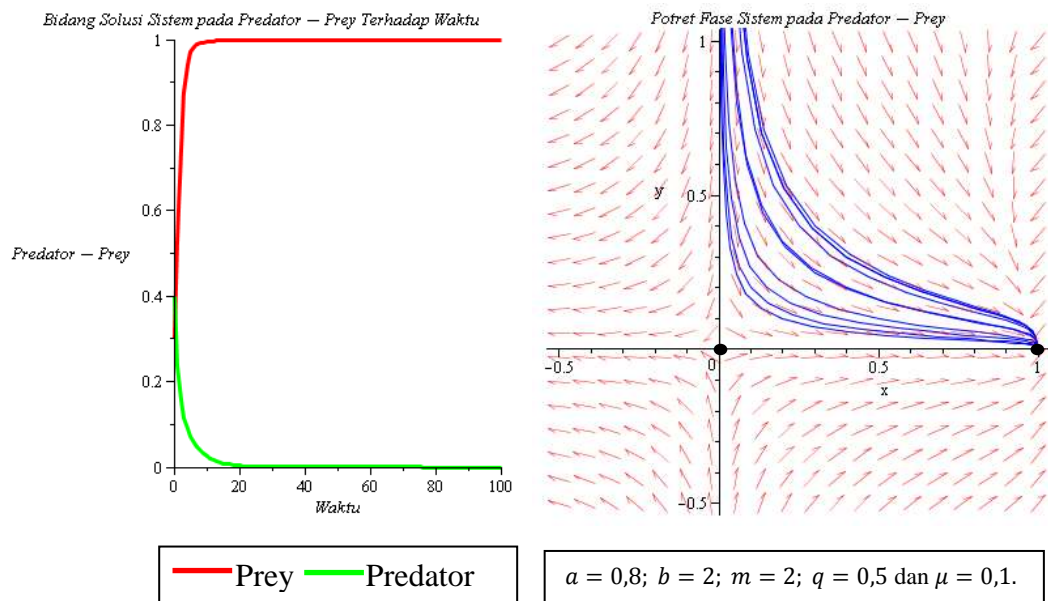
eigen $\lambda_1 = 1$ dan $\lambda_2 = -\mu - q = -(\mu + q)$ untuk $E_0(0,0)$, $\lambda_1 = -1$ dan

$\lambda_2 = \frac{b}{m^2+1} - \mu - q$ untuk $E_1(1,0)$ serta $\lambda_{1,2} = \frac{-p_1 \pm \sqrt{p_1^2 - 4p_2}}{2}$, dengan $p_1 =$

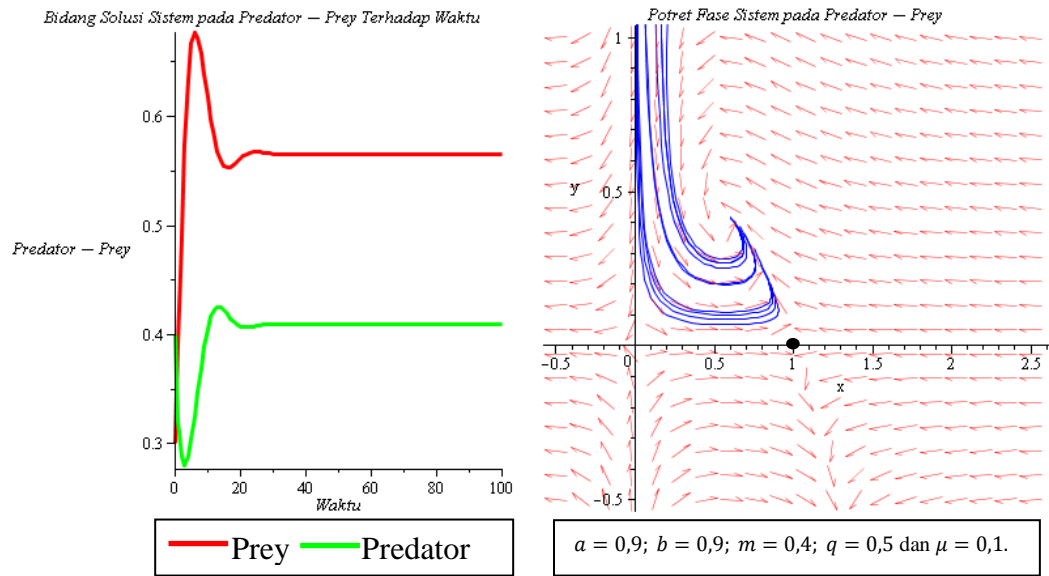
$$2(m\sqrt{A} - 1) \left(1 - \frac{1}{1+A}\right) + 1, p_2 = \frac{2bA(1-m\sqrt{A})}{(1+A)^2} \quad \text{dan} \quad A = \frac{\mu+q}{b-(\mu+q)} \quad \text{untuk}$$

$$E_2 \left(m \sqrt{\frac{\mu+q}{b-(\mu+q)}}, m \frac{\left(1-m \sqrt{\frac{\mu+q}{b-(\mu+q)}}\right) \left(1+\frac{\mu+q}{b-(\mu+q)}\right)}{a \sqrt{\frac{\mu+q}{b-(\mu+q)}}} \right).$$

Hasil simulasi dapat dilihat pada Gambar 2.7, 2.8, 2.9 dan 2.10.



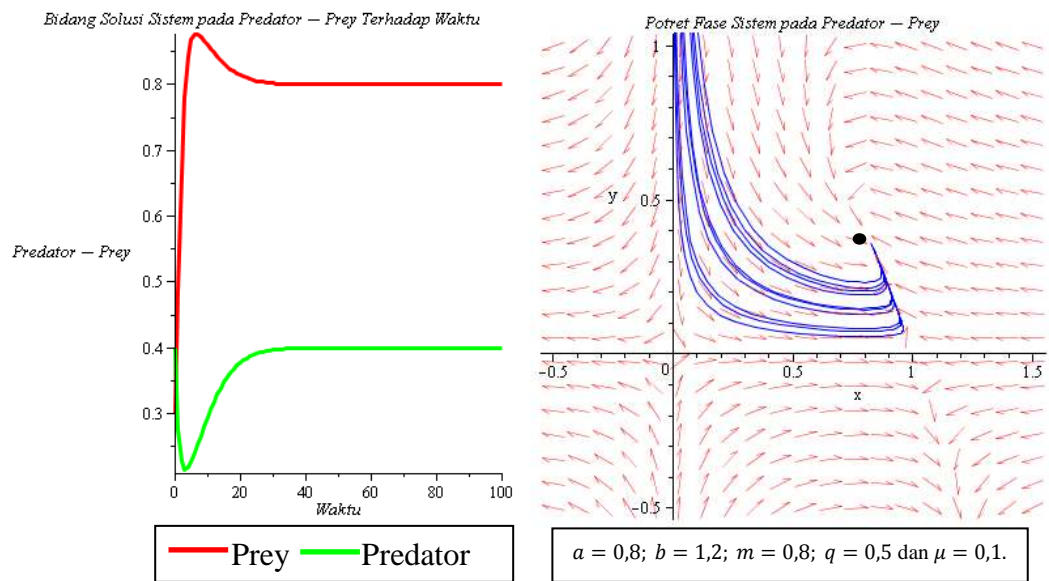
Gambar 2.7 Bidang solusi serta potret fase sistem pada populasi *predator-prey* untuk $E_0(0,0)$ dengan syarat $b > \mu + q$ menyebabkan E_0 bersifat pelana tidak stabil serta untuk $E_1(1,0)$ dengan syarat $\frac{b}{m^2+1} < \mu + q$ dan $b > \mu + q$ menyebabkan E_1 bersifat simpul stabil



Gambar 2.8 Bidang solusi serta potret fase sistem pada populasi *predator-prey*

untuk $E_1(1,0)$ dengan syarat $b > \mu + q$ dan $\frac{b}{m^2+1} > \mu + q$

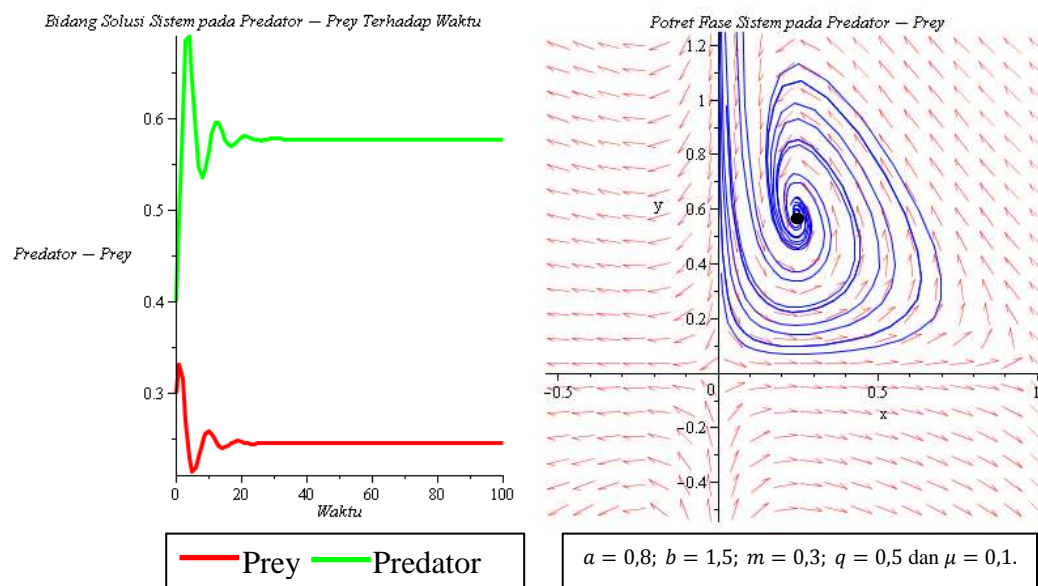
menyebabkan E_1 bersifat pelana tidak stabil



Gambar 2.9 Bidang solusi serta potret fase sistem pada populasi *predator-prey*

untuk $E_2(0,8; 0,4)$ dengan syarat $b > \mu + q, m \sqrt{\frac{\mu+q}{b-(\mu+q)}} < 1, p_1 >$

0 dan $p_1^2 - 4p_2 > 0$ menyebabkan E_2 bersifat simpul stabil



Gambar 2.10 Bidang solusi serta potret fase sistem pada populasi *predator-prey*

untuk $E_2(0,244948743; 0,5779655447)$ dengan syarat $b > \mu +$

$$q, m \sqrt{\frac{\mu+q}{b-(\mu+q)}} < 1, p_1 > 0 \text{ dan } p_1^2 - 4p_2 < 0 \text{ menyebabkan } E_2$$

simpul stabil

2.6 Sistem Dinamik

Secara umum sistem dinamik didefinisikan sebagai sebuah masalah nyata yang dimodelkan secara matematis dengan menggunakan persamaan-persamaan diferensial dimana dalam persamaannya mengandung parameter-parameter yang saling berhubungan, serta perubahan parameter pada persamaan tersebut akan menyebabkan perubahan kestabilan dari titik ekuilibrium. Definisi formal dari sistem dinamik adalah

Definisi 2.1 (Perko, 2000: 182)

Sistem dinamik pada E adalah pemetaan $\phi \in C^1$,

$$\phi: \mathbb{R} \times E \rightarrow E,$$

dengan E adalah himpunan bagian terbuka dari \mathbb{R}^n dan jika $\phi_t(x) = \phi(t, x)$, maka ϕ_t memenuhi

$$(1) \quad \phi_0(x) = x, \forall x \in E \text{ dan}$$

$$(2) \quad \phi_t \circ \phi_s(x) = \phi_{t+s}(x), \forall x \in E \text{ dan } t, s \in \mathbb{R}.$$

Jika dikaji secara geometri, sistem dinamik menggambarkan pergerakan titik-titik di dalam ruang fase sepanjang kurva-kurva solusi dari sistem persamaan diferensialnya. Jika menyebut solusi suatu sistem dinamik dalam bentuk grafik maka akan muncul sesuatu yang disebut dengan orbit.

2.6.1 Orbit

Definisi 2.2 (Wiggins, 1990: 2)

Orbit melalui x_0 , dinotasikan sebagai $\text{Or}(x_0)$, adalah himpunan titik-titik x dalam ruang keadaan X yang berada pada suatu *flow* sehingga $x = \varphi^t(x_0)$, yakni

$$\text{Or}(x_0) = \{x \in X : x = \varphi^t(x_0), t \in T\}.$$

Dalam kenyataannya tidak semua sistem persamaan diferensial dapat ditentukan solusi dari sistemnya, maka dari itu satu tujuan utama dari sistem dinamik adalah mempelajari perilaku dari solusi sistem di sekitar titik ekuilibrium.

2.6.2 Titik Ekuilibrium

Orbit paling sederhana adalah titik ekuilibrium. Definisi titik ekuilibrium secara formal adalah

Definisi 2.3 (Supriyono, 2014: 6)

Titik $x_0 \in X$ dikatakan **titik ekuilibrium** jika memenuhi $\varphi^t(x_0) = x_0$ untuk semua $t \in T$.

Untuk mempelajari perilaku dari solusi sistem tersebut digunakan suatu pendekatan yang disebut analisis kestabilan. Analisis ini dapat dilakukan dengan beberapa cara seperti melakukan penyelidikan terhadap perilaku titik setimbang dari persamaan diferensial. Titik ekuilibrium dan kestabilannya dapat memberikan informasi mengenai perilaku solusi periodik dari persamaan diferensial.

2.6.3 Solusi Periodik

Definisi solusi periodik secara formal adalah

Definisi 2.4 (Hale & Kocak, 1991: 181)

Misalkan x_0 bukanlah suatu titik ekuilibrium, suatu solusi $\varphi^t(x_0)$ dikatakan **solusi periodik** dengan periode $T_0 > 0$. Jika $\varphi^{t+T_0}(x_0) = \varphi^t(x_0)$, untuk setiap $t \in T$.

Orbit tertutup dalam sistem dinamik disebut *cycle*. Dalam sistem dinamik yang kontinu, *cycle* disebut juga sebagai *limit cycle*.

2.6.4 Limit Cycle

Definisi *limit cycle* secara formal adalah

Definisi 2.5 (Kuznetsov, 1998: 10)

Sebuah *cycle* dari sistem dinamik kontinu yang pada daerah sekitarnya tidak ada *cycle* lain, disebut *cycle* batas atau **limit cycle**.

Dalam sistem dinamik kurva-kurva solusi bisa dihimpun sebagai suatu himpunan kurva solusi atau sering disebut sebagai potret fase.

2.6.5 Potret Fase

Definisi potret fase secara formal adalah

Definisi 2.6 (Panigoro, 2011: 4)

Potret fase adalah gabungan beberapa orbit dari sistem persamaan diferensial yang ditampilkan dalam satu bidang.

2.7 Nilai eigen dan Vektor eigen

Dalam mempelajari keadaan dinamik dari suatu sistem khususnya sistem linear dapat menggunakan sifat dari nilai eigen sistem dinamik tersebut. Secara formal definisi nilai eigen dan vektor eigen adalah sebagai berikut.

Definisi 2.7 (Anton, 1987: 277)

Misalkan A matriks $n \times n$ dan $x \in \mathbb{R}^n, x \neq 0$. Vektor x disebut **vektor eigen** atau vektor karakteristik dari A jika

$$Ax = \lambda x, \tag{2.16}$$

untuk suatu $\lambda \in \mathbb{R}$. Bilangan λ yang memenuhi persamaan di atas disebut **nilai eigen** atau nilai karakteristik. Vektor x disebut vektor eigen yang bersesuaian dengan λ .

Untuk mencari nilai dan vektor eigen dari suatu matriks A berordo $n \times n$ adalah sebagai berikut.

Jelas $Ax = \lambda x$

$$\Leftrightarrow Ax = \lambda Ix$$

$$\Leftrightarrow (\lambda I - A)x = 0.$$

Tampak bahwa x merupakan penyelesaian dari sistem persamaan linear (SPL) homogen $(\lambda I - A)x = 0$. Karena $x \neq 0$, maka sistem persamaan homogen

$(\lambda I - A)x = 0$ mempunyai penyelesaian *nontrivial*. Ini hanya mungkin jika $\det(\lambda I - A) = 0$, artinya λ adalah penyelesaian persamaan dari $\det(\lambda I - A) = 0$. Det $(\lambda I - A) = 0$ ini disebut persamaan karakteristik dari matriks A .

2.8 Sistem Linear

Dalam persamaan diferensial untuk menyelesaikan sistem linear salah satunya adalah menggunakan metode pemisahan variabel. Diberikan persamaan diferensial orde pertama sebagai berikut.

$$\dot{x} = ax. \quad (2.17)$$

Maka solusi umum dari persamaan di atas adalah

$$x(t) = x_0 e^{at}.$$

Berdasar hal itu, maka pada sistem persamaan diferensial linear

$$\dot{x} = Ax, \quad (2.18)$$

dengan $x \in \mathbb{R}^n$, A adalah matriks $n \times n$ dan

$$\dot{x} = \frac{dx}{dt} = \begin{pmatrix} \frac{dx_1}{dt} \\ \vdots \\ \frac{dx_n}{dt} \end{pmatrix}.$$

Maka akan memiliki solusi

$$x(t) = x_0 e^{At}, \quad (2.19)$$

dengan x_0 adalah kondisi awal. Jika $x(t) = x_0 e^{At}$ adalah solusi sistem persamaan diferensial linear (2.18) maka perlu dibuktikan $\dot{x} = Ax$ adalah turunan dari $x(t) = x_0 e^{At}$.

2.9 Kestabilan Sistem Linear

Definisi 2.8 (Strogatz, 1994)

Misalkan

$$A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}.$$

Persamaan karakteristik dari matriks A adalah

$$\det(A - \lambda I) = |A - \lambda I| = 0. \quad (2.20)$$

Dari persamaan (2.20), maka persamaan karakteristiknya menjadi

$$\begin{vmatrix} a - \lambda & b \\ c & d - \lambda \end{vmatrix} = 0,$$

demikian sehingga diperoleh persamaan

$$\lambda^2 - \tau\lambda + \Delta = 0,$$

dengan

$$\tau = \text{trace}(A) = a + d = \lambda_1 + \lambda_2,$$

$$\Delta = \det(A) = ad - bc = \lambda_1\lambda_2.$$

Dengan demikian diperoleh nilai eigen dari matriks A sebagai berikut.

$$\lambda_{1,2} = \frac{\tau \pm \sqrt{\tau^2 - 4\Delta}}{2}.$$

Menurut Strogatz (1994), untuk menentukan kestabilan titik tetap dari suatu sistem dapat dilihat dari nilai Δ . Ada 3 kasus untuk nilai Δ , yaitu:

(1) $\Delta < 0$

Jika kedua nilai eigen berbeda tanda maka titik tetap bersifat sadel.

(2) $\Delta > 0$

(a) $\tau^2 - 4\Delta > 0$

(i) Jika $\tau > 0$ dan kedua nilai eigen real bernilai positif, maka titik tetap bersifat simpul tak stabil.

(ii) Jika $\tau < 0$ dan kedua nilai eigen real bernilai negatif, maka titik tetap bersifat simpul stabil.

(b) $\tau^2 - 4\Delta < 0$

(i) Jika $\tau > 0$ dan kedua nilai eigen merupakan bilangan kompleks, maka titik tetap bersifat spiral tak stabil.

(ii) Jika $\tau < 0$ dan kedua nilai eigen merupakan bilangan kompleks, maka titik tetap bersifat spiral stabil.

(iii) Jika $\tau = 0$ dan kedua nilai eigen merupakan bilangan kompleks, maka titik tetap bersifat center.

(c) $\tau^2 - 4\Delta = 0$

Parabola adalah garis batas antara simpul dan spiral. *Star nodes* atau *degenerate* terletak pada parabola ini. Jika kedua nilai eigen bernilai sama maka titik tetap bersifat simpul sejati.

(3) $\Delta = 0$

Jika salah satu nilai eigen bernilai nol, maka titik tetap bersifat titik tetap tak terisolasi.

2.10 Sistem *Nonlinear*

Diberikan persamaan *nonlinear* berikut.

$$\dot{x} = f(x). \quad (2.21)$$

Jika persamaan (2.21) mempunyai titik ekuilibrium \bar{x} maka persamaan (2.21) dapat ditulis sebagai

$$\dot{x} = Df(\bar{x})x + \varphi(x). \quad (2.22)$$

Bentuk $\varphi(x)$ disebut sebagai bagian *nonlinear* dari persamaan (2.21) dan $Df(\bar{x})$ disebut sebagai bagian linear dari persamaan (2.21), dengan $Df(\bar{x})$ disebut sebagai **matriks Jacobian** dari persamaan (2.21) pada titik ekuilibrium \bar{x} . Secara formal definisi matriks jacobian adalah

Definisi 2.9 (Clark, 1999: 140)

Matriks yang berhubungan dengan sebuah fungsi $f: \mathbb{R}^n \rightarrow \mathbb{R}^n$ yang memiliki koordinat fungsi f_1, f_2, \dots, f_m dengan entri (i, j) dari $\frac{\partial f_i}{\partial x_j}(x_0)$, turunan parsial pertama dari daerah asal fungsi f .

$$J = \begin{pmatrix} \frac{\partial f_1}{\partial x_1}(x_0) & \cdots & \frac{\partial f_1}{\partial x_n}(x_0) \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1}(x_0) & \cdots & \frac{\partial f_m}{\partial x_n}(x_0) \end{pmatrix}. \quad (2.23)$$

Jika $Df(\bar{x})$ tidak mempunyai nilai eigen dengan $\text{Re}(\lambda_j) = 0$ maka sifat kestabilan dari sistem (2.21) dapat dilihat dari persamaan

$$\dot{x} = Df(\bar{x})x. \quad (2.24)$$

Sistem (2.22) kemudian disebut sebagai persamaan hasil linearisasi dari persamaan (2.21).

2.11 Bifurkasi

Menurut Guckenheimer dan Holmes (1990), pada sistem dinamik tak linear akan sering dijumpai transisi dari keadaan stabil ke suatu keadaan tidak stabil ataupun sebaliknya yaitu transisi dari keadaan tidak stabil ke keadaan stabil. Kondisi seperti ini disebut dengan bifurkasi. Analisa bifurkasi adalah bagian terpenting dari suatu sistem. Misalkan diberikan sistem dinamik sebagai berikut.

$$\dot{x} = f(x, \mu),$$

dengan $x \in \mathbb{R}^n$ adalah suatu variabel dan $\mu \in \mathbb{R}$ adalah parameter. Sebagai konsekuensi adanya parameter yang dapat mempengaruhi perubahan kestabilan titik setimbang, maka dikatakan strukturnya tidak stabil. Nilai parameter yang mempengaruhi sifat kualitatif sistem berubah sesuai dengan nilai bifurkasi.

Dengan demikian bifurkasi dapat didefinisikan sebagai perubahan kualitatif yang terjadi pada penyelesaian persamaan diferensial. Perubahan kualitatif meliputi perubahan stabilitas dan jumlah titik setimbang yang diakibatkan perubahan parameter.

2.12 Bifurkasi Hopf

Pandang persamaan sistem persamaan diferensial

$$\begin{aligned} \dot{x} &= f(x, y, \mu), \\ \dot{y} &= g(x, y, \mu). \end{aligned} \tag{2.25}$$

Diasumsikan sistem (2.25) mempunyai titik setimbang (x^*, y^*) dan $\mu = \mu^*$ adalah nilai parameter yang menyebabkan terjadinya bifurkasi.

Menurut Guckenheimer dan Holmes (1990), bifurkasi Hopf terjadi jika titik setimbang (x^*, y^*) mempunyai sepasang nilai eigen kompleks yaitu $\lambda(\mu^*) = p(\mu^*) \pm iq(\mu^*)$ dengan $p(\mu^*) = 0, q(\mu^*) \neq 0$ dan memenuhi syarat transversal. Untuk menentukan bifurkasi Hopf, syarat yang harus dipenuhi adalah sebagai berikut.

Misal $\lambda(\mu^*)$ adalah nilai eigen pada titik setimbang (x^*, y^*) , maka

- (1) $\text{Re}\lambda(\mu^*) = 0$ (bagian real = 0).
- (2) $\text{Im}\lambda(\mu^*) = \omega \neq 0$ (bagian imajiner $\neq 0$).

$$(3) \quad d = \frac{d}{d\mu^*} [\operatorname{Re}\lambda(\mu^*)] \Big|_{\mu=\mu^*} \neq 0 \text{ (syarat transversal).}$$

Sehingga jika ketiga syarat tersebut terpenuhi maka akan terjadi bifurkasi Hopf di sekitar titik setimbang (x^*, y^*) .

2.13 Potret Fase dari Sistem Linear

Dipunyai persamaan-persamaan sebagai berikut.

$$\begin{aligned} \dot{x} &= ax + by, \\ \dot{y} &= cx + dy, \end{aligned} \tag{2.26}$$

dengan a, b, c dan d konstanta-konstanta. Misalkan $ad - bc \neq 0$, maka titik $(0,0)$ adalah satu-satunya titik kritik dari sistem (2.26). Penyelesaian dari sistem (2.26) berbentuk $x = Ae^{\lambda t}$ dan $y = Be^{\lambda t}$, dimana λ adalah nilai eigen dari matriks

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix},$$

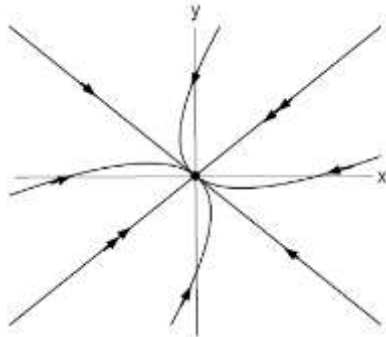
yaitu, λ merupakan akar persamaan karakteristik

$$\lambda^2 - (a + d)\lambda + ad - bc = 0. \tag{2.27}$$

Potret fase dari sistem (2.27) hampir seluruhnya tergantung pada nilai-nilai eigennya (λ_1 dan λ_2).

(1) Jika nilai-nilai eigennya real berbeda, dengan $\lambda_1 < \lambda_2 < 0$ ini disebut *node*: semua trayektori menuju ke tak nol yang berarti titik kritik nol adalah stabil.

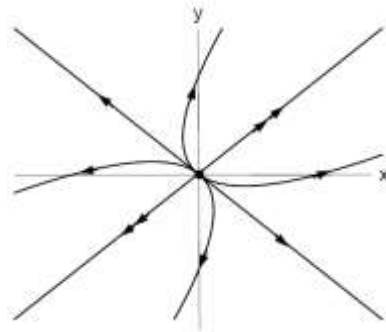
Trayektori pada kasus ini dapat dilihat pada Gambar 2.11.



Gambar 2.11 Trayektori untuk *node point*

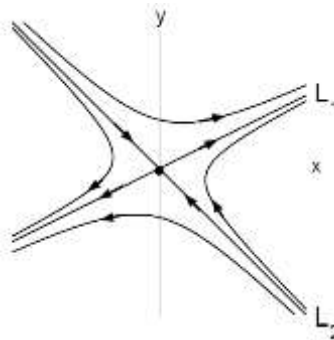
(2) Jika nilai-nilai eigennya real berbeda, dengan $\lambda_1 > \lambda_2 > 0$ ini disebut *nodal source*: semua trayektori keluar dari titik kritiknya menjadi tak stabil.

Trayektori pada kasus ini dapat dilihat pada Gambar 2.12.



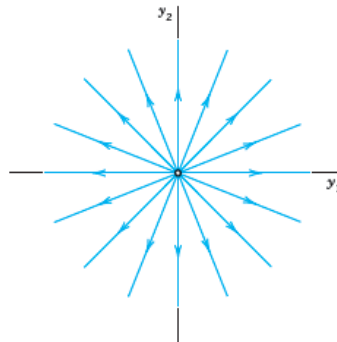
Gambar 2.12 Trayektori untuk titik *nodal source*

(3) Jika nilai-nilai eigennya real berbeda, dengan $\lambda_1 < 0 < \lambda_2$ ini disebut *saddle point*: semua trayektori akan menjauhi ke tak hingga sepanjang vektor eigen, ini mengakibatkan titik kritik akan selalu tak stabil. Trayektori pada kasus ini dapat dilihat pada Gambar 2.13.



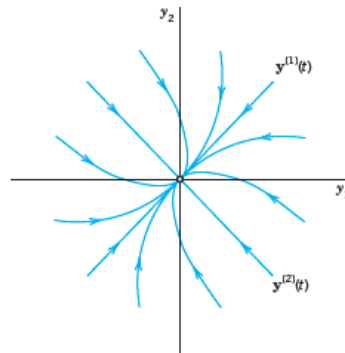
Gambar 2.13 Trayektori untuk *saddle point*

- (4) Jika nilai-nilai eigennya sama, dengan dua vektor eigen yang bebas linear, maka akan diperoleh apa yang dinamakan *star point* atau *proper node*: bila $\lambda < 0$ maka titik kritiknya akan stabil dan tak stabil untuk $\lambda > 0$. Trayektori pada kasus ini dapat dilihat pada Gambar 2.14.

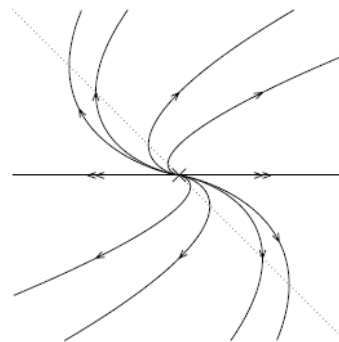


Gambar 2.14 Trayektori untuk *star point*

- (5) Jika nilai-nilai eigennya sama, dengan satu vektor eigen, maka akan diperoleh apa yang dinamakan *improper node*: bila $\lambda < 0$ maka titik kritiknya akan stabil dan arah trayektorinya akan menuju ke titik nol, sedangkan untuk $\lambda > 0$ arah trayektorinya akan keluar meninggalkan titik nol dan titik kritiknya akan tak stabil. Trayektori pada kasus ini dapat dilihat pada Gambar 2.15 dan 2.16.

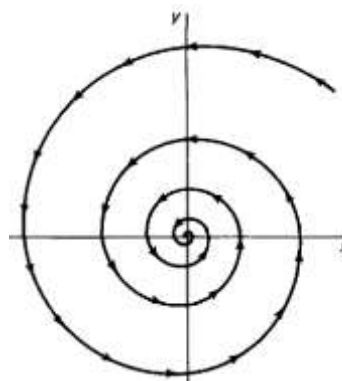


Gambar 2.15 Trayektori untuk *improper node* dengan $\lambda < 0$



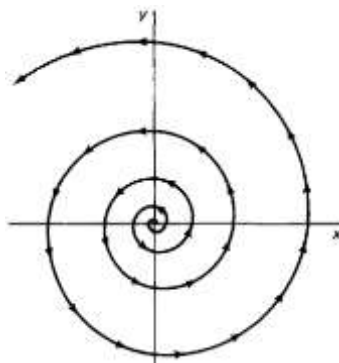
Gambar 2.16 Trayektori untuk *improper node* dengan $\lambda > 0$

(6) Jika nilai-nilai eigennya merupakan bilangan kompleks $\lambda_{\pm} = \rho \pm iw$ dengan $\rho < 0$, maka akan menghasilkan perilaku yang disebut *stabel spiral*: semua trayektori akan menuju titik nol dan titik kritiknya akan stabil. Trayektori pada kasus ini dapat dilihat pada Gambar 2.17.



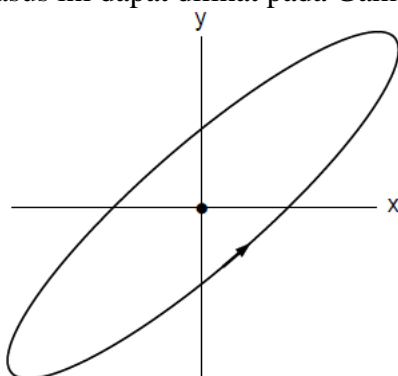
Gambar 2.17 Trayektori untuk *stabel spiral*

(7) Jika nilai-nilai eigennya merupakan bilangan kompleks $\lambda_{\pm} = \rho \pm iw$ dengan $\rho > 0$, maka akan menghasilkan perilaku yang disebut *unstable spiral*: semua trayektori akan keluar meninggalkan titik nol dan titik kritiknya akan tak stabil. Trayektori pada kasus ini dapat dilihat pada Gambar 2.18.



Gambar 2.18 Trayektori untuk *unstable spiral*

(8) Jika nilai eigennya imajiner murni, dalam kasus ini nilai eigennya dapat dinyatakan sebagai $i\lambda_{\pm} = \pm iw$ dalam hal ini solusi merupakan osilator stabil secara alami. Titik kritik dalam hal ini disebut *Center Point*. Trayektorinya berupa elips. Trayektori pada kasus ini dapat dilihat pada Gambar 2.19.



Gambar 2.19 Trayektori untuk *center point*

2.14 Maple

Maple adalah salah satu *software* yang sering digunakan dalam simulasi pemodelan matematika. Maple sendiri dikembangkan oleh Waterloo Inc. Kanada. Menu-menu yang terdapat pada tampilan program Maple ini terdiri dari menu *File, Edit, View, Insert, Format, Spreadsheet, Option, Window* dan *Help*.

Sebagian besar menu-menu di atas merupakan menu standar yang dikembangkan untuk program aplikasi pada sistem operasi *windows*. Maple memiliki banyak kelebihan salah satunya kemampuan untuk menyederhanakan persamaan, hingga suatu solusi persamaan diferensial dapat dipahami dengan baik. Selain itu Maple juga memiliki kemampuan untuk membuat animasi grafik dari suatu fenomena gerakan yang dimodelkan kedalam persamaan diferensial yang memiliki nilai awal dan syarat batas (Kartono, 2001).

Dalam pemodelan matematika khususnya yang menggunakan sistem persamaan diferensial pernyataan yang sering digunakan dalam Maple meliputi: *diff* digunakan untuk mendiferensialkan (menurunkan) suatu fungsi, *dsolve* digunakan untuk menyelesaikan persamaan diferensial, *evalf* memberikan nilai numerik dari suatu persamaan dan *simplify* digunakan untuk menyederhanakan suatu persamaan. Namun tentu saja pernyataan-pernyataan awal seperti *restart* dan deklarasi variabel atau konstanta yang diperlukan tidak boleh diabaikan. Untuk membuat grafik pada Maple digunakan perintah *plot, plot2d, plot3d*, tergantung dimensi dari pernyataan yang dimiliki. Untuk membuat gerakan animasi digunakan perintah *animate3d* (Kartono, 2001).

BAB 3

METODE PENELITIAN

Langkah awal dari pembuatan tulisan ini adalah menentukan tema yang ingin diangkat pada penelitian, dalam tulisan ini temanya adalah kasus bifurkasi Hopf pada model *predator-prey*. Kemudian dalam upaya melakukan penelitian ini, penulis mencari dan mengumpulkan rujukan-rujukan pendukung seperti jurnal internasional, buku teks dan artikel-artikel yang berkaitan dari internet. Lalu dari rujukan-rujukan tersebut hal-hal yang diperlukan dalam penelitian dituangkan pada bagian latar belakang dan landasan teori sehingga gambaran dan arah penelitian menjadi semakin jelas.

Berikut diberikan gambaran secara umum langkah-langkah dalam melakukan penelitian pada model *predator-prey* ditinjau dari terjadinya bifurkasi satu parameter yang dikaji dalam tulisan ini:

- (1) Menurunkan model matematika dari modifikasi sistem *predator-prey* Leslie Gower dengan fungsi respon Holling tipe II.
- (2) Menentukan semua titik tetap (titik kesetimbangan) dari sistem persamaan diferensial (sistem dinamik).
- (3) Menentukan persamaan karakteristik dan nilai eigen dari matriks Jacobian sistem yang dihitung pada setiap titik kesetimbangan.
- (4) Menentukan nilai parameter terjadinya bifurkasi Hopf, yaitu nilai parameter pada saat nilai eigen titik kesetimbangan imajiner murni.

- (5) Menghitung syarat transversal untuk lebih meyakinkan bahwa terjadi bifurkasi Hopf pada titik kesetimbangan dalam $E^*(x^*, y^*)$.
- (6) Membuat simulasi numerik dari modifikasi model *predator-prey* Leslie Gower dengan fungsi respon Holling tipe II dengan menggunakan *software* Maple.

Tahap akhir dari penelitian ini adalah penarikan kesimpulan dari pembahasan masalah dan pemecahan yang disesuaikan dengan konsep yang terkait.

BAB 5

PENUTUP

5.1 Simpulan

Dari pembahasan yang telah dilakukan, dapat diambil kesimpulan sebagai berikut.

- (5) Berdasarkan penondimensionalan model, diperoleh model matematika pada modifikasi sistem *predator-prey* Leslie Gower dengan Fungsi Respon Holling Tipe II sebagai berikut.

$$\begin{aligned} \text{(a)} \quad \frac{dx}{d\tau} &= x \left(1 - x - \frac{\alpha y}{m+x} - \frac{h}{c+x} \right), \\ \text{(b)} \quad \frac{dy}{d\tau} &= \rho y \left(1 - \frac{\beta y}{m+x} \right), \end{aligned} \tag{5.1}$$

dengan $x(0) > 0$ dan $y(0) > 0$.

- (6) Dari analisa model matematika yang dalam hal ini sistem persamaan (5.1) yang dianalisa diperoleh

Teorema 4.1

- (6) Tanpa adanya syarat, sistem persamaan (5.1) hanya mempunyai 2 titik ekuilibrium yaitu titik ekuilibrium $E_0(0,0)$ dan $E_1\left(0, \frac{m}{\beta}\right)$.
- (7) Jika $(c+1)^2 > 4h$ dan $0 < h < c < 1$, maka sistem persamaan (4.13) hanya mempunyai 3 titik ekuilibrium yaitu titik ekuilibrium $E_0(0,0)$, $E_1\left(0, \frac{m}{\beta}\right)$ dan $E_H(x_H, 0)$.

- (8) Jika $(c + 1)^2 > 4h$ dan $0 < c < h < 1$, maka sistem persamaan (4.13) hanya mempunyai 4 titik ekuilibrium yaitu titik ekuilibrium $E_0(0,0), E_1\left(0, \frac{m}{\beta}\right), E_H(x_H, 0)$ dan $E_L(x_L, 0)$.
- (9) Jika $(c + 1)^2 > 4h, 0 < h < c < 1, \left(\frac{\alpha}{\beta} - c - 1\right)^2 > 4h + c^2$ dan $\frac{\alpha}{\beta} + c < 1$, maka sistem persamaan (4.13) hanya mempunyai 4 titik ekuilibrium yaitu titik ekuilibrium $E_0(0,0), E_1\left(0, \frac{m}{\beta}\right), E_H(x_H, 0)$ dan $E^*(x^*, y^*)$.
- (10) Jika $(c + 1)^2 > 4h, 0 < c < h < 1, \left(\frac{\alpha}{\beta} - c - 1\right)^2 > 4h + c^2$ dan $\frac{\alpha}{\beta} + c < 1$, maka sistem persamaan (4.13) mempunyai 5 titik ekuilibrium yaitu titik ekuilibrium $E_0(0,0), E_1\left(0, \frac{m}{\beta}\right), E_H(x_H, 0), E_L(x_L, 0)$ dan $E^*(x^*, y^*)$.

Teorema 4.2

Dipunyai E_0 dan E_1 adalah titik ekuilibrium sistem (5.1) seperti pada Teorema 4.1.

- (3) Titik ekuilibrium E_0 pelana tidak stabil ketika $\frac{h}{c} > 1$ dan simpul tidak stabil ketika $\frac{h}{c} < 1$.
- (4) Titik ekuilibrium E_1 simpul stabil ketika $\frac{\alpha}{\beta} + \frac{h}{c} > 1$ dan titik pelana (*saddle point*) ketika $\frac{\alpha}{\beta} + \frac{h}{c} < 1$.

Teorema 4.3

Misalkan kondisi $(c + 1)^2 > 4h$ dan $c < 1$ dipenuhi serta E_H dan E_L adalah titik ekuilibrium sistem (5.1).

(3) Jika $h > c$, maka diperoleh

(c) Titik ekuilibrium E_H titik pelana (*saddle point*) ketika $c - \sqrt{A} < 0$ dan $c - \sqrt{A} > 0$.

(d) Titik ekuilibrium E_L simpul tidak stabil.

(4) Jika $h < c$, maka diperoleh titik ekuilibrium E_H pelana tidak stabil ketika $c - \sqrt{A} < 0$ dan $c - \sqrt{A} > 0$.

Teorema 4.4

Misalkan kondisi $\left(\frac{\alpha}{\beta} - c - 1\right)^2 > 4h + c^2$ dan $\frac{\alpha}{\beta} + c < 1$ dipenuhi serta

diberikan $w_0 = c - \sqrt{B} + \frac{\beta\alpha(1-c+\sqrt{B})-\alpha^2}{\beta^2(2m-c+\sqrt{B}-\frac{\alpha}{\beta}+1)} - \frac{2hc}{\left(1-\frac{\alpha}{\beta}+c\right)^2 + \sqrt{B}\left(c-\frac{\alpha}{\beta}+1\right)-2h}$, maka

(3) Titik ekuilibrium dalam $E^*(x^*, y^*)$ spiral tidak stabil ketika $w < w_0$ dan spiral stabil ketika $w > w_0$.

(4) Terjadi bifurkasi *Hopf* di titik ekuilibrium dalam $E^*(x^*, y^*)$ ketika $w = w_0$.

(7) Dalam menganalisis keberadaan bifurkasi Hopf, dinamika populasi *predator-prey* model ini dibagi menjadi tiga kasus dimana tiap kasusnya mengalami kenaikan nilai parameter tingkat pertumbuhan populasi *predator* ($\rho = w$). Dalam kasus nyata, perubahan parameter $w = \rho$ mengakibatkan perubahan kestabilan sistem *predator-prey*. Ketika tingkat pertumbuhan populasi *predator* tinggi, populasi *prey* dan *predator* akan

stabil. Sebaliknya ketika tingkat pertumbuhan populasi *predator* yang rendah menyebabkan kedua populasi mengalami penurunan jumlah, dimana populasi *prey* hampir terjadi kepunahan.

- (8) Simulasi model yang dilakukan memberikan hasil yang sama dengan hasil analisis.

5.2 Saran

Pada penulisan skripsi ini hanya dibahas tentang bifurkasi Hopf dan hanya pada satu titik ekuilibrium pada modifikasi model *Predator-Prey* Leslie Gower dengan fungsi respon Holling Tipe II. Jika pembaca tertarik bisa membahas titik-titik ekuilibrium yang lain. Pembaca juga bisa meneliti jenis bifurkasi lain yang mungkin terjadi pada sistem ini. Selain itu juga bisa menggunakan model *predator-prey* lain yang mungkin lebih mewakili keadaan sebenarnya.

DAFTAR PUSTAKA

- Abadi, Dian. S. & Choirotul. U. 2013. Stability Analysis of Lotka-Volterra Model with Holling Type II Functional Response. *Scientific Research Journal*, 1(5): 22-26.
- Altwegg, R. 2006. Functional Response and Prey Defence Level in an Experimental Predator-Prey System. *Evolutionary Ecology Research*, 8: 115-128.
- Anton, H. 1987. *Aljabar Linear Elementer, Edisi Kelima*. Jakarta: Erlangga.
- Bell, E. T. 1952. *Mathematics: Queen and Servant of Science*. London: G. Bell & Sons, Ltd.
- Boyce, W. E. & DiPrima R. C. 2009. *Elementary Differential Equations and Boundary Value Problems*. New York: Ninth Edition John Wiley & Sons.
- Clark, D. N. 1999. *Analysis Calculus and Differential Equation*. New York: CRC Press.
- Du, N. H., N. M. Man & T. T. Trung. 2007. Dynamics of Predator-Prey Population with Modified Leslie-Gower and Holling-Type II Schemes. *Acta Mathematica Vietnamica*, 32(1): 99-111.
- Finizio & Ladas. 1998. *Penerapan Diferensial Biasa dengan Penerapan Modern, Edisi Kedua*. Terjemahan Widiarti Santoso. Jakarta: Erlangga.
- Guckenheimer, J. M. & P. J. Holmes. 1990. *Nonlinear Oscillations, Dynamical Systems and Bifurcations of Vector Fields*. New York. Appl. Math Sciences: Springer-Verlag.
- Haberman. 1977. *Mathematical Models, Mechanical Vibration, Population Dynamic, and Traffic Flow, an Introduction to Applied Mathematic*. New Jersey: Prentice-Hall, Inc.
- Hale, J. K & H. Kocak. 1991. *Dynamics and bifurcations*. New York: Springer Verlag.
- Hsu, S. B. & T. W. Huang. 1999. Hopf Bifurcation Analysis for a Predator-Prey System of Holling and Leslie Type. *Taiwanese Journal of Mathematics*, 3: 35-53.

- Hunsicker, M. E., dkk. 2011. Functional Responses and Scaling in Predator-Prey Interactions of Marine Fishes: Contemporary Issues and Emerging Concepts. *Ecology Letters*, 14: 1288-1299.
- Kartono. 2001. *Maple untuk Persamaan Diferensial*. Yogyakarta: J & J Learning.
- Kuznetsov, Y. A. 1998. *Elements of Applied Bifurcation Theory, Second Edition*. New York: Springer-Verlag.
- Leslie, P. H. 1948. Some Further Notes on the Use of Matrices in Population Mathematics. *Biometrika*. 35(3/4): 213-245.
- Leslie, P. H. & J. C. Gower. 1960. The Properties of a Stochastic Model for the Predator-Prey Type of Interaction between Two Species. *Biometrika*, 47 (3/4): 219-234.
- Lin, C. M. & Ho, C. P. 2006. Local and Global Stability for a Predator-Prey Model of Modified Leslie-Gower and Holling-Type II with Time-Delay. *Science*, 8: 33-61.
- Ma, Y. F. 2012. Global Hopf Bifurcation in the Leslie-Gower Predator-Prey Model with Two Delays. *Nonlinear Analysis: Real World Applications*, 13: 370-375.
- Naispospos & Noeniek. S. 1983. *Aljabar Linear*. Terjemahan G. Hadley. Jakarta: Erlangga.
- Ndam, J. N. & T. G. Kassem. 2009. A Mathematical Model for the Dynamics of Predator-Prey Interactions in a Three-Trophic Level Food Web. *Continental J. Applied Sciences*, 4: 32-43.
- Panigoro, H. S. 2011. *Barisan Hingga Bifurkasi Period-Doubling Pada Interaksi Nonlinear Sepasang Osilator*. Bandung: Institut Teknologi Bandung.
- Peng, G. J., Y. L. Jiang & C. P. Li. 2009. Bifurcations of a Holling-type II Predator-Prey System with Constant Rate Harvesting. *Int. J. Bifurcation Chaos Appl. Sci. Eng.*, 19: 2499-2514.
- Perko, L. 1991. *Differential Equation and Dynamical System*. New York: Springer-Verlag Berlin Heidelberg.
- Perko, L. 2000. *Differential Equations and Dynamical System*. New York: Springer-Verlag.

- Rahadian, F. 2009. *Model Persaingan Dua Spesies*. Skripsi. Semarang: Jurusan Matematika UNNES.
- Ruan, S. 2009. On Nonlinear Dynamics of Predator-Prey Models with Discrete Delay. *Math. Model. Nat. Phenom*, 4: 140-188.
- Skalski, G. T. & J. F. Gilliam. 2001. Functional Responses with Predator Interference: Viable Alternatives to The Holling Type II Model. *Ecology*, 82(11): 3083-3092.
- Song, X. Y. & Y. F. Li. 2008. Dynamic Behaviors of The Periodic Predator-Prey Model with Modified Leslie-Gower Holling-type II Schemes and Impulsive Effect. *Nonlinear Analysis: Real World Applications*, 9: 64-79.
- Strogatz, S. H. 1994. *Nonlinear Dynamics and Chaos with Application to Physics, Biology, Chemistry and Engineering*. Massachusetts (US): Addison-Wesley Publishing Company.
- Supriyono. 2014. *Diktat Kuliah Sistem Dinamik*. Semarang: Jurusan Matematika UNNES.
- Tadu, S. S., F. A. Oroh & K. Nakil. 2013. *Analisis Bifurkasi Satu Parameter pada Sistem Predator-Prey dengan Respon Fungsi Tak Monoton*. Skripsi. Gorontalo: FMIPA Universitas Negeri Gorontalo.
- Timuneno, Henry. M., R. H. Utomo & Widowati. 2008. Model Pertumbuhan Logistik dengan Waktu Tunda. *Jurnal Matematika*, 11(1): 43-51.
- Tsai, C. H. & H. C. Pao. 2004. Global Stability for the Leslie-Gower Predator-Prey System with Time-Delay and Holling's Type Functional Response. *Tunghai Science*, 6: 43-72.
- Widowati & Sutimin. 2007. *Buku Ajar Pemodelan Matematika*. Semarang: Jurusan Matematika UNDIP.
- Wiggins, S. 1990. *Introduction to Applied Nonlinear Dynamic System and Chaos*. Springer-Verlag: New York.
- Wijayanti, P. 2014. *Analisis Model Predator-Prey Dua Spesies dengan Fungsi Respon Holling Tipe III*. Skripsi. Semarang: Jurusan Matematika UNNES.
- Zhu, C. R. & K. Q. Lan. 2011. Phase Portraits, Hopf Bifurcations and Limit Cycles of The Holling-Tanner Models for Predator-Prey Interactions. *Nonlinear Analysis: Real World Applications*, 12: 1961-1973.

LAMPIRAN

Lampiran 1

Penondimensiolan Model

Model persamaan (4.10)

$$(3) \quad \frac{dx_1}{dt} = rx_1 \left(1 - \frac{x_1}{K}\right) - \frac{a_1 x_1 x_2}{n+x_1} - \frac{qEx_1}{m_1 E + m_2 x_1},$$

$$(4) \quad \frac{dx_2}{dt} = sx_2 \left(1 - \frac{a_2 x_2}{n+x_1}\right),$$

dengan $x_1(0) > 0$ dan $x_2(0) > 0$.

Dilakukan penondimensiolan untuk mendapatkan sistem persamaan dengan parameter yang lebih sederhana. Skala parameter yang digunakan, yaitu

$$x_1 = Kx, \quad a_1 x_2 = Ky \Leftrightarrow x_2 = \frac{Ky}{a_1}, \quad rt = \tau \Leftrightarrow t = \frac{\tau}{r}, \quad \alpha = \frac{1}{r}, \quad c = \frac{m_1 E}{m_2 K} \Leftrightarrow m_1 E = cm_2 K, \quad \beta = \frac{a_2}{a_1}, \quad m = \frac{n}{K} \Leftrightarrow n = mK, \quad h = \frac{qE}{rm_2 K} \Leftrightarrow qE = hrm_2 K \quad \text{dan} \quad \rho = \frac{s}{r} \Leftrightarrow s = \rho r.$$

Dari persamaan (1) pada sistem persamaan (4.10) diperoleh

$$\begin{aligned} \frac{dx_1}{dt} &= rx_1 \left(1 - \frac{x_1}{K}\right) - \frac{a_1 x_1 x_2}{n+x_1} - \frac{qEx_1}{m_1 E + m_2 x_1} \\ \Leftrightarrow \frac{d(Kx)}{d\left(\frac{\tau}{r}\right)} &= r(Kx) \left(1 - \frac{Kx}{K}\right) - \frac{a_1(Kx)\left(\frac{Ky}{a_1}\right)}{mK+Kx} - \frac{(hrm_2 K)(Kx)}{cm_2 K + m_2(Kx)} \\ \Leftrightarrow \frac{dx}{d\tau} Kr &= Krx(1-x) - \frac{KxKy}{K(m+x)} - \frac{m_2 KhrKx}{cm_2 K + m_2 Kx} \\ \Leftrightarrow \frac{dx}{d\tau} Kr &= Krx(1-x) - \frac{xKy}{m+x} - \frac{m_2 KhrKx}{m_2 K(c+x)} \\ \Leftrightarrow \frac{dx}{d\tau} Kr &= Krx(1-x) - \frac{r^{\frac{1}{r}} xKy}{m+x} - \frac{hrKx}{c+x} \\ \Leftrightarrow \frac{dx}{d\tau} Kr &= Krx(1-x) - Kr \frac{\frac{1}{r} xy}{m+x} - Kr \frac{hx}{c+x} \\ \Leftrightarrow \frac{dx}{d\tau} &= x(1-x) - \frac{\frac{1}{r} xy}{m+x} - \frac{hx}{c+x} \\ \Leftrightarrow \frac{dx}{d\tau} &= x \left(1 - x - \frac{\alpha y}{m+x} - \frac{h}{c+x}\right). \end{aligned}$$

Dari persamaan (2) pada sistem persamaan (4.10) diperoleh

$$\begin{aligned} \frac{dx_2}{dt} &= sx_2 \left(1 - \frac{a_2 x_2}{n+x_1}\right) \\ \Leftrightarrow \frac{d\left(\frac{Ky}{a_1}\right)}{d\left(\frac{\tau}{r}\right)} &= (\rho r) \left(\frac{Ky}{a_1}\right) \left(1 - \frac{a_2 \left(\frac{Ky}{a_1}\right)}{mK+Kx}\right) \end{aligned}$$

$$\Leftrightarrow \frac{Kr}{a_1} \frac{dy}{d\tau} = \frac{Kr}{a_1} \rho y \left(1 - \frac{Ka_2 \frac{y}{a_1}}{K(m+x)} \right)$$

$$\Leftrightarrow \frac{dy}{d\tau} = \rho y \left(1 - \frac{a_2 \frac{y}{a_1}}{m+x} \right)$$

$$\Leftrightarrow \frac{dy}{d\tau} = \rho y \left(1 - \frac{\beta y}{m+x} \right).$$

Jadi sistem (4.10) ekuivalen dengan sistem berikut.

$$(9) \quad \frac{dx}{d\tau} = x \left(1 - x - \frac{\alpha y}{m+x} - \frac{h}{c+x} \right),$$

$$(10) \quad \frac{dy}{d\tau} = \rho y \left(1 - \frac{\beta y}{m+x} \right),$$

dengan $x(0) > 0$ dan $y(0) > 0$.

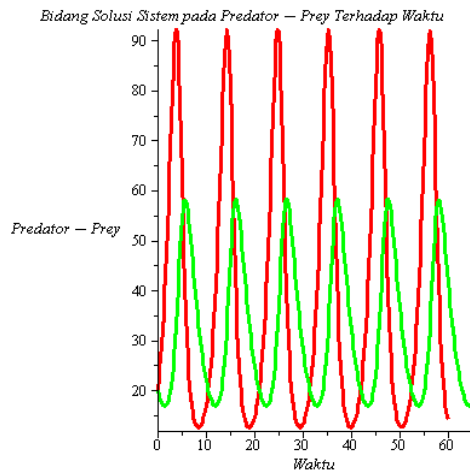
Lampiran 2

Print Out Maple 12 Dinamika Populasi Predator-Prey**Print Out Maple 12 Dinamika Populasi Predator-Prey dengan Fungsi Respon****Holling Tipe I****Titik Ekuilibrium T_0 dan T_1**

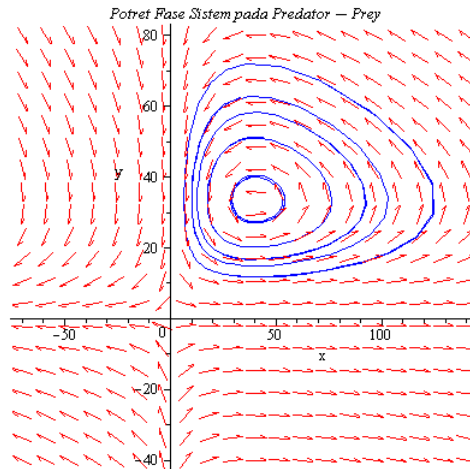
```

> restart :
> with(DEtools) :
> with(plots) :
> a := 1; b := 0.03; d := 0.4; c := 0.01;
                                     a := 1
                                     b := 0.03
                                     d := 0.4
                                     c := 0.01
> x0 := 0; y0 := 0;
                                     x0 := 0
                                     y0 := 0
> x1 :=  $\frac{d}{c}$ ; y1 :=  $\frac{a}{b}$ ;
                                     x1 := 40.00000000
                                     y1 := 33.33333333
> PD1 := diff(x(t), t) = a·x(t) - b·x(t)·y(t);
                                     PD1 :=  $\frac{d}{dt} x(t) = x(t) - 0.03 x(t) y(t)$ 
> PD2 := diff(y(t), t) = -d·y(t) + c·x(t)·y(t);
                                     PD2 :=  $\frac{d}{dt} y(t) = -0.4 y(t) + 0.01 x(t) y(t)$ 
>
display({phaseportrait([PD1, PD2], [x(t), y(t)], t = 0..60, [[x(0)
= 20, y(0) = 20]], stepsize = 0.2, scene = [t, x(t)], labels = [
'Waktu', 'Predator - Prey'], linecolour = [red]}),
{phaseportrait([PD1, PD2], [x(t), y(t)], t = 0..65, [[x(0) = 20,
y(0) = 20]], style = POINT, stepsize = 0.2, scene = [t, y(t)], labels
= ['Waktu', 'Predator - Prey'], title =
'Bidang Solusi Sistem pada Predator - Prey Terhadap Waktu',
linecolour = [green]})});

```



- > $dfp := dfieldplot([PD1, PD2], [x(t), y(t)], t = 0..12, x = -70..140, y = -40..80, title = 'Potret Fase Sistem pada Predator – Prey')$:
- > $fcns := \{x(t), y(t)\}$:
- > $sistem := PD1, PD2$;
- $sistem := \frac{d}{dt} x(t) = x(t) - 0.03 x(t) y(t), \frac{d}{dt} y(t) = -0.4 y(t) + 0.01 x(t) y(t)$
- > $awal1 := dsolve(\{x(0) = 15, y(0) = 15, sistem\}, fcns, type = numeric, method = classical)$:
- > $awal2 := dsolve(\{x(0) = 20, y(0) = 20, sistem\}, fcns, type = numeric, method = classical)$:
- > $awal3 := dsolve(\{x(0) = 30, y(0) = 30, sistem\}, fcns, type = numeric, method = classical)$:
- > $awal4 := dsolve(\{x(0) = 40, y(0) = 40, sistem\}, fcns, type = numeric, method = classical)$:
- > $awal5 := dsolve(\{x(0) = 50, y(0) = 50, sistem\}, fcns, type = numeric, method = classical)$:
- > $awal6 := dsolve(\{x(0) = 60, y(0) = 60, sistem\}, fcns, type = numeric, method = classical)$:
- > $gbawal1 := odeplot(awal1, [[x(t), y(t)]], -10..5, color = blue)$:
- > $gbawal2 := odeplot(awal2, [[x(t), y(t)]], -10..5, color = blue)$:
- > $gbawal3 := odeplot(awal3, [[x(t), y(t)]], -10..5, color = blue)$:
- > $gbawal4 := odeplot(awal4, [[x(t), y(t)]], -10..5, color = blue)$:
- > $gbawal5 := odeplot(awal5, [[x(t), y(t)]], -10..5, color = blue)$:
- > $gbawal6 := odeplot(awal6, [[x(t), y(t)]], -10..5, color = blue)$:
- > $display(gbawal1, gbawal2, gbawal3, gbawal4, gbawal5, gbawal6, dfp)$;



Print Out Maple 12 Dinamika Populasi Predator-Prey dengan Fungsi Respon Holling Tipe II

Titik Ekuilibrium T_0 dan T_1 (dengan Syarat $\mu < \frac{2}{3}$)

> restart :

> with(DEtools) :

> with(plots) :

> $\mu := \frac{4}{10}$;

$$\mu := \frac{2}{5}$$

> $\mu < \frac{2}{3}$:

> $x0 := 0; y0 := 0$;

$$x0 := 0$$

$$y0 := 0$$

> $x1 := 1; y1 := 0$;

$$x1 := 1$$

$$y1 := 0$$

> $PD1 := \text{diff}(x(t), t) = x(t) \cdot (1 - x(t)) - \frac{2 \cdot x(t) \cdot y(t)}{1 + 2 \cdot x(t)}$;

$$PD1 := \frac{d}{dt} x(t) = x(t) (1 - x(t)) - \frac{2 x(t) y(t)}{1 + 2 x(t)}$$

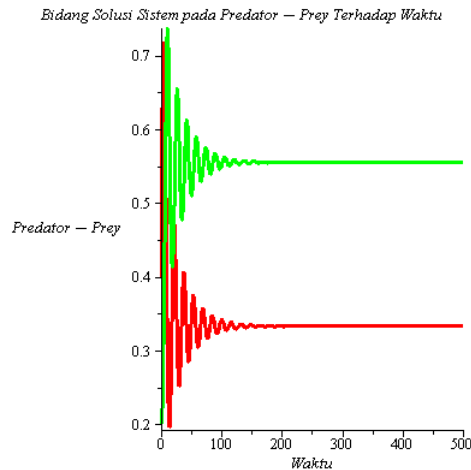
> $PD2 := \text{diff}(y(t), t) = y(t) \cdot \left(\frac{2 \cdot x(t)}{1 + 2 \cdot x(t)} - \mu \right)$;

$$PD2 := \frac{d}{dt} y(t) = y(t) \left(\frac{2 x(t)}{1 + 2 x(t)} - \frac{2}{5} \right)$$

```

> display( {phaseportrait ([PD1, PD2], [x(t), y(t)], t = 0 ..500, [[x(0)
= 0.4, y(0) = 0.2]], stepsize = 1, scene = [t, x(t)], labels = [
'Waktu','Predator - Prey'], linecolour = [red]]},
{phaseportrait ([PD1, PD2], [x(t), y(t)], t = 0 ..500, [[x(0)
= 0.3, y(0) = 0.2]], style = POINT, stepsize = 1, scene = [t, y(t)],
labels = ['Waktu','Predator - Prey'], title =
'Bidang Solusi Sistem pada Predator - Prey Terhadap Waktu',
linecolour = [green ]})});

```



```

> dfp := dfieldplot ([PD1, PD2], [x(t), y(t)], t = 0 ..100, x = -0.5 ..2, y =
-0.5 ..1, title='Potret Fase Sistem pada Predator - Prey') :

```

```

> fcns := {x(t), y(t)} :

```

```

> sistem := PD1, PD2;

```

$$\begin{aligned}
 \text{sistem} &:= \frac{d}{dt} x(t) = x(t) (1 - x(t)) - \frac{2x(t)y(t)}{1 + 2x(t)}, \frac{d}{dt} y(t) \\
 &= y(t) \left(\frac{2x(t)}{1 + 2x(t)} - \frac{2}{5} \right)
 \end{aligned}$$

```

> awal1 := dsolve ({x(0) = 0.2, y(0) = 0.4, sistem}, fcns, type
= numeric, method = classical) :

```

```

> awal2 := dsolve ({x(0) = 0.4, y(0) = 0.3, sistem}, fcns, type
= numeric, method = classical) :

```

```

> awal3 := dsolve ({x(0) = 0.35, y(0) = 0.4, sistem}, fcns, type
= numeric, method = classical) :

```

```

> awal4 := dsolve ({x(0) = 0.2, y(0) = 0.4, sistem}, fcns, type
= numeric, method = classical) :

```

```

> awal5 := dsolve ({x(0) = 0.4, y(0) = 0.5, sistem}, fcns, type
= numeric, method = classical) :

```

```

> awal6 := dsolve ({x(0) = 0.45, y(0) = 0.3, sistem}, fcns, type
= numeric, method = classical) :

```

```

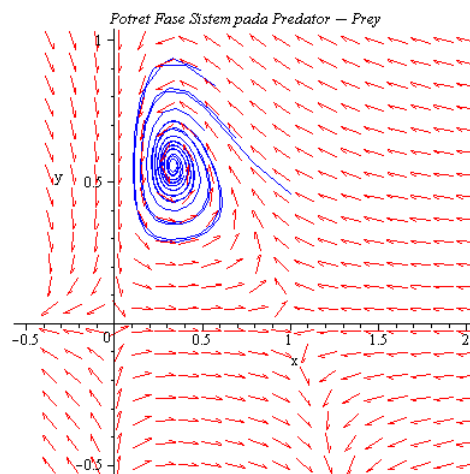
> awal7 := dsolve ({x(0) = 0.35, y(0) = 0.5, sistem}, fcns, type
= numeric, method = classical) :

```

- > `awal8 := dsolve({x(0) = 0.2, y(0) = 0.4, sistem}, fcns, type = numeric, method = classical) :`
- > `awal9 := dsolve({x(0) = 0.4, y(0) = 0.3, sistem}, fcns, type = numeric, method = classical) :`
- > `awal10 := dsolve({x(0) = 0.3, y(0) = 0.35, sistem}, fcns, type = numeric, method = classical) :`

- > `gbawal1 := odeplot(awal1, [[x(t), y(t)]], -10..35, color = blue) :`
- > `gbawal2 := odeplot(awal2, [[x(t), y(t)]], -10..35, color = blue) :`
- > `gbawal3 := odeplot(awal3, [[x(t), y(t)]], -10..35, color = blue) :`
- > `gbawal4 := odeplot(awal4, [[x(t), y(t)]], -10..35, color = blue) :`
- > `gbawal5 := odeplot(awal5, [[x(t), y(t)]], -10..35, color = blue) :`
- > `gbawal6 := odeplot(awal6, [[x(t), y(t)]], -10..35, color = blue) :`
- > `gbawal7 := odeplot(awal7, [[x(t), y(t)]], -10..35, color = blue) :`
- > `gbawal8 := odeplot(awal8, [[x(t), y(t)]], -10..35, color = blue) :`
- > `gbawal9 := odeplot(awal9, [[x(t), y(t)]], -10..35, color = blue) :`
- > `gbawal10 := odeplot(awal10, [[x(t), y(t)]], -10..35, color = blue) :`

- > `display(gbawal1, gbawal2, gbawal3, gbawal4, gbawal5, gbawal6, gbawal7, gbawal8, gbawal9, gbawal10, dfp) ;`



Titik Ekuilibrium T_1 dengan Syarat $\mu > \frac{2}{3}$

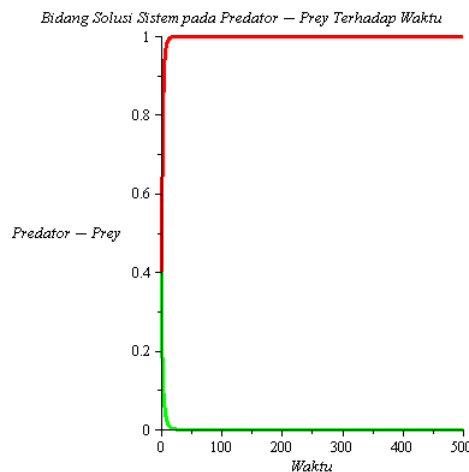
- > `restart :`
- > `with(DEtools) :`
- > `with(plots) :`
- > $\mu := \frac{9}{10} ;$

- > $\mu > \frac{2}{3} :$
- > `x1 := 1; y1 := 0;`

$$\mu := \frac{9}{10}$$

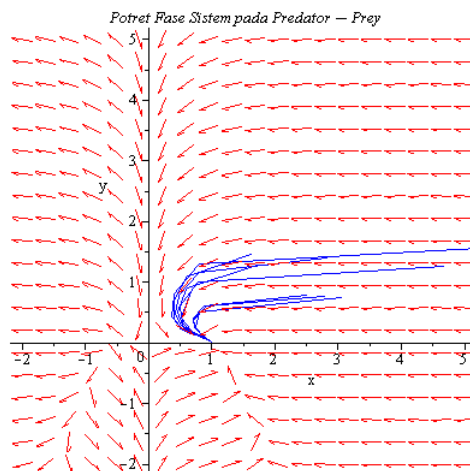
$$x1 := 1$$

- > $PD1 := \text{diff}(x(t), t) = x(t) \cdot (1 - x(t)) - \frac{2 \cdot x(t) \cdot y(t)}{1 + 2 \cdot x(t)}$;
- $$PD1 := \frac{d}{dt} x(t) = x(t) (1 - x(t)) - \frac{2 x(t) y(t)}{1 + 2 x(t)}$$
- > $PD2 := \text{diff}(y(t), t) = y(t) \cdot \left(\frac{2 \cdot x(t)}{1 + 2 \cdot x(t)} - \mu \right)$;
- $$PD2 := \frac{d}{dt} y(t) = y(t) \left(\frac{2 x(t)}{1 + 2 x(t)} - \frac{9}{10} \right)$$
- >
- ```
display({phaseportrait([PD1, PD2], [x(t), y(t)], t = 0..500, [[x(0)
= 0.4, y(0) = 0.3]], stepsize = 1, scene = [t, x(t)], labels = [
'Waktu', 'Predator - Prey'], linecolour = [red]}),
{phaseportrait([PD1, PD2], [x(t), y(t)], t = 0..500, [[x(0)
= 0.3, y(0) = 0.4]], style = POINT, stepsize = 1, scene = [t, y(t)],
labels = ['Waktu', 'Predator - Prey'], title =
'Bidang Solusi Sistem pada Predator - Prey Terhadap Waktu',
linecolour = [green]})});
```



- >  $dfp := \text{dfieldplot}([PD1, PD2], [x(t), y(t)], t = 0..1000, x = -2..5, y = -2..5, \text{title} = \text{'Potret Fase Sistem pada Predator - Prey'})$ ;
- >  $fns := \{x(t), y(t)\}$ ;
- >  $sistem := PD1, PD2$ ;
- $$sistem := \frac{d}{dt} x(t) = x(t) (1 - x(t)) - \frac{2 x(t) y(t)}{1 + 2 x(t)}, \frac{d}{dt} y(t) = y(t) \left( \frac{2 x(t)}{1 + 2 x(t)} - \frac{9}{10} \right)$$
- >  $awal1 := \text{dsolve}(\{x(0) = 0.8, y(0) = 0.5, sistem\}, fns, \text{type} = \text{numeric}, \text{method} = \text{classical})$ ;
- >  $awal2 := \text{dsolve}(\{x(0) = 0.7, y(0) = 0.9, sistem\}, fns, \text{type} = \text{numeric}, \text{method} = \text{classical})$ ;
- >  $awal3 := \text{dsolve}(\{x(0) = 0.6, y(0) = 0.9, sistem\}, fns, \text{type} = \text{numeric}, \text{method} = \text{classical})$ ;

- > `awal4 := dsolve ({x(0) = 0.4, y(0) = 0.8, sistem}, fcns, type = numeric, method = classical) :`
- > `awal5 := dsolve ({x(0) = 0.9, y(0) = 0.6, sistem}, fcns, type = numeric, method = classical) :`
- > `awal6 := dsolve ({x(0) = 0.5, y(0) = 0.3, sistem}, fcns, type = numeric, method = classical) :`
- > `awal7 := dsolve ({x(0) = 0.9, y(0) = 0.6, sistem}, fcns, type = numeric, method = classical) :`
- > `awal8 := dsolve ({x(0) = 0.7, y(0) = 0.4, sistem}, fcns, type = numeric, method = classical) :`
- > `awal9 := dsolve ({x(0) = 0.4, y(0) = 0.7, sistem}, fcns, type = numeric, method = classical) :`
- > `awal10 := dsolve ({x(0) = 0.5, y(0) = 0.6, sistem}, fcns, type = numeric, method = classical) :`
  
- > `gbawal1 := odeplot (awal1, [[x(t), y(t)]], -10..100, color = blue) :`
- > `gbawal2 := odeplot (awal2, [[x(t), y(t)]], -10..90, color = blue) :`
- > `gbawal3 := odeplot (awal3, [[x(t), y(t)]], -10..60, color = blue) :`
- > `gbawal4 := odeplot (awal4, [[x(t), y(t)]], -10..55, color = blue) :`
- > `gbawal5 := odeplot (awal5, [[x(t), y(t)]], -10..60, color = blue) :`
- > `gbawal6 := odeplot (awal6, [[x(t), y(t)]], -10..50, color = blue) :`
- > `gbawal7 := odeplot (awal7, [[x(t), y(t)]], -10..70, color = blue) :`
- > `gbawal8 := odeplot (awal8, [[x(t), y(t)]], -10..65, color = blue) :`
- > `gbawal9 := odeplot (awal9, [[x(t), y(t)]], -10..50, color = blue) :`
- > `gbawal10 := odeplot (awal10, [[x(t), y(t)]], -10..70, color = blue) :`
  
- > `display (gbawal1, gbawal2, gbawal3, gbawal4, gbawal5, gbawal6, gbawal7, gbawal8, gbawal9, gbawal10, dfp) ;`



### Titik Ekuilibrium $T_2$ Kasus $\mu < \frac{1}{3}$

- > restart :
- > with(DEtools) :
- > with(plots) :
- >  $\mu := \frac{1}{5}$ ;

$$\mu := \frac{1}{5}$$

- >  $\mu < \frac{1}{3}$  :

- >  $x2 := \text{evalf}\left(\frac{\mu}{2 \cdot (1 - \mu)}\right)$ ;  $y2 := \text{evalf}\left(\frac{2 - 3 \cdot \mu}{4 \cdot (1 - \mu)^2}\right)$ ;

$$x2 := 0.1250000000$$

$$y2 := 0.5468750000$$

- >  $PD1 := \text{diff}(x(t), t) = x(t) \cdot (1 - x(t)) - \frac{2 \cdot x(t) \cdot y(t)}{1 + 2 \cdot x(t)}$ ;

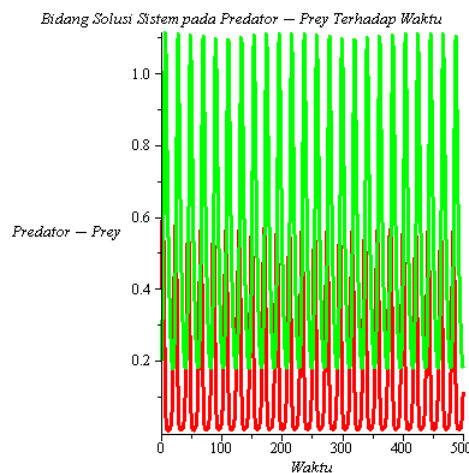
$$PD1 := \frac{d}{dt} x(t) = x(t) (1 - x(t)) - \frac{2 \cdot x(t) \cdot y(t)}{1 + 2 \cdot x(t)}$$

- >  $PD2 := \text{diff}(y(t), t) = y(t) \cdot \left(\frac{2 \cdot x(t)}{1 + 2 \cdot x(t)} - \mu\right)$ ;

$$PD2 := \frac{d}{dt} y(t) = y(t) \left(\frac{2 \cdot x(t)}{1 + 2 \cdot x(t)} - \frac{1}{5}\right)$$

>

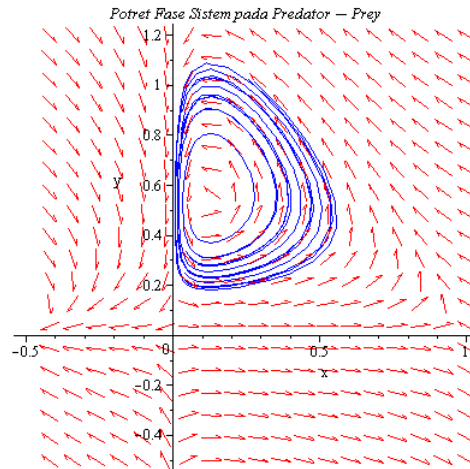
```
display({phaseportrait([PD1, PD2], [x(t), y(t)], t = 0 .. 500, [[x(0)
= 0.4, y(0) = 0.2]], stepsize = 1, scene = [t, x(t)], labels = [
'Waktu', 'Predator - Prey'], linecolour = [red])},
{phaseportrait([PD1, PD2], [x(t), y(t)], t = 0 .. 500, [[x(0)
= 0.3, y(0) = 0.2]], style = POINT, stepsize = 1, scene = [t, y(t)],
labels = ['Waktu', 'Predator - Prey'], title =
'Bidang Solusi Sistem pada Predator - Prey Terhadap Waktu',
linecolour = [green])});
```



- >  $dfp := \text{dfieldplot}([PD1, PD2], [x(t), y(t)], t = 0 .. 100, x = -0.5 .. 1, y = -0.5 .. 1.2, \text{title} = \text{'Potret Fase Sistem pada Predator - Prey'})$ ;

- >  $fcns := \{x(t), y(t)\}$  :

- > *sistem* := PD1, PD2;
 
$$\begin{aligned} \text{sistem} &:= \frac{d}{dt} x(t) = x(t) (1 - x(t)) - \frac{2x(t)y(t)}{1 + 2x(t)}, \frac{d}{dt} y(t) \\ &= y(t) \left( \frac{2x(t)}{1 + 2x(t)} - \frac{1}{5} \right) \end{aligned}$$
- > *awal1* := dsolve ( {x(0) = 0.2, y(0) = 0.4, sistem}, fcns, type = numeric, method = classical ) :
- > *awal2* := dsolve ( {x(0) = 0.4, y(0) = 0.3, sistem}, fcns, type = numeric, method = classical ) :
- > *awal3* := dsolve ( {x(0) = 0.35, y(0) = 0.4, sistem}, fcns, type = numeric, method = classical ) :
- > *awal4* := dsolve ( {x(0) = 0.2, y(0) = 0.4, sistem}, fcns, type = numeric, method = classical ) :
- > *awal5* := dsolve ( {x(0) = 0.4, y(0) = 0.5, sistem}, fcns, type = numeric, method = classical ) :
- > *awal6* := dsolve ( {x(0) = 0.45, y(0) = 0.3, sistem}, fcns, type = numeric, method = classical ) :
- > *awal7* := dsolve ( {x(0) = 0.35, y(0) = 0.5, sistem}, fcns, type = numeric, method = classical ) :
- > *awal8* := dsolve ( {x(0) = 0.2, y(0) = 0.4, sistem}, fcns, type = numeric, method = classical ) :
- > *awal9* := dsolve ( {x(0) = 0.4, y(0) = 0.3, sistem}, fcns, type = numeric, method = classical ) :
- > *awal10* := dsolve ( {x(0) = 0.3, y(0) = 0.35, sistem}, fcns, type = numeric, method = classical ) :
- > *gbawal1* := odeplot ( *awal1*, [[x(t), y(t)]], -5 ..25, color = blue ) :
- > *gbawal2* := odeplot ( *awal2*, [[x(t), y(t)]], -5 ..25, color = blue ) :
- > *gbawal3* := odeplot ( *awal3*, [[x(t), y(t)]], -5 ..25, color = blue ) :
- > *gbawal4* := odeplot ( *awal4*, [[x(t), y(t)]], -5 ..25, color = blue ) :
- > *gbawal5* := odeplot ( *awal5*, [[x(t), y(t)]], -5 ..25, color = blue ) :
- > *gbawal6* := odeplot ( *awal6*, [[x(t), y(t)]], -5 ..25, color = blue ) :
- > *gbawal7* := odeplot ( *awal7*, [[x(t), y(t)]], -5 ..25, color = blue ) :
- > *gbawal8* := odeplot ( *awal8*, [[x(t), y(t)]], -5 ..25, color = blue ) :
- > *gbawal9* := odeplot ( *awal9*, [[x(t), y(t)]], -5 ..25, color = blue ) :
- > *gbawal10* := odeplot ( *awal10*, [[x(t), y(t)]], -5 ..25, color = blue ) :
- > display ( *gbawal1*, *gbawal2*, *gbawal3*, *gbawal4*, *gbawal5*, *gbawal6*, *gbawal7*, *gbawal8*, *gbawal9*, *gbawal10*, dfp );



### Titik Ekuilibrium $T_2$ Kasus $\mu = \frac{1}{3}$

- > restart :
- > with(DEtools) :
- > with(plots) :
- >  $\mu := \frac{1}{3}$ ;

$$\mu := \frac{1}{3}$$

- >  $x2 := \text{evalf}\left(\frac{\mu}{2 \cdot (1 - \mu)}\right)$ ;  $y2 := \text{evalf}\left(\frac{2 - 3 \cdot \mu}{4 \cdot (1 - \mu)^2}\right)$ ;

$$x2 := 0.2500000000$$

$$y2 := 0.5625000000$$

- >  $PD1 := \text{diff}(x(t), t) = x(t) \cdot (1 - x(t)) - \frac{2 \cdot x(t) \cdot y(t)}{1 + 2 \cdot x(t)}$ ;

$$PD1 := \frac{d}{dt} x(t) = x(t) (1 - x(t)) - \frac{2 x(t) y(t)}{1 + 2 x(t)}$$

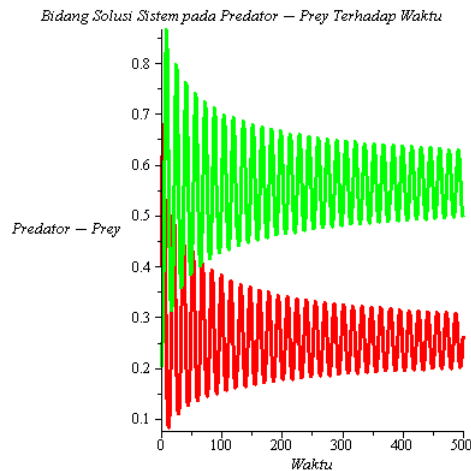
- >  $PD2 := \text{diff}(y(t), t) = y(t) \cdot \left(\frac{2 \cdot x(t)}{1 + 2 \cdot x(t)} - \mu\right)$ ;

$$PD2 := \frac{d}{dt} y(t) = y(t) \left(\frac{2 x(t)}{1 + 2 x(t)} - \frac{1}{3}\right)$$

>

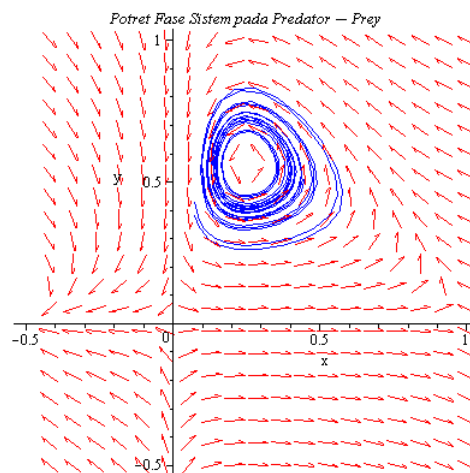
```
display({phaseportrait([PD1, PD2], [x(t), y(t)], t = 0..500, [[x(0)
= 0.4, y(0) = 0.2]], stepsize = 1, scene = [t, x(t)], labels = [
'Waktu', 'Predator - Prey'], linecolour = [red]}),
{phaseportrait([PD1, PD2], [x(t), y(t)], t = 0..500, [[x(0)
= 0.3, y(0) = 0.2]], style = POINT, stepsize = 1, scene = [t, y(t)],
labels = ['Waktu', 'Predator - Prey'], title =
'Bidang Solusi Sistem pada Predator - Prey Terhadap Waktu',
linecolour = [green]})});
```





- >  $dfp := dfieldplot([PD1, PD2], [x(t), y(t)], t = 0 .. 100, x = -0.5 .. 1, y = -0.5 .. 1, title = 'Potret Fase Sistem pada Predator - Prey')$  :
  - >  $fcns := \{x(t), y(t)\}$  :
  - >  $sistem := PD1, PD2$ ;
- $$sistem := \frac{d}{dt} x(t) = x(t) (1 - x(t)) - \frac{2 x(t) y(t)}{1 + 2 x(t)}, \frac{d}{dt} y(t) = y(t) \left( \frac{2 x(t)}{1 + 2 x(t)} - \frac{1}{3} \right)$$
- >  $awal1 := dsolve(\{x(0) = 0.2, y(0) = 0.4, sistem\}, fcns, type = numeric, method = classical)$  :
  - >  $awal2 := dsolve(\{x(0) = 0.4, y(0) = 0.3, sistem\}, fcns, type = numeric, method = classical)$  :
  - >  $awal3 := dsolve(\{x(0) = 0.35, y(0) = 0.4, sistem\}, fcns, type = numeric, method = classical)$  :
  - >  $awal4 := dsolve(\{x(0) = 0.2, y(0) = 0.4, sistem\}, fcns, type = numeric, method = classical)$  :
  - >  $awal5 := dsolve(\{x(0) = 0.4, y(0) = 0.5, sistem\}, fcns, type = numeric, method = classical)$  :
  - >  $awal6 := dsolve(\{x(0) = 0.45, y(0) = 0.3, sistem\}, fcns, type = numeric, method = classical)$  :
  - >  $awal7 := dsolve(\{x(0) = 0.35, y(0) = 0.5, sistem\}, fcns, type = numeric, method = classical)$  :
  - >  $awal8 := dsolve(\{x(0) = 0.2, y(0) = 0.4, sistem\}, fcns, type = numeric, method = classical)$  :
  - >  $awal9 := dsolve(\{x(0) = 0.4, y(0) = 0.3, sistem\}, fcns, type = numeric, method = classical)$  :
  - >  $awal10 := dsolve(\{x(0) = 0.3, y(0) = 0.35, sistem\}, fcns, type = numeric, method = classical)$  :
  - >  $gbawal1 := odeplot(awal1, [[x(t), y(t)]], -5 .. 35, color = blue)$  :

- > `gbrawal2 := odeplot(awal2, [[x(t), y(t)]], -5..35, color = blue) :`
- > `gbrawal3 := odeplot(awal3, [[x(t), y(t)]], -5..35, color = blue) :`
- > `gbrawal4 := odeplot(awal4, [[x(t), y(t)]], -5..35, color = blue) :`
- > `gbrawal5 := odeplot(awal5, [[x(t), y(t)]], -5..35, color = blue) :`
- > `gbrawal6 := odeplot(awal6, [[x(t), y(t)]], -5..35, color = blue) :`
- > `gbrawal7 := odeplot(awal7, [[x(t), y(t)]], -5..35, color = blue) :`
- > `gbrawal8 := odeplot(awal8, [[x(t), y(t)]], -5..35, color = blue) :`
- > `gbrawal9 := odeplot(awal9, [[x(t), y(t)]], -5..35, color = blue) :`
- > `gbrawal10 := odeplot(awal10, [[x(t), y(t)]], -5..35, color = blue) :`
- > `display(gbrawal1, gbrawal2, gbrawal3, gbrawal4, gbrawal5,`  
`gbrawal6, gbrawal7, gbrawal8, gbrawal9, gbrawal10, dfp) ;`



### Titik Ekuilibrium $T_2$ Kasus $\mu > \frac{1}{3}$

- > `restart :`
- > `with(DEtools) :`
- > `with(plots) :`
- >  $\mu := \frac{2}{5} ;$

$$\mu := \frac{2}{5}$$

- >  $\mu > \frac{1}{3} :$

- >  $x2 := \text{evalf}\left(\frac{\mu}{2 \cdot (1 - \mu)}\right) ; y2 := \text{evalf}\left(\frac{2 - 3 \cdot \mu}{4 \cdot (1 - \mu)^2}\right) ;$

$$x2 := 0.3333333333.$$

$$y2 := 0.5555555555.$$

- >  $PD1 := \text{diff}(x(t), t) = x(t) \cdot (1 - x(t)) - \frac{2 \cdot x(t) \cdot y(t)}{1 + 2 \cdot x(t)} ;$

$$PD1 := \frac{d}{dt} x(t) = x(t) (1 - x(t)) - \frac{2 x(t) y(t)}{1 + 2 x(t)}$$

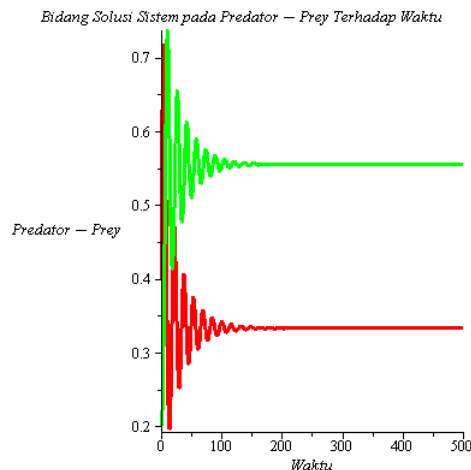
- >  $PD2 := \text{diff}(y(t), t) = y(t) \cdot \left(\frac{2 \cdot x(t)}{1 + 2 \cdot x(t)} - \mu\right) ;$

$$PD2 := \frac{d}{dt} y(t) = y(t) \left(\frac{2 x(t)}{1 + 2 x(t)} - \frac{2}{5}\right)$$

```

> display({phaseportrait ([PD1, PD2], [x(t), y(t)], t = 0 ..500, [[x(0)
= 0.4, y(0) = 0.2]], stepsize = 1, scene = [t, x(t)], labels = [
'Waktu','Predator - Prey'], linecolour = [red]]},
{phaseportrait ([PD1, PD2], [x(t), y(t)], t = 0 ..500, [[x(0)
= 0.3, y(0) = 0.2]], style = POINT, stepsize = 1, scene = [t, y(t)],
labels = ['Waktu','Predator - Prey'], title =
'Bidang Solusi Sistem pada Predator - Prey Terhadap Waktu',
linecolour = [green])});

```



```

> dfp := dfieldplot ([PD1, PD2], [x(t), y(t)], t = 0 ..100, x = -0.5 ..1, y =
-0.5 ..1, title = 'Potret Fase Sistem pada Predator - Prey') :

```

```

> fcns := {x(t), y(t)} :

```

```

> sistem := PD1, PD2;

```

$$\begin{aligned}
 \text{sistem} &:= \frac{d}{dt} x(t) = x(t) (1 - x(t)) - \frac{2x(t)y(t)}{1 + 2x(t)}, \frac{d}{dt} y(t) \\
 &= y(t) \left( \frac{2x(t)}{1 + 2x(t)} - \frac{2}{5} \right)
 \end{aligned}$$

```

> awal1 := dsolve ({x(0) = 0.2, y(0) = 0.4, sistem}, fcns, type
= numeric, method = classical) :

```

```

> awal2 := dsolve ({x(0) = 0.4, y(0) = 0.3, sistem}, fcns, type
= numeric, method = classical) :

```

```

> awal3 := dsolve ({x(0) = 0.35, y(0) = 0.4, sistem}, fcns, type
= numeric, method = classical) :

```

```

> awal4 := dsolve ({x(0) = 0.2, y(0) = 0.4, sistem}, fcns, type
= numeric, method = classical) :

```

```

> awal5 := dsolve ({x(0) = 0.4, y(0) = 0.5, sistem}, fcns, type
= numeric, method = classical) :

```

```

> awal6 := dsolve ({x(0) = 0.45, y(0) = 0.3, sistem}, fcns, type
= numeric, method = classical) :

```

```

> awal7 := dsolve ({x(0) = 0.35, y(0) = 0.5, sistem}, fcns, type
= numeric, method = classical) :

```

- ```

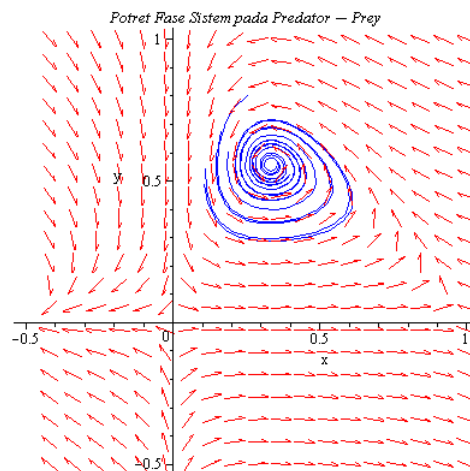
> awal8 := dsolve({x(0) = 0.2, y(0) = 0.4, sistem}, fcns, type
= numeric, method = classical) :

> awal9 := dsolve({x(0) = 0.4, y(0) = 0.3, sistem}, fcns, type
= numeric, method = classical) :

> awal10 := dsolve({x(0) = 0.3, y(0) = 0.35, sistem}, fcns, type
= numeric, method = classical) :

> gbawal1 := odeplot(awal1, [[x(t), y(t)]], -5..35, color = blue) :
> gbawal2 := odeplot(awal2, [[x(t), y(t)]], -5..35, color = blue) :
> gbawal3 := odeplot(awal3, [[x(t), y(t)]], -5..35, color = blue) :
> gbawal4 := odeplot(awal4, [[x(t), y(t)]], -5..35, color = blue) :
> gbawal5 := odeplot(awal5, [[x(t), y(t)]], -5..35, color = blue) :
> gbawal6 := odeplot(awal6, [[x(t), y(t)]], -5..35, color = blue) :
> gbawal7 := odeplot(awal7, [[x(t), y(t)]], -5..35, color = blue) :
> gbawal8 := odeplot(awal8, [[x(t), y(t)]], -5..35, color = blue) :
> gbawal9 := odeplot(awal9, [[x(t), y(t)]], -5..35, color = blue) :
> gbawal10 := odeplot(awal10, [[x(t), y(t)]], -5..35, color = blue) :
> display(gbawal1, gbawal2, gbawal3, gbawal4, gbawal5,
gbawal6, gbawal7, gbawal8, gbawal9, gbawal10, dfp);

```



Print Out Maple 12 Dinamika Populasi Predator-Prey dengan Fungsi Respon Holling Tipe III

Titik Ekuilibrium E_0 (dengan Syarat $b > \mu + q$) serta E_1 (dengan Syarat

$$\frac{b}{m^2+1} < \mu + q \text{ dan } b > \mu + q)$$

- ```

> restart :
> with(DEtools) :
> with(plots) :
> a := 0.8; b := 2; m := 2; q := 0.5; mu := 0.1;
a := 0.8
b := 2
m := 2

```

```

q := 0.5
μ := 0.1

> x0 := 0; y0 := 0;
x0 := 0
y0 := 0

> x1 := 1; y1 := 0;
x1 := 1
y1 := 0

> μ + q;
0.6

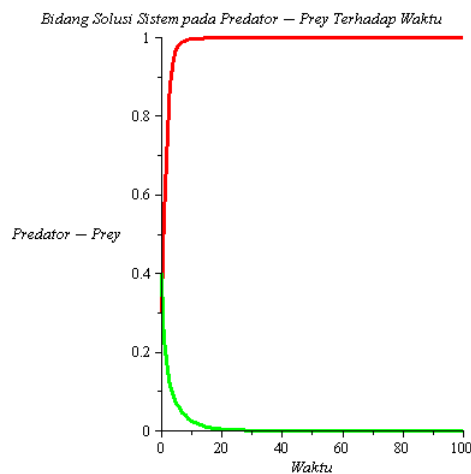
> b > μ + q :
> evalf((b / (m^2 + 1));
0.4000000000

> b / (m^2 + 1) < μ + q :
> PD1 := diff(x(t), t) = x(t) · (1 - x(t)) - a · x(t)^2 · y(t) / (m^2 + x(t)^2);
PD1 := d/dt x(t) = x(t) (1 - x(t)) - 0.8 x(t)^2 y(t) / (4 + x(t)^2)

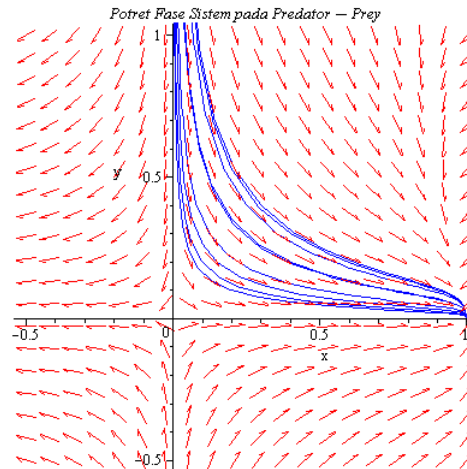
> PD2 := diff(y(t), t) = b · x(t)^2 · y(t) / (m^2 + x(t)^2) - (μ + q) · y(t);
PD2 := d/dt y(t) = 2 x(t)^2 y(t) / (4 + x(t)^2) - 0.6 y(t)

> display({ phaseportrait([PD1, PD2], [x(t), y(t)], t = 0 .. 100, [[x(0) = 0.3, y(0) = 0.4]], stepsize = 1, scene = [t, x(t)], labels = ['Waktu', 'Predator - Prey'], linecolour = [red]),
phaseportrait([PD1, PD2], [x(t), y(t)], t = 0 .. 100, [[x(0) = 0.3, y(0) = 0.4]], style = POINT, stepsize = 1, scene = [t, y(t)], labels = ['Waktu', 'Predator - Prey'], title = 'Bidang Solusi Sistem pada Predator - Prey Terhadap Waktu', linecolour = [green]) };

```



- >  $dfp := dfieldplot([PD1, PD2], [x(t), y(t)], t = 0 .. 100, x = -0.5 .. 1, y = -0.5 .. 1, title = 'Potret Fase Sistem pada Predator - Prey') :$
  - >  $fcns := \{x(t), y(t)\} :$
  - >  $sistem := PD1, PD2;$
- $$sistem := \frac{d}{dt} x(t) = x(t) (1 - x(t)) - \frac{0.8 x(t)^2 y(t)}{4 + x(t)^2}, \frac{d}{dt} y(t) = \frac{2 x(t)^2 y(t)}{4 + x(t)^2} - 0.6 y(t)$$
- >  $awal1 := dsolve(\{x(0) = 0.2, y(0) = 0.1, sistem\}, fcns, type = numeric, method = classical) :$
  - >  $awal2 := dsolve(\{x(0) = 0.3, y(0) = 0.4, sistem\}, fcns, type = numeric, method = classical) :$
  - >  $awal3 := dsolve(\{x(0) = 0.1, y(0) = 0.2, sistem\}, fcns, type = numeric, method = classical) :$
  - >  $awal4 := dsolve(\{x(0) = 0.25, y(0) = 0.4, sistem\}, fcns, type = numeric, method = classical) :$
  - >  $awal5 := dsolve(\{x(0) = 0.3, y(0) = 0.25, sistem\}, fcns, type = numeric, method = classical) :$
  - >  $awal6 := dsolve(\{x(0) = 0.4, y(0) = 0.3, sistem\}, fcns, type = numeric, method = classical) :$
  - >  $awal7 := dsolve(\{x(0) = 0.5, y(0) = 0.1, sistem\}, fcns, type = numeric, method = classical) :$
  - >  $awal8 := dsolve(\{x(0) = 0.2, y(0) = 0.35, sistem\}, fcns, type = numeric, method = classical) :$
  - >  $awal9 := dsolve(\{x(0) = 0.4, y(0) = 0.3, sistem\}, fcns, type = numeric, method = classical) :$
  - >  $awal10 := dsolve(\{x(0) = 0.15, y(0) = 0.2, sistem\}, fcns, type = numeric, method = classical) :$
- >  $gbawal1 := odeplot(awal1, [[x(t), y(t)]], -5 .. 20, color = blue) :$
  - >  $gbawal2 := odeplot(awal2, [[x(t), y(t)]], -5 .. 20, color = blue) :$
  - >  $gbawal3 := odeplot(awal3, [[x(t), y(t)]], -5 .. 20, color = blue) :$
  - >  $gbawal4 := odeplot(awal4, [[x(t), y(t)]], -5 .. 20, color = blue) :$
  - >  $gbawal5 := odeplot(awal5, [[x(t), y(t)]], -5 .. 20, color = blue) :$
  - >  $gbawal6 := odeplot(awal6, [[x(t), y(t)]], -5 .. 20, color = blue) :$
  - >  $gbawal7 := odeplot(awal7, [[x(t), y(t)]], -5 .. 20, color = blue) :$
  - >  $gbawal8 := odeplot(awal8, [[x(t), y(t)]], -5 .. 20, color = blue) :$
  - >  $gbawal9 := odeplot(awal9, [[x(t), y(t)]], -5 .. 20, color = blue) :$
  - >  $gbawal10 := odeplot(awal10, [[x(t), y(t)]], -5 .. 20, color = blue) :$
  - >  $display(gbawal1, gbawal2, gbawal3, gbawal4, gbawal5, gbawal6, gbawal7, gbawal8, gbawal9, gbawal10, dfp) ;$



**Titik Ekuilibrium  $E_1$  dengan Syarat  $b > \mu + q$  dan  $\frac{b}{m^2+1} > \mu + q$**

> restart :

> with(DEtools) :

> with(plots) :

> a := 0.9; b := 0.9; m := 0.4; q := 0.5;  $\mu$  := 0.1;

$$a := 0.9$$

$$b := 0.9$$

$$m := 0.4$$

$$q := 0.5$$

$$\mu := 0.1$$

>  $\mu + q$ ;

$$0.6$$

>  $b > \mu + q$  :

>  $\frac{b}{m^2 + 1}$ ;

$$0.7758620691$$

>  $\frac{b}{m^2 + 1} > \mu + q$  :

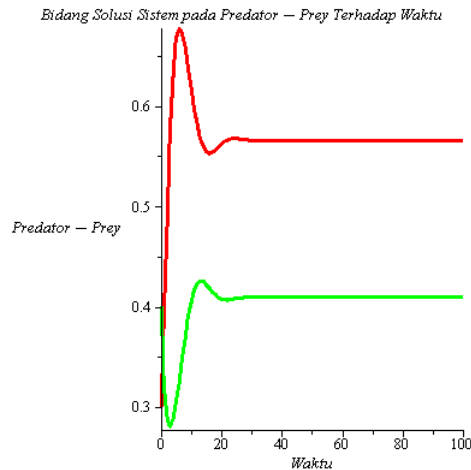
> PD1 := diff(x(t), t) = x(t) · (1 - x(t)) -  $\frac{a \cdot x(t)^2 y(t)}{m^2 + x(t)^2}$ ;

$$PD1 := \frac{d}{dt} x(t) = x(t) (1 - x(t)) - \frac{0.9 x(t)^2 y(t)}{0.16 + x(t)^2}$$

> PD2 := diff(y(t), t) =  $\frac{b \cdot x(t)^2 y(t)}{m^2 + x(t)^2}$  - ( $\mu + q$ ) · y(t);

$$PD2 := \frac{d}{dt} y(t) = \frac{0.9 x(t)^2 y(t)}{0.16 + x(t)^2} - 0.6 y(t)$$

```
> display({phaseportrait ([PD1, PD2], [x(t), y(t)], t = 0 .. 100, [[x(0)
= 0.3, y(0) = 0.4]], stepsize = 1, scene = [t, x(t)], labels = [
'Waktu', 'Predator - Prey'], linecolour = [red])},
{phaseportrait ([PD1, PD2], [x(t), y(t)], t = 0 .. 100, [[x(0)
= 0.3, y(0) = 0.4]], style = POINT, stepsize = 1, scene = [t, y(t)],
labels = ['Waktu', 'Predator - Prey'], title =
'Bidang Solusi Sistem pada Predator - Prey Terhadap Waktu',
linecolour = [green])});
```



```
> dfp := dfieldplot ([PD1, PD2], [x(t), y(t)], t = 0 .. 100, x = -0.5 .. 2.5, y
= -0.5 .. 1, title = 'Potret Fase Sistem pada Predator - Prey') :
```

```
> fcns := {x(t), y(t)} :
```

```
> sistem := PD1, PD2;
```

$$\begin{aligned} \text{sistem} &:= \frac{d}{dt} x(t) = x(t) (1 - x(t)) - \frac{0.9 x(t)^2 y(t)}{0.16 + x(t)^2}, \frac{d}{dt} y(t) \\ &= \frac{0.9 x(t)^2 y(t)}{0.16 + x(t)^2} - 0.6 y(t) \end{aligned}$$

```
> awal1 := dsolve ({x(0) = 0.2, y(0) = 0.1, sistem}, fcns, type
= numeric, method = classical) :
```

```
> awal2 := dsolve ({x(0) = 0.3, y(0) = 0.4, sistem}, fcns, type
= numeric, method = classical) :
```

```
> awal3 := dsolve ({x(0) = 0.1, y(0) = 0.2, sistem}, fcns, type
= numeric, method = classical) :
```

```
> awal4 := dsolve ({x(0) = 0.25, y(0) = 0.4, sistem}, fcns, type
= numeric, method = classical) :
```

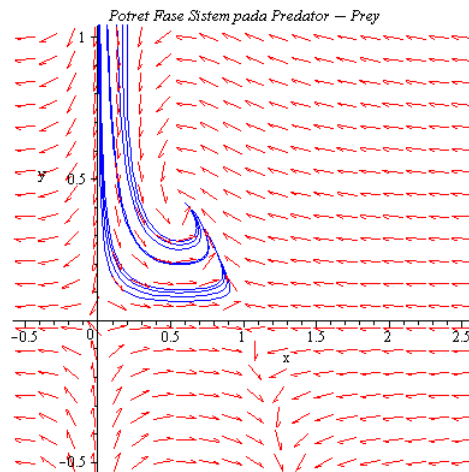
```
> awal5 := dsolve ({x(0) = 0.3, y(0) = 0.25, sistem}, fcns, type
= numeric, method = classical) :
```

```
> awal6 := dsolve ({x(0) = 0.4, y(0) = 0.3, sistem}, fcns, type
= numeric, method = classical) :
```

```
> awal7 := dsolve ({x(0) = 0.5, y(0) = 0.1, sistem}, fcns, type
= numeric, method = classical) :
```



- > `awal8 := dsolve({x(0) = 0.2, y(0) = 0.35, sistem}, fcns, type = numeric, method = classical) :`
- > `awal9 := dsolve({x(0) = 0.4, y(0) = 0.3, sistem}, fcns, type = numeric, method = classical) :`
- > `awal10 := dsolve({x(0) = 0.15, y(0) = 0.2, sistem}, fcns, type = numeric, method = classical) :`
  
- > `gbawal1 := odeplot(awal1, [[x(t), y(t)]], -5..10, color = blue) :`
- > `gbawal2 := odeplot(awal2, [[x(t), y(t)]], -5..10, color = blue) :`
- > `gbawal3 := odeplot(awal3, [[x(t), y(t)]], -5..10, color = blue) :`
- > `gbawal4 := odeplot(awal4, [[x(t), y(t)]], -5..10, color = blue) :`
- > `gbawal5 := odeplot(awal5, [[x(t), y(t)]], -5..10, color = blue) :`
- > `gbawal6 := odeplot(awal6, [[x(t), y(t)]], -5..10, color = blue) :`
- > `gbawal7 := odeplot(awal7, [[x(t), y(t)]], -5..10, color = blue) :`
- > `gbawal8 := odeplot(awal8, [[x(t), y(t)]], -5..10, color = blue) :`
- > `gbawal9 := odeplot(awal9, [[x(t), y(t)]], -5..10, color = blue) :`
- > `gbawal10 := odeplot(awal10, [[x(t), y(t)]], -5..10, color = blue) :`
- > `display(gbawal1, gbawal2, gbawal3, gbawal4, gbawal5, gbawal6, gbawal7, gbawal8, gbawal9, gbawal10, dfp) :`



**Titik Ekuilibrium  $E_2$  dengan Syarat  $b > \mu + q, m \sqrt{\frac{\mu+q}{b-(\mu+q)}} < 1, p_1 > 0$  dan**

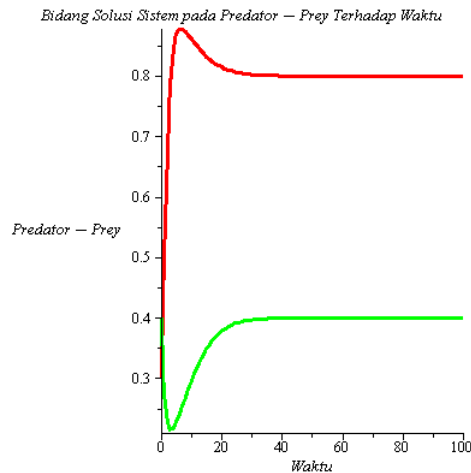
$$p_1^2 - 4p_2 > 0$$

- > `restart :`
- > `with(DEtools) :`
- > `with(plots) :`
- > `a := 0.8; b := 1.2; m := 0.8; q := 0.5; μ := 0.1;`
  - `a := 0.8`
  - `b := 1.2`
  - `m := 0.8`
  - `q := 0.5`
  - `μ := 0.1`

```

> $\mu + q;$
0.6
> $b > \mu + q;$
> $A := \frac{\mu + q}{b - (\mu + q)};$
A := 1.000000000
> $m \cdot \sqrt{A};$
0.800000000
> $m \cdot \sqrt{A} < 1;$
> $p1 := 2 \cdot (m \cdot \sqrt{A} - 1) \cdot \left(1 - \frac{1}{1 + A}\right) + 1;$
p1 := 0.800000000
> $p2 := \frac{2 \cdot b \cdot A \cdot (1 - m \cdot \sqrt{A})}{(1 + A)^2};$
p2 := 0.120000000
> $p1^2 - 4 \cdot p2;$
0.160000000
> $D = p1^2 - 4 \cdot p2 > 0;$
> $x2 := m \cdot \sqrt{A}; y2 := \frac{m \cdot (1 - m \cdot \sqrt{A}) \cdot (1 + A)}{a \cdot \sqrt{A}};$
x2 := 0.800000000
y2 := 0.400000000
> $PD1 := \text{diff}(x(t), t) = x(t) \cdot (1 - x(t)) - \frac{a \cdot x(t)^2 \cdot y(t)}{m^2 + x(t)^2};$
 $PD1 := \frac{d}{dt} x(t) = x(t) (1 - x(t)) - \frac{0.8 x(t)^2 y(t)}{0.64 + x(t)^2}$
> $PD2 := \text{diff}(y(t), t) = \frac{b \cdot x(t)^2 \cdot y(t)}{m^2 + x(t)^2} - (\mu + q) \cdot y(t);$
 $PD2 := \frac{d}{dt} y(t) = \frac{1.2 x(t)^2 y(t)}{0.64 + x(t)^2} - 0.6 y(t)$
>
 $\text{display}(\{\text{phaseportrait}([PD1, PD2], [x(t), y(t)], t = 0..100, [[x(0) = 0.3, y(0) = 0.4]], \text{stepsize} = 1, \text{scene} = [t, x(t)], \text{labels} = ['Waktu', 'Predator - Prey'], \text{linecolour} = [red]), \{\text{phaseportrait}([PD1, PD2], [x(t), y(t)], t = 0..100, [[x(0) = 0.3, y(0) = 0.4]], \text{style} = POINT, \text{stepsize} = 1, \text{scene} = [t, y(t)], \text{labels} = ['Waktu', 'Predator - Prey'], \text{title} = 'Bidang Solusi Sistem pada Predator - Prey Terhadap Waktu', \text{linecolour} = [green]}\});$

```



>  $dfp := dfplot([PD1, PD2], [x(t), y(t)], t = 0..100, x = -0.5..1.5, y = -0.5..1, title = 'Potret Fase Sistem pada Predator – Prey')$  :

>  $fcns := \{x(t), y(t)\}$  :

>  $sistem := PD1, PD2$ ;

$$\begin{aligned} sistem &:= \frac{d}{dt} x(t) = x(t) (1 - x(t)) - \frac{0.8x(t)^2 y(t)}{0.64 + x(t)^2}, \frac{d}{dt} y(t) \\ &= \frac{1.2x(t)^2 y(t)}{0.64 + x(t)^2} - 0.6y(t) \end{aligned}$$

>  $awal1 := dsolve(\{x(0) = 0.2, y(0) = 0.3, sistem\}, fcns, type = numeric, method = classical)$  :

>  $awal2 := dsolve(\{x(0) = 0.3, y(0) = 0.4, sistem\}, fcns, type = numeric, method = classical)$  :

>  $awal3 := dsolve(\{x(0) = 0.1, y(0) = 0.2, sistem\}, fcns, type = numeric, method = classical)$  :

>  $awal4 := dsolve(\{x(0) = 0.25, y(0) = 0.4, sistem\}, fcns, type = numeric, method = classical)$  :

>  $awal5 := dsolve(\{x(0) = 0.3, y(0) = 0.25, sistem\}, fcns, type = numeric, method = classical)$  :

>  $awal6 := dsolve(\{x(0) = 0.4, y(0) = 0.35, sistem\}, fcns, type = numeric, method = classical)$  :

>  $awal7 := dsolve(\{x(0) = 0.5, y(0) = 0.1, sistem\}, fcns, type = numeric, method = classical)$  :

>  $awal8 := dsolve(\{x(0) = 0.2, y(0) = 0.35, sistem\}, fcns, type = numeric, method = classical)$  :

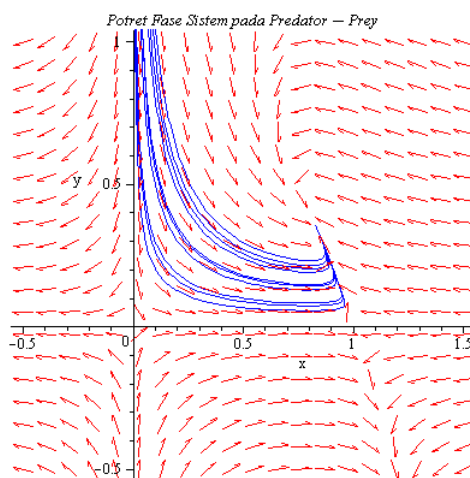
>  $awal9 := dsolve(\{x(0) = 0.4, y(0) = 0.3, sistem\}, fcns, type = numeric, method = classical)$  :

>  $awal10 := dsolve(\{x(0) = 0.15, y(0) = 0.2, sistem\}, fcns, type = numeric, method = classical)$  :

```

> gbrawal1 := odeplot(awal1, [[x(t), y(t)]], -5..15, color = blue) :
> gbrawal2 := odeplot(awal2, [[x(t), y(t)]], -5..15, color = blue) :
> gbrawal3 := odeplot(awal3, [[x(t), y(t)]], -5..15, color = blue) :
> gbrawal4 := odeplot(awal4, [[x(t), y(t)]], -5..15, color = blue) :
> gbrawal5 := odeplot(awal5, [[x(t), y(t)]], -5..15, color = blue) :
> gbrawal6 := odeplot(awal6, [[x(t), y(t)]], -5..15, color = blue) :
> gbrawal7 := odeplot(awal7, [[x(t), y(t)]], -5..15, color = blue) :
> gbrawal8 := odeplot(awal8, [[x(t), y(t)]], -5..15, color = blue) :
> gbrawal9 := odeplot(awal9, [[x(t), y(t)]], -5..15, color = blue) :
> gbrawal10 := odeplot(awal10, [[x(t), y(t)]], -5..15, color = blue) :
> display(gbrawal1, gbrawal2, gbrawal3, gbrawal4, gbrawal5,
 gbrawal6, gbrawal7, gbrawal8, gbrawal9, gbrawal10, dfp);

```



**Titik Ekuilibrium  $E_2$  dengan Syarat  $b > \mu + q$ ,  $m \sqrt{\frac{\mu+q}{b-(\mu+q)}} < 1$ ,  $p_1 > 0$  dan**

$$p_1^2 - 4p_2 < 0$$

```

> restart :
> with(DEtools) :
> with(plots) :
> a := 0.8; b := 1.5; m := 0.3; q := 0.5; mu := 0.1;
 a := 0.8
 b := 1.5
 m := 0.3
 q := 0.5
 mu := 0.1

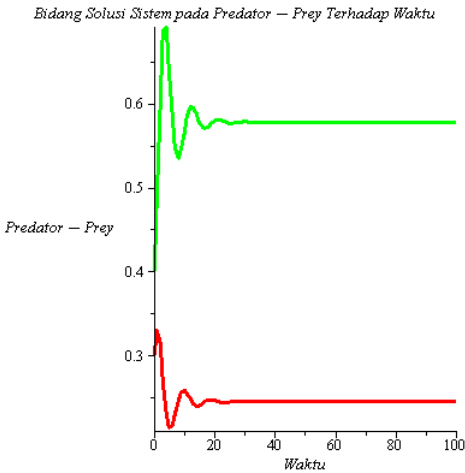
> mu + q;
 0.6

> b > mu + q :

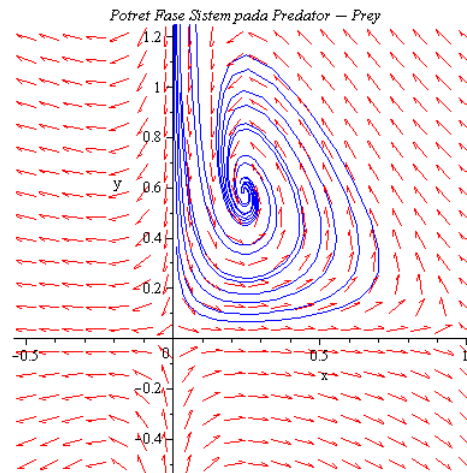
> A := (mu + q) / (b - (mu + q));
 A := 0.6666666666

>
> m * sqrt(A) < 1 :

```

- >  $p1 := 2 \cdot (m \cdot \sqrt{A} - 1) \cdot \left(1 - \frac{1}{1+A}\right) + 1;$   
 $p1 := 0.395959179;$
- >  $p1 > 0;$
- >  $p2 := \frac{2 \cdot b \cdot A \cdot (1 - m \sqrt{A})}{(1+A)^2};$   
 $p2 := 0.543636738;$
- >  $p1^2 - 4 \cdot p2;$   
 $-2.01776328;$
- >  $x2 := m \cdot \sqrt{A}; y2 := m \cdot \frac{(1 - m \sqrt{A}) \cdot (1+A)}{a \cdot \sqrt{A}};$   
 $x2 := 0.244948974;$   
 $y2 := 0.577965544;$
- >  $PD1 := \text{diff}(x(t), t) = x(t) \cdot (1 - x(t)) - \frac{a \cdot x(t)^2 \cdot y(t)}{m^2 + x(t)^2};$   
 $PD1 := \frac{d}{dt} x(t) = x(t) (1 - x(t)) - \frac{0.8 x(t)^2 y(t)}{0.09 + x(t)^2}$
- >
- >  $PD2 := \text{diff}(y(t), t) = \frac{b \cdot x(t)^2 \cdot y(t)}{m^2 + x(t)^2} - (\mu + q) \cdot y(t);$   
 $PD2 := \frac{d}{dt} y(t) = \frac{1.5 x(t)^2 y(t)}{0.09 + x(t)^2} - 0.6 y(t)$
- >
- $\text{display}(\{\text{phaseportrait}([PD1, PD2], [x(t), y(t)], t = 0..100, [[x(0) = 0.3, y(0) = 0.4]], \text{stepsize} = 1, \text{scene} = [t, x(t)], \text{labels} = ['Waktu', 'Predator - Prey'], \text{linecolour} = [\text{red}]]),$   
 $\{\text{phaseportrait}([PD1, PD2], [x(t), y(t)], t = 0..100, [[x(0) = 0.3, y(0) = 0.4]], \text{style} = \text{POINT}, \text{stepsize} = 1, \text{scene} = [t, y(t)],$   
 $\text{labels} = ['Waktu', 'Predator - Prey'], \text{title} =$   
 $'\text{Bidang Solusi Sistem pada Predator - Prey Terhadap Waktu}',$   
 $\text{linecolour} = [\text{green}]]\});$
- 
- >  $dfp := \text{dfieldplot}([PD1, PD2], [x(t), y(t)], t = 0..1000, x = -0.5..1, y = -0.5..1.2, \text{title} = '\text{Potret Fase Sistem pada Predator - Prey}'):$

- > *fcns* := {*x(t)*, *y(t)*} :
  - > *sistem* := *PD1*, *PD2*;
- $$\begin{aligned}
 \text{sistem} &:= \frac{d}{dt} x(t) = x(t) (1 - x(t)) - \frac{0.8 x(t)^2 y(t)}{0.09 + x(t)^2}, \frac{d}{dt} y(t) \\
 &= \frac{1.5 x(t)^2 y(t)}{0.09 + x(t)^2} - 0.6 y(t)
 \end{aligned}$$
- > *awal1* := *dsolve* ({*x(0)* = 0.2, *y(0)* = 0.1, *sistem*}, *fcns*, *type* = *numeric*, *method* = *classical*) :
  - > *awal2* := *dsolve* ({*x(0)* = 0.3, *y(0)* = 0.4, *sistem*}, *fcns*, *type* = *numeric*, *method* = *classical*) :
  - > *awal3* := *dsolve* ({*x(0)* = 0.1, *y(0)* = 0.2, *sistem*}, *fcns*, *type* = *numeric*, *method* = *classical*) :
  - > *awal4* := *dsolve* ({*x(0)* = 0.2, *y(0)* = 0.1, *sistem*}, *fcns*, *type* = *numeric*, *method* = *classical*) :
  - > *awal5* := *dsolve* ({*x(0)* = 0.3, *y(0)* = 0.25, *sistem*}, *fcns*, *type* = *numeric*, *method* = *classical*) :
  - > *awal6* := *dsolve* ({*x(0)* = 0.4, *y(0)* = 0.3, *sistem*}, *fcns*, *type* = *numeric*, *method* = *classical*) :
  - > *awal7* := *dsolve* ({*x(0)* = 0.5, *y(0)* = 0.1, *sistem*}, *fcns*, *type* = *numeric*, *method* = *classical*) :
  - > *awal8* := *dsolve* ({*x(0)* = 0.2, *y(0)* = 0.35, *sistem*}, *fcns*, *type* = *numeric*, *method* = *classical*) :
  - > *awal9* := *dsolve* ({*x(0)* = 0.4, *y(0)* = 0.3, *sistem*}, *fcns*, *type* = *numeric*, *method* = *classical*) :
  - > *awal10* := *dsolve* ({*x(0)* = 0.15, *y(0)* = 0.2, *sistem*}, *fcns*, *type* = *numeric*, *method* = *classical*) :
- > *gbawal1* := *odeplot* (*awal1*, [[*x(t)*, *y(t)*]], -5 .. 15, *color* = *blue*) :
  - > *gbawal2* := *odeplot* (*awal2*, [[*x(t)*, *y(t)*]], -5 .. 15, *color* = *blue*) :
  - > *gbawal3* := *odeplot* (*awal3*, [[*x(t)*, *y(t)*]], -5 .. 15, *color* = *blue*) :
  - > *gbawal4* := *odeplot* (*awal4*, [[*x(t)*, *y(t)*]], -5 .. 15, *color* = *blue*) :
  - > *gbawal5* := *odeplot* (*awal5*, [[*x(t)*, *y(t)*]], -5 .. 15, *color* = *blue*) :
  - > *gbawal6* := *odeplot* (*awal6*, [[*x(t)*, *y(t)*]], -5 .. 15, *color* = *blue*) :
  - > *gbawal7* := *odeplot* (*awal7*, [[*x(t)*, *y(t)*]], -5 .. 15, *color* = *blue*) :
  - > *gbawal8* := *odeplot* (*awal8*, [[*x(t)*, *y(t)*]], -5 .. 15, *color* = *blue*) :
  - > *gbawal9* := *odeplot* (*awal9*, [[*x(t)*, *y(t)*]], -5 .. 15, *color* = *blue*) :
  - > *gbawal10* := *odeplot* (*awal10*, [[*x(t)*, *y(t)*]], -5 .. 15, *color* = *blue*) :
  - > *display* (*gbawal1*, *gbawal2*, *gbawal3*, *gbawal4*, *gbawal5*, *gbawal6*, *gbawal7*, *gbawal8*, *gbawal9*, *gbawal10*, *dfp*) ;



**Print Out Maple 12 Dinamika Populasi Predator-Prey untuk Titik Ekuilibrium Tidak Terjadinya Bifurkasi Hopf**

**Simulasi di Titik Ekuilibrium  $E_0(0,0)$  (Kasus 1),  $E_1\left(0, \frac{m}{\beta}\right)$  (Kasus 1),  $E_H(x_H, 0)$  (Kasus  $h > c$  dengan  $c - \sqrt{A} < 0$ ) dan  $E_L(x_L, 0)$**

```

> restart :
> with(DEtools) :
> with(plots) :
> $\alpha := 0.4; \beta := 1; c := 0.01; h := 0.1; m := 0.1; \rho := 0.1;$
 $\alpha := 0.4$
 $\beta := 1$
 $c := 0.01$
 $h := 0.1$
 $m := 0.1$
 $\rho := 0.1$

> $\frac{h}{c};$
 10.00000000

> $\frac{h}{c} > 1;$
> $\frac{\alpha}{\beta} + \frac{h}{c};$
 10.40000000

> $\frac{\alpha}{\beta} + \frac{h}{c} > 1;$
> $A := (c + 1)^2 - 4 \cdot h;$
 A := 0.6201

> $c - \sqrt{A};$
 -0.777464284

> $c - \sqrt{A} < 0;$

```

```

> A > 0 :
> 2·h - 1/2;
 -0.3000000000

> c > 2·h - 1/2 :
> x0 := 0; y0 := 0;
 x0 := 0
 y0 := 0

> x1 := 0; y1 := m/β;
 x1 := 0
 y1 := 0.1

> xH := (1 - c + √A)/2; yH := 0;
 xH := 0.888732142
 yH := 0

> xL := (1 - c - √A)/2; yH := 0;
 xL := 0.101267857
 yH := 0

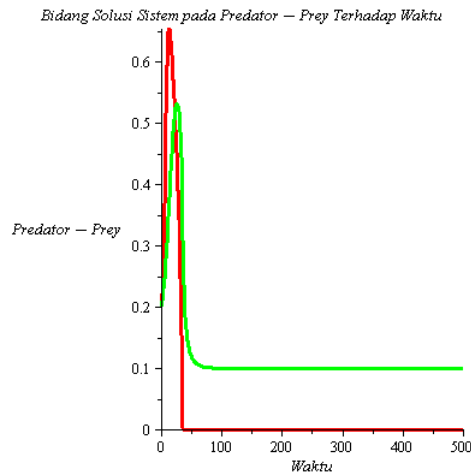
> PD1 := diff(x(t), t) = x(t) · (1 - x(t) - α·y(t)/(m + x(t)) - h/(c + x(t)));
 PD1 := d/dt x(t) = x(t) (1 - x(t) - 0.4y(t)/(0.1 + x(t)) - 0.1/(0.01 + x(t)))

> PD2 := diff(y(t), t) = ρ·y(t) · (1 - β·y(t)/(m + x(t)));
 PD2 := d/dt y(t) = 0.1y(t) (1 - y(t)/(0.1 + x(t)))

> display({phaseportrait([PD1, PD2], [x(t), y(t)], t = 0..500, [[x(0)
= 0.2, y(0) = 0.2]], stepsize = 0.8, scene = [t, x(t)], labels = [
'Waktu', 'Predator - Prey'], linecolour = [red]}),
{phaseportrait([PD1, PD2], [x(t), y(t)], t = 0..500, [[x(0)
= 0.2, y(0) = 0.2]], style = POINT, stepsize = 0.8, scene = [t,
y(t)], labels = ['Waktu', 'Predator - Prey'], title =
'Bidang Solusi Sistem pada Predator - Prey Terhadap Waktu',
linecolour = [green]}});

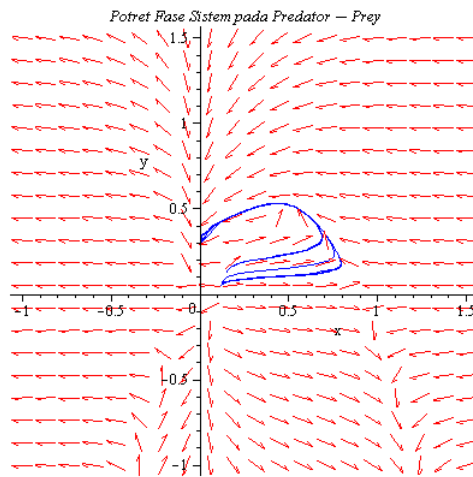
```





- >  $dfp := dfieldplot([PD1, PD2], [x(t), y(t)], t = 0 .. 100, x = -1 .. 1.5, y = -1 .. 1.5, title = 'Potret Fase Sistem pada Predator – Prey')$  :
  - >  $fcns := \{x(t), y(t)\}$  :
  - >  $sistem := PD1, PD2$ ;
- $$sistem := \frac{d}{dt} x(t) = x(t) \left( 1 - x(t) - \frac{0.4y(t)}{0.1 + x(t)} - \frac{0.1}{0.01 + x(t)} \right),$$
- $$\frac{d}{dt} y(t) = 0.1y(t) \left( 1 - \frac{y(t)}{0.1 + x(t)} \right)$$
- >  $awal1 := dsolve(\{x(0) = 0.3, y(0) = 0.2, sistem\}, fcns, type = numeric, method = classical)$  :
  - >  $awal2 := dsolve(\{x(0) = 0.4, y(0) = 0.1, sistem\}, fcns, type = numeric, method = classical)$  :
  - >  $awal3 := dsolve(\{x(0) = 0.35, y(0) = 0.1, sistem\}, fcns, type = numeric, method = classical)$  :
  - >  $awal4 := dsolve(\{x(0) = 0.3, y(0) = 0.2, sistem\}, fcns, type = numeric, method = classical)$  :
  - >  $awal5 := dsolve(\{x(0) = 0.4, y(0) = 0.1, sistem\}, fcns, type = numeric, method = classical)$  :
  - >  $awal6 := dsolve(\{x(0) = 0.4, y(0) = 0.1, sistem\}, fcns, type = numeric, method = classical)$  :
  - >  $awal7 := dsolve(\{x(0) = 0.3, y(0) = 0.2, sistem\}, fcns, type = numeric, method = classical)$  :
  - >  $awal8 := dsolve(\{x(0) = 0, y(0) = 0.1, sistem\}, fcns, type = numeric, method = classical)$  :
  - >  $awal9 := dsolve(\{x(0) = 0.4, y(0) = 0.1, sistem\}, fcns, type = numeric, method = classical)$  :
  - >  $awal10 := dsolve(\{x(0) = 0.15, y(0) = 0.1, sistem\}, fcns, type = numeric, method = classical)$  :
  - >  $gbawal1 := odeplot(awal1, [[x(t), y(t)]], -10 .. 100, color = blue)$  :

- > `gbrawal2 := odeplot(awal2, [[x(t), y(t)]], -10..70, color = blue) :`
- > `gbrawal3 := odeplot(awal3, [[x(t), y(t)]], -10..60, color = blue) :`
- > `gbrawal4 := odeplot(awal4, [[x(t), y(t)]], -10..65, color = blue) :`
- > `gbrawal5 := odeplot(awal5, [[x(t), y(t)]], -10..60, color = blue) :`
- > `gbrawal6 := odeplot(awal6, [[x(t), y(t)]], -10..60, color = blue) :`
- > `gbrawal7 := odeplot(awal7, [[x(t), y(t)]], -10..70, color = blue) :`
- > `gbrawal8 := odeplot(awal8, [[x(t), y(t)]], -10..65, color = blue) :`
- > `gbrawal9 := odeplot(awal9, [[x(t), y(t)]], -10..60, color = blue) :`
- > `gbrawal10 := odeplot(awal10, [[x(t), y(t)]], -10..70, color = blue) :`
- > `display(gbrawal1, gbrawal2, gbrawal3, gbrawal4, gbrawal5, gbrawal6, gbrawal7, gbrawal8, gbrawal9, gbrawal10, dfp);`



**Simulasi di Titik Ekuilibrium  $E_0(0, 0)$  (Kasus 2),  $E_1\left(0, \frac{m}{\beta}\right)$  (Kasus 2) dan  $E_H(x_H, 0)$  (Kasus  $h < c$  dengan  $c - \sqrt{A} < 0$ )**

- > `restart :`
- > `with(DEtools) :`
- > `with(plots) :`
- >  `$\alpha := 0.4; \beta := 1; c := 0.9; h := 0.1; m := 0.1; \rho := 0.1;$`

`$\alpha := 0.4$`

`$\beta := 1$`

`$c := 0.9$`

`$h := 0.1$`

`$m := 0.1$`

`$\rho := 0.1$`

- >  `$\frac{h}{c};$`

`0.1111111111`

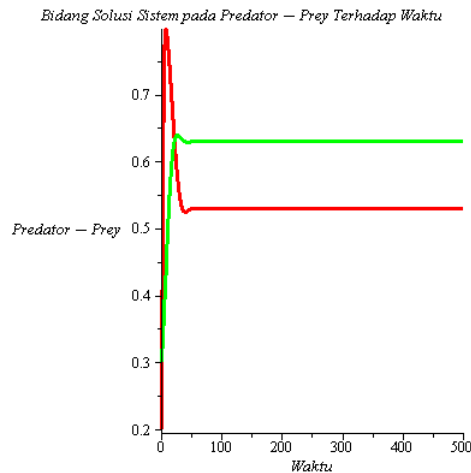
- >  `$\frac{h}{c} < 1;$`

- >  `$\frac{\alpha}{\beta} + \frac{h}{c};$`

```

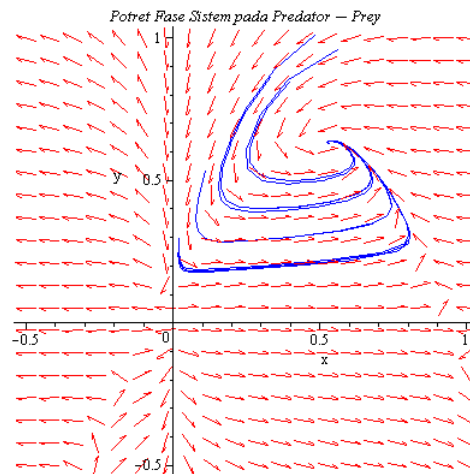
0.5111111111
> $\frac{\alpha}{\beta} + \frac{h}{c} < 1$:
> $A := (c + 1)^2 - 4 \cdot h$;
 A := 3.21
> $A > 0$:
> $c - \sqrt{A}$;
 -0.89164728'
> $c - \sqrt{A} < 0$:
> $2 \cdot h - \frac{1}{2}$;
 -0.3000000000'
> $c > 2h - \frac{1}{2}$:
> $xH := \frac{1 - c + \sqrt{A}}{2}$;
 xH := 0.945823643:
> $yH := 0$;
 yH := 0
> $PD1 := \text{diff}(x(t), t) = x(t) \cdot \left(1 - x(t) - \frac{\alpha \cdot y(t)}{m + x(t)} - \frac{h}{c + x(t)} \right)$;
 $PD1 := \frac{d}{dt} x(t) = x(t) \left(1 - x(t) - \frac{0.4 y(t)}{0.1 + x(t)} - \frac{0.1}{0.9 + x(t)} \right)$
> $PD2 := \text{diff}(y(t), t) = \rho \cdot y(t) \cdot \left(1 - \frac{\beta \cdot y(t)}{m + x(t)} \right)$;
 $PD2 := \frac{d}{dt} y(t) = 0.1 y(t) \left(1 - \frac{y(t)}{0.1 + x(t)} \right)$
>
display({phaseportrait([PD1, PD2], [x(t), y(t)], t = 0..500, [[x(0)
= 0.2, y(0) = 0.2]], stepsize = 1, scene = [t, x(t)], labels = [
'Waktu', 'Predator - Prey'], linecolour = [red]]),
{phaseportrait([PD1, PD2], [x(t), y(t)], t = 0..500, [[x(0)
= 0.3, y(0) = 0.3]], style = POINT, stepsize = 1, scene = [t, y(t)],
labels = ['Waktu', 'Predator - Prey'], title =
'Bidang Solusi Sistem pada Predator - Prey Terhadap Waktu',
linecolour = [green]])});

```



- >  $dfp := dfieldplot([PD1, PD2], [x(t), y(t)], t = 0 .. 100, x = -0.5 .. 1, y = -0.5 .. 1, title = 'Potret Fase Sistem pada Predator – Prey')$  :
  - >  $fcns := \{x(t), y(t)\}$  :
  - >  $sistem := PD1, PD2$ ;
- $$sistem := \frac{d}{dt} x(t) = x(t) \left( 1 - x(t) - \frac{0.4 y(t)}{0.1 + x(t)} - \frac{0.1}{0.9 + x(t)} \right),$$
- $$\frac{d}{dt} y(t) = 0.1 y(t) \left( 1 - \frac{y(t)}{0.1 + x(t)} \right)$$
- >  $awal1 := dsolve(\{x(0) = 0.3, y(0) = 0.5, sistem\}, fcns, type = numeric, method = classical)$  :
  - >  $awal2 := dsolve(\{x(0) = 0.3, y(0) = 0.2, sistem\}, fcns, type = numeric, method = classical)$  :
  - >  $awal3 := dsolve(\{x(0) = 0.4, y(0) = 0.3, sistem\}, fcns, type = numeric, method = classical)$  :
  - >  $awal4 := dsolve(\{x(0) = 0.3, y(0) = 0.4, sistem\}, fcns, type = numeric, method = classical)$  :
  - >  $awal5 := dsolve(\{x(0) = 0.4, y(0) = 0.5, sistem\}, fcns, type = numeric, method = classical)$  :
  - >  $awal6 := dsolve(\{x(0) = 0.2, y(0) = 0.4, sistem\}, fcns, type = numeric, method = classical)$  :
  - >  $awal7 := dsolve(\{x(0) = 0.3, y(0) = 0.4, sistem\}, fcns, type = numeric, method = classical)$  :
  - >  $awal8 := dsolve(\{x(0) = 0.4, y(0) = 0.3, sistem\}, fcns, type = numeric, method = classical)$  :
  - >  $awal9 := dsolve(\{x(0) = 0.35, y(0) = 0.2, sistem\}, fcns, type = numeric, method = classical)$  :
  - >  $awal10 := dsolve(\{x(0) = 0.4, y(0) = 0.2, sistem\}, fcns, type = numeric, method = classical)$  :
  - >  $gbawal1 := odeplot(awal1, [[x(t), y(t)]], -10 .. 100, color = blue)$  :

- > `gbrawal2 := odeplot(awal2, [[x(t), y(t)]], -10..70, color = blue) :`
- > `gbrawal3 := odeplot(awal3, [[x(t), y(t)]], -10..60, color = blue) :`
- > `gbrawal4 := odeplot(awal4, [[x(t), y(t)]], -10..65, color = blue) :`
- > `gbrawal5 := odeplot(awal5, [[x(t), y(t)]], -10..60, color = blue) :`
- > `gbrawal6 := odeplot(awal6, [[x(t), y(t)]], -10..60, color = blue) :`
- > `gbrawal7 := odeplot(awal7, [[x(t), y(t)]], -10..70, color = blue) :`
- > `gbrawal8 := odeplot(awal8, [[x(t), y(t)]], -10..65, color = blue) :`
- > `gbrawal9 := odeplot(awal9, [[x(t), y(t)]], -10..60, color = blue) :`
- > `gbrawal10 := odeplot(awal10, [[x(t), y(t)]], -10..70, color = blue) :`
- > `display(gbrawal1, gbrawal2, gbrawal3, gbrawal4, gbrawal5, gbrawal6, gbrawal7, gbrawal8, gbrawal9, gbrawal10, dfp);`



### Simulasi di Titik Ekuilibrium $E_H(x_H, 0)$ Kasus $h < c$ dengan $c - \sqrt{A} > 0$

- > `restart :`
- > `with(DEtools) :`
- > `with(plots) :`
- >  `$\alpha := 0.4; \beta := 1; c := 0.55; h := 0.52; m := 0.1; \rho := 0.1;$` 
  - `$\alpha := 0.4$`
  - `$\beta := 1$`
  - `$c := 0.55$`
  - `$h := 0.52$`
  - `$m := 0.1$`
  - `$\rho := 0.1$`
- >  `$A := (c + 1)^2 - 4 \cdot h;$` 
  - `$A := 0.3225$`
- >  `$c - \sqrt{A};$` 
  - `$-0.017890834$`
- >  `$c - \sqrt{A} > 0 :$`
- >  `$A > 0 :$`
- >  `$2 \cdot h - \frac{1}{2};$` 
  - `$0.540000000$`

$$> c < 2 \cdot h - \frac{1}{2} :$$

$$> xH := \frac{1 - c + \sqrt{A}}{2}; yH := 0;$$

$$xH := 0.508945417.$$

$$yH := 0$$

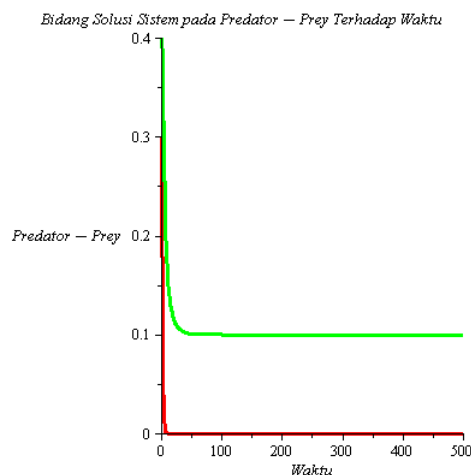
$$> PD1 := \text{diff}(x(t), t) = x(t) \cdot \left( 1 - x(t) - \frac{\alpha \cdot y(t)}{m + x(t)} - \frac{h}{c + x(t)} \right);$$

$$PD1 := \frac{d}{dt} x(t) = x(t) \left( 1 - x(t) - \frac{0.4 y(t)}{0.1 + x(t)} - \frac{0.52}{0.55 + x(t)} \right)$$

$$> PD2 := \text{diff}(y(t), t) = \rho \cdot y(t) \cdot \left( 1 - \frac{\beta \cdot y(t)}{m + x(t)} \right);$$

$$PD2 := \frac{d}{dt} y(t) = 0.1 y(t) \left( 1 - \frac{y(t)}{0.1 + x(t)} \right)$$

> `display({phaseportrait([PD1, PD2], [x(t), y(t)], t = 0..500, [[x(0) = 0.3, y(0) = 0.4]], stepsize = 0.8, scene = [t, x(t)], labels = ['Waktu', 'Predator - Prey'], linecolour = [red]}), {phaseportrait([PD1, PD2], [x(t), y(t)], t = 0..500, [[x(0) = 0.3, y(0) = 0.4]], style = POINT, stepsize = 0.8, scene = [t, y(t)], labels = ['Waktu', 'Predator - Prey'], title = 'Bidang Solusi Sistem pada Predator - Prey Terhadap Waktu', linecolour = [green]})};`



> `dfp := dfieldplot([PD1, PD2], [x(t), y(t)], t = 0..100, x = -1..1.5, y = -1..1.5, title = 'Potret Fase Sistem pada Predator - Prey') :`

> `fcns := {x(t), y(t)} :`

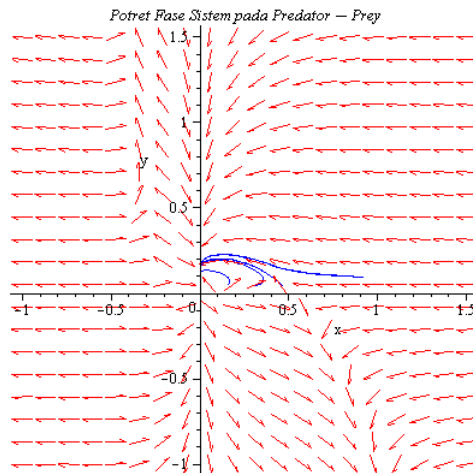
> `sistem := PD1, PD2;`

$$sistem := \frac{d}{dt} x(t) = x(t) \left( 1 - x(t) - \frac{0.4 y(t)}{0.1 + x(t)} - \frac{0.52}{0.55 + x(t)} \right),$$

$$\frac{d}{dt} y(t) = 0.1 y(t) \left( 1 - \frac{y(t)}{0.1 + x(t)} \right)$$

> `awall := dsolve({x(0) = 0.3, y(0) = 0.2, sistem}, fcns, type = numeric, method = classical) :`

- > *awal2 := dsolve ({x(0) = 0.4, y(0) = 0.1, sistem}, fcns, type = numeric, method = classical) :*
- > *awal3 := dsolve ({x(0) = 0.35, y(0) = 0.1, sistem}, fcns, type = numeric, method = classical) :*
- > *awal4 := dsolve ({x(0) = 0.3, y(0) = 0.2, sistem}, fcns, type = numeric, method = classical) :*
- > *awal5 := dsolve ({x(0) = 0.4, y(0) = 0.1, sistem}, fcns, type = numeric, method = classical) :*
- > *awal6 := dsolve ({x(0) = 0.4, y(0) = 0.1, sistem}, fcns, type = numeric, method = classical) :*
- > *awal7 := dsolve ({x(0) = 0.3, y(0) = 0.2, sistem}, fcns, type = numeric, method = classical) :*
- > *awal8 := dsolve ({x(0) = 0, y(0) = 0.1, sistem}, fcns, type = numeric, method = classical) :*
- > *awal9 := dsolve ({x(0) = 0.4, y(0) = 0.1, sistem}, fcns, type = numeric, method = classical) :*
- > *awal10 := dsolve ({x(0) = 0.15, y(0) = 0.1, sistem}, fcns, type = numeric, method = classical) :*
  
- > *gbawal1 := odeplot (awal1, [[x(t), y(t)]], -10..100, color = blue) :*
- > *gbawal2 := odeplot (awal2, [[x(t), y(t)]], -10..70, color = blue) :*
- > *gbawal3 := odeplot (awal3, [[x(t), y(t)]], -10..60, color = blue) :*
- > *gbawal4 := odeplot (awal4, [[x(t), y(t)]], -10..65, color = blue) :*
- > *gbawal5 := odeplot (awal5, [[x(t), y(t)]], -10..60, color = blue) :*
- > *gbawal6 := odeplot (awal6, [[x(t), y(t)]], -10..60, color = blue) :*
- > *gbawal7 := odeplot (awal7, [[x(t), y(t)]], -10..70, color = blue) :*
- > *gbawal8 := odeplot (awal8, [[x(t), y(t)]], -10..65, color = blue) :*
- > *gbawal9 := odeplot (awal9, [[x(t), y(t)]], -10..60, color = blue) :*
- > *gbawal10 := odeplot (awal10, [[x(t), y(t)]], -10..70, color = blue) :*
  
- > *display (gbawal1, gbawal2, gbawal3, gbawal4, gbawal5, gbawal6, gbawal7, gbawal8, gbawal9, gbawal10, dfp) ;*



**Simulasi di Titik Ekuilibrium  $E_H(x_H, 0)$  Kasus  $h > c$  dengan  $c - \sqrt{A} > 0$**

```

> restart :
> with(DEtools) :
> with(plots) :
> $\alpha := 0.4; \beta := 1; c := 0.49; h := 0.5; m := 0.1; \rho := 0.1;$
 $\alpha := 0.4$
 $\beta := 1$
 $c := 0.49$
 $h := 0.5$
 $m := 0.1$
 $\rho := 0.1$

> $A := (c + 1)^2 - 4 \cdot h;$
 $A := 0.2201$

> $c - \sqrt{A};$
 0.020851835;

> $c - \sqrt{A} > 0;$
> $A > 0;$
> $2 \cdot h - \frac{1}{2};$
 0.5000000000

> $c < 2 \cdot h - \frac{1}{2};$
> $x0 := 0; y0 := 0;$
 $x0 := 0$
 $y0 := 0$

> $x1 := 0; y1 := \frac{m}{\beta};$
 $x1 := 0$
 $y1 := 0.1$

> $xH := \frac{1 - c + \sqrt{A}}{2}; yH := 0;$
 $xH := 0.489574082$
 $yH := 0$

```



&gt;

$$PD1 := \text{diff}(x(t), t) = x(t) \cdot \left( 1 - x(t) - \frac{\alpha \cdot y(t)}{m + x(t)} - \frac{h}{c + x(t)} \right);$$

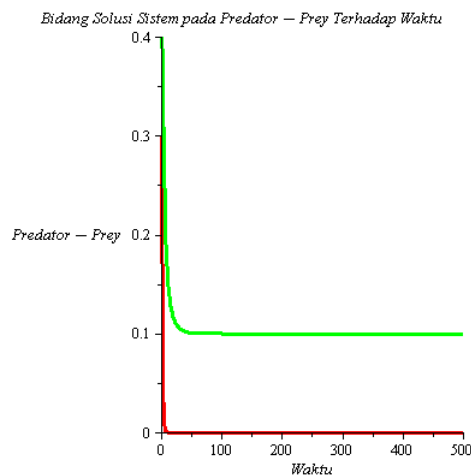
$$PD1 := \frac{d}{dt} x(t) = x(t) \left( 1 - x(t) - \frac{0.4 y(t)}{0.1 + x(t)} - \frac{0.5}{0.49 + x(t)} \right)$$

$$PD2 := \text{diff}(y(t), t) = \rho \cdot y(t) \cdot \left( 1 - \frac{\beta \cdot y(t)}{m + x(t)} \right);$$

$$PD2 := \frac{d}{dt} y(t) = 0.1 y(t) \left( 1 - \frac{y(t)}{0.1 + x(t)} \right)$$

&gt;

```
display({phaseportrait([PD1, PD2], [x(t), y(t)], t = 0..500, [[x(0)
= 0.3, y(0) = 0.4]], stepsize = 0.8, scene = [t, x(t)], labels = [
'Waktu', 'Predator - Prey'], linecolour = [red])},
{phaseportrait([PD1, PD2], [x(t), y(t)], t = 0..500, [[x(0)
= 0.3, y(0) = 0.4]], style = POINT, stepsize = 0.8, scene = [t,
y(t)], labels = ['Waktu', 'Predator - Prey'], title =
'Bidang Solusi Sistem pada Predator - Prey Terhadap Waktu',
linecolour = [green])});
```



```
> dfp := dfieldplot([PD1, PD2], [x(t), y(t)], t = 0..1000, x = -1..2.5, y
= -1..2.5, title = 'Potret Fase Sistem pada Predator - Prey');
```

```
> fcns := {x(t), y(t)};
```

```
> sistem := PD1, PD2;
```

$$\text{sistem} := \frac{d}{dt} x(t) = x(t) \left( 1 - x(t) - \frac{0.4 y(t)}{0.1 + x(t)} - \frac{0.5}{0.49 + x(t)} \right),$$

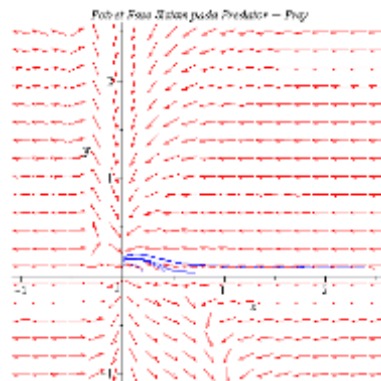
$$\frac{d}{dt} y(t) = 0.1 y(t) \left( 1 - \frac{y(t)}{0.1 + x(t)} \right)$$

```
> awal1 := dsolve({x(0) = 0.3, y(0) = 0.2, sistem}, fcns, type
= numeric, method = classical);
```

```
> awal2 := dsolve({x(0) = 0.4, y(0) = 0.1, sistem}, fcns, type
= numeric, method = classical);
```

```
> awal3 := dsolve({x(0) = 0.35, y(0) = 0.1, sistem}, fcns, type
= numeric, method = classical);
```

- > `awal4 := dsolve ({x(0) = 0.3, y(0) = 0.2, sistem}, fcns, type = numeric, method = classical) :`
- > `awal5 := dsolve ({x(0) = 0.4, y(0) = 0.1, sistem}, fcns, type = numeric, method = classical) :`
- > `awal6 := dsolve ({x(0) = 0.4, y(0) = 0.1, sistem}, fcns, type = numeric, method = classical) :`
- > `awal7 := dsolve ({x(0) = 0.3, y(0) = 0.2, sistem}, fcns, type = numeric, method = classical) :`
- > `awal8 := dsolve ({x(0) = 0, y(0) = 0.1, sistem}, fcns, type = numeric, method = classical) :`
- > `awal9 := dsolve ({x(0) = 0.4, y(0) = 0.1, sistem}, fcns, type = numeric, method = classical) :`
- > `awal10 := dsolve ({x(0) = 0.15, y(0) = 0.1, sistem}, fcns, type = numeric, method = classical) :`
  
- > `gbrawal1 := odeplot (awal1, [[x(t), y(t)]], -10..30, color = blue) :`
- > `gbrawal2 := odeplot (awal2, [[x(t), y(t)]], -10..30, color = blue) :`
- > `gbrawal3 := odeplot (awal3, [[x(t), y(t)]], -10..30, color = blue) :`
- > `gbrawal4 := odeplot (awal4, [[x(t), y(t)]], -10..35, color = blue) :`
- > `gbrawal5 := odeplot (awal5, [[x(t), y(t)]], -10..30, color = blue) :`
- > `gbrawal6 := odeplot (awal6, [[x(t), y(t)]], -10..30, color = blue) :`
- > `gbrawal7 := odeplot (awal7, [[x(t), y(t)]], -10..30, color = blue) :`
- > `gbrawal8 := odeplot (awal8, [[x(t), y(t)]], -10..35, color = blue) :`
- > `gbrawal9 := odeplot (awal9, [[x(t), y(t)]], -10..30, color = blue) :`
- > `gbrawal10 := odeplot (awal10, [[x(t), y(t)]], -10..30, color = blue) :`
  
- > `display (gbrawal1, gbrawal2, gbrawal3, gbrawal4, gbrawal5, gbrawal6, gbrawal7, gbrawal8, gbrawal9, gbrawal10, dfp) ;`



**Print Out Maple 12 Dinamika Populasi Predator-Prey untuk Titik  
Ekuilibrum Terjadinya Bifurkasi Hopf**

**Titik Ekuilibrum  $E_*$  Untuk Kasus  $\frac{s_0}{s} > 1$**

```

> restart :
> with(DEtools) :
> with(plots) :
> $\alpha := 0.4; \beta := 1; c := 0.05; h := 0.1; m := 0.1;$
 $\alpha := 0.4$
 $\beta := 1$
 $c := 0.05$
 $h := 0.1$
 $m := 0.1$
> $B := c^2 + \left(-\frac{2 \cdot \alpha}{\beta} + 2\right) \cdot c + \frac{\alpha^2}{\beta^2} - 2 \cdot \frac{\alpha}{\beta} - 4 \cdot h + 1;$
 $B := 0.0225$
> $xbintang := \frac{1 - \frac{\alpha}{\beta} - c + \sqrt{B}}{2};$
 $xbintang := 0.3500000000$
> $ybintang := \frac{2 \cdot m - \frac{\alpha}{\beta} - c + \sqrt{B} + 1}{2 \cdot \beta};$
 $ybintang := 0.4500000000$
> $s0 := c - \sqrt{B} + \frac{-\alpha^2 + \beta \cdot \alpha \cdot (-c + \sqrt{B} + 1)}{-\alpha \cdot \beta + \beta^2 \cdot (2 \cdot m - c + \sqrt{B} + 1)}$

$$- \frac{2 \cdot h \cdot c}{\left(-\frac{\alpha}{\beta} + c + 1\right)^2 + \sqrt{B} \cdot \left(c - \frac{\alpha}{\beta} + 1\right) - 2 \cdot h};$$

 $s0 := 0.179861111$
> $\frac{s0}{s} = 1.033684547 > 1 :$
> $s := \frac{s0}{1.033684547};$
 $s := 0.1739999999$
> $\rho := s;$
 $\rho := 0.1739999999$

```

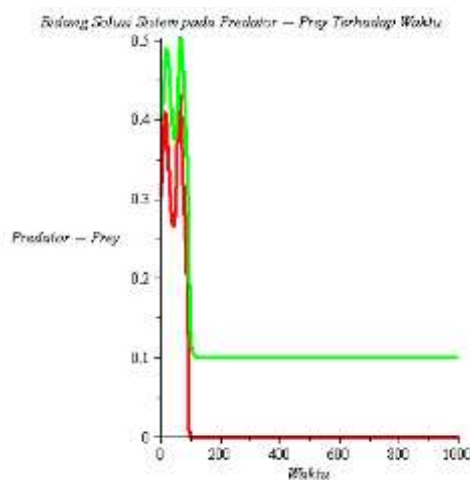
$$\begin{aligned}
 > x = \frac{1}{2} \cdot \left( 1 - c - \frac{\alpha}{\beta} \right) + \frac{1}{2} \\
 & \quad \cdot \sqrt{\left( 1 - c - \frac{\alpha}{\beta} \right)^2 - 4 \cdot c \cdot \left( \frac{\alpha}{\beta} + \frac{h}{c} - 1 \right)}; \\
 & \quad x = 0.3500000000
 \end{aligned}$$

$$\begin{aligned}
 > y = \frac{1}{\beta} \left( m + \left( \frac{1}{2} \cdot \left( 1 - c - \frac{\alpha}{\beta} \right) + \frac{1}{2} \right. \right. \\
 & \quad \left. \left. \cdot \sqrt{\left( 1 - c - \frac{\alpha}{\beta} \right)^2 - 4 \cdot c \cdot \left( \frac{\alpha}{\beta} + \frac{h}{c} - 1 \right)} \right) \right); \\
 & \quad y = 0.4500000000
 \end{aligned}$$

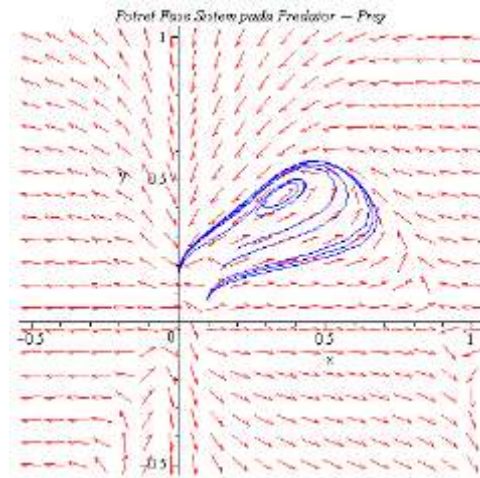
$$\begin{aligned}
 > PD1 := \text{diff}(x(t), t) = x(t) \cdot \left( 1 - x(t) - \frac{\alpha \cdot y(t)}{m + x(t)} - \frac{h}{c + x(t)} \right); \\
 & \quad PD1 := \frac{d}{dt} x(t) = x(t) \left( 1 - x(t) - \frac{0.4 y(t)}{0.1 + x(t)} - \frac{0.1}{0.05 + x(t)} \right)
 \end{aligned}$$

$$\begin{aligned}
 > PD2 := \text{diff}(y(t), t) = \rho \cdot y(t) \cdot \left( 1 - \frac{\beta \cdot y(t)}{m + x(t)} \right); \\
 & \quad PD2 := \frac{d}{dt} y(t) = 0.1739999999 y(t) \left( 1 - \frac{y(t)}{0.1 + x(t)} \right)
 \end{aligned}$$

> `display( {phaseportrait([PD1, PD2], [x(t), y(t)], t = 0..1000, [[x(0) = 0.3, y(0) = 0.4]], stepsize = 0.3, scene = [t, x(t)], labels = ['Waktu', 'Predator - Prey'], linecolour = [red]}), {phaseportrait([PD1, PD2], [x(t), y(t)], t = 0..1000, [[x(0) = 0.3, y(0) = 0.4]], style = POINT, stepsize = 0.3, scene = [t, y(t)], labels = ['Waktu', 'Predator - Prey'], title = 'Bidang Solusi Sistem pada Predator - Prey Terhadap Waktu', linecolour = [green]})};`



- >  $dfp := dfieldplot([PD1, PD2], [x(t), y(t)], t = 0..100, x = -.5..1, y = -.50..1, title = 'Potret Fase Sistem pada Predator - Prey')$  :
- >  $fcns := \{x(t), y(t)\}$  :
- >  $sistem := PD1, PD2$  ;
- $sistem := \frac{d}{dt} x(t) = x(t) \left( 1 - x(t) - \frac{0.4 y(t)}{0.1 + x(t)} - \frac{0.1}{0.05 + x(t)} \right)$  ,
- $\frac{d}{dt} y(t) = 0.1739999999 y(t) \left( 1 - \frac{y(t)}{0.1 + x(t)} \right)$
- >  $awal1 := dsolve(\{x(0) = 0.3, y(0) = 0.4, sistem\}, fcns, type = numeric, method = classical)$  :
- >  $awal2 := dsolve(\{x(0) = 0.4, y(0) = 0.2, sistem\}, fcns, type = numeric, method = classical)$  :
- >  $awal3 := dsolve(\{x(0) = 0.5, y(0) = 0.4, sistem\}, fcns, type = numeric, method = classical)$  :
- >  $awal4 := dsolve(\{x(0) = 0.35, y(0) = 0.2, sistem\}, fcns, type = numeric, method = classical)$  :
- >  $awal5 := dsolve(\{x(0) = 0.4, y(0) = 0.5, sistem\}, fcns, type = numeric, method = classical)$  :
- >  $awal6 := dsolve(\{x(0) = 0.4, y(0) = 0.3, sistem\}, fcns, type = numeric, method = classical)$  :
- >  $awal7 := dsolve(\{x(0) = 0.2, y(0) = 0.3, sistem\}, fcns, type = numeric, method = classical)$  :
- >  $awal8 := dsolve(\{x(0) = 0.4, y(0) = 0.3, sistem\}, fcns, type = numeric, method = classical)$  :
- >  $awal9 := dsolve(\{x(0) = 0.2, y(0) = 0.4, sistem\}, fcns, type = numeric, method = classical)$  :
- >  $awal10 := dsolve(\{x(0) = 0.3, y(0) = 0.2, sistem\}, fcns, type = numeric, method = classical)$  :
- >  $gbawal1 := odeplot(awal1, [[x(t), y(t)]], -10..70, color = blue)$  :
- >  $gbawal2 := odeplot(awal2, [[x(t), y(t)]], -10..60, color = blue)$  :
- >  $gbawal3 := odeplot(awal3, [[x(t), y(t)]], -10..65, color = blue)$  :
- >  $gbawal4 := odeplot(awal4, [[x(t), y(t)]], -10..60, color = blue)$  :
- >  $gbawal5 := odeplot(awal5, [[x(t), y(t)]], -10..50, color = blue)$  :
- >  $gbawal6 := odeplot(awal6, [[x(t), y(t)]], -10..60, color = blue)$  :
- >  $gbawal7 := odeplot(awal7, [[x(t), y(t)]], -10..65, color = blue)$  :
- >  $gbawal8 := odeplot(awal8, [[x(t), y(t)]], -10..60, color = blue)$  :
- >  $gbawal9 := odeplot(awal9, [[x(t), y(t)]], -10..70, color = blue)$  :
- >  $gbawal10 := odeplot(awal10, [[x(t), y(t)]], -10..80, color = blue)$  :
- >  $display(gbawal1, gbawal2, gbawal3, gbawal4, gbawal5, gbawal6, gbawal7, gbawal8, gbawal9, gbawal10, dfp)$  ;



### Titik Ekuilibrium $E_*$ Untuk Kasus $\frac{s_0}{s} = 1$

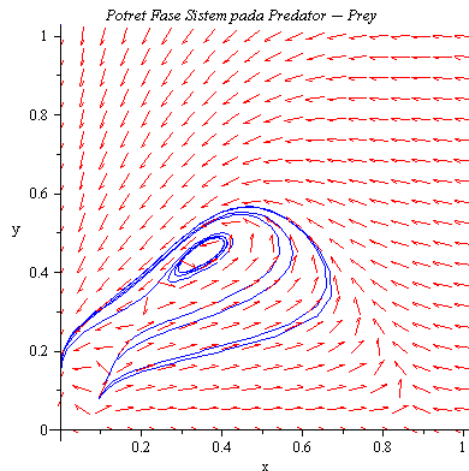
- > restart :
- > with(DEtools) :
- > with(plots) :
- >  $\alpha := 0.4; \beta := 1; c := 0.05; h := 0.1; m := 0.1;$ 
  - $\alpha := 0.4$
  - $\beta := 1$
  - $c := 0.05$
  - $h := 0.1$
  - $m := 0.1$
- >  $B := c^2 + \left(-\frac{2 \cdot \alpha}{\beta} + 2\right) \cdot c + \frac{\alpha^2}{\beta^2} - 2 \cdot \frac{\alpha}{\beta} - 4 \cdot h + 1;$ 
  - $B := 0.0225$
- >  $xbintang := \frac{1 - \frac{\alpha}{\beta} - c + \sqrt{B}}{2};$ 
  - $xbintang := 0.3500000000$
- >  $ybintang := \frac{2 \cdot m - \frac{\alpha}{\beta} - c + \sqrt{B} + 1}{2 \cdot \beta};$ 
  - $ybintang := 0.4500000000$
- >  $s0 := c - \sqrt{B} + \frac{-\alpha^2 + \beta \cdot \alpha \cdot (-c + \sqrt{B} + 1)}{-\alpha \cdot \beta + \beta^2 \cdot (2 \cdot m - c + \sqrt{B} + 1)}$ 

$$- \frac{2 \cdot h \cdot c}{\left(-\frac{\alpha}{\beta} + c + 1\right)^2 + \sqrt{B} \cdot \left(c - \frac{\alpha}{\beta} + 1\right) - 2 \cdot h};$$
  - $s0 := 0.1798611111$
- >  $\frac{s0}{s} = 1 :$
- >  $s := s0;$

- $s := 0.179861111$
- >  $\rho := s;$   $\rho := 0.179861111$
- >  $x = \frac{1}{2} \cdot \left( 1 - c - \frac{\alpha}{\beta} \right) + \frac{1}{2} \cdot \sqrt{\left( 1 - c - \frac{\alpha}{\beta} \right)^2 - 4 \cdot c \cdot \left( \frac{\alpha}{\beta} + \frac{h}{c} - 1 \right)};$   
 $x = 0.350000000$
- >  $y = \frac{1}{\beta} \left( m + \left( \frac{1}{2} \cdot \left( 1 - c - \frac{\alpha}{\beta} \right) + \frac{1}{2} \cdot \sqrt{\left( 1 - c - \frac{\alpha}{\beta} \right)^2 - 4 \cdot c \cdot \left( \frac{\alpha}{\beta} + \frac{h}{c} - 1 \right)} \right) \right);$   
 $y = 0.450000000$
- >  $PD1 := \text{diff}(x(t), t) = x(t) \cdot \left( 1 - x(t) - \frac{\alpha \cdot y(t)}{m + x(t)} - \frac{h}{c + x(t)} \right);$   
 $PD1 := \frac{d}{dt} x(t) = x(t) \left( 1 - x(t) - \frac{0.4 \cdot y(t)}{0.1 + x(t)} - \frac{0.1}{0.05 + x(t)} \right)$
- >  $PD2 := \text{diff}(y(t), t) = \rho \cdot y(t) \cdot \left( 1 - \frac{\beta \cdot y(t)}{m + x(t)} \right);$   
 $PD2 := \frac{d}{dt} y(t) = 0.179861111 y(t) \left( 1 - \frac{y(t)}{0.1 + x(t)} \right)$
- >  $\text{display}(\{ \text{phaseportrait}([PD1, PD2], [x(t), y(t)], t = 0..1000, [[x(0) = 0.3, y(0) = 0.4]], \text{stepsize} = 1, \text{scene} = [t, x(t)], \text{labels} = ['Waktu', 'Predator - Prey'], \text{linecolour} = [red] \}, \{ \text{phaseportrait}([PD1, PD2], [x(t), y(t)], t = 0..1000, [[x(0) = 0.3, y(0) = 0.4]], \text{style} = POINT, \text{stepsize} = 1, \text{scene} = [t, y(t)], \text{labels} = ['Waktu', 'Predator - Prey'], \text{title} = 'Bidang Solusi Sistem pada Predator - Prey Terhadap Waktu', \text{linecolour} = [green] \} \});$
- Bidang Solusi Sistem pada Predator - Prey Terhadap Waktu*
- >  $\text{dfp} := \text{dfieldplot}([PD1, PD2], [x(t), y(t)], t = 0..100, x = 0..1, y = 0..1, \text{title} = \text{'Potret Fase Sistem pada Predator - Prey'}) ;$

- >  $fcns := \{x(t), y(t)\} :$
- >  $sistem := PD1, PD2;$
- $sistem := \frac{d}{dt} x(t) = x(t) \left( 1 - x(t) - \frac{0.4y(t)}{0.1 + x(t)} - \frac{0.1}{0.05 + x(t)} \right),$
- $\frac{d}{dt} y(t) = 0.1798611111y(t) \left( 1 - \frac{y(t)}{0.1 + x(t)} \right)$
- >  $awal1 := dsolve(\{x(0) = 0.3, y(0) = 0.4, sistem\}, fcns, type = numeric, method = classical) :$
- >  $awal2 := dsolve(\{x(0) = 0.4, y(0) = 0.2, sistem\}, fcns, type = numeric, method = classical) :$
- >  $awal3 := dsolve(\{x(0) = 0.35, y(0) = 0.2, sistem\}, fcns, type = numeric, method = classical) :$
- >  $awal4 := dsolve(\{x(0) = 0.3, y(0) = 0.4, sistem\}, fcns, type = numeric, method = classical) :$
- >  $awal5 := dsolve(\{x(0) = 0.4, y(0) = 0.5, sistem\}, fcns, type = numeric, method = classical) :$
- >  $awal6 := dsolve(\{x(0) = 0.4, y(0) = 0.3, sistem\}, fcns, type = numeric, method = classical) :$
- >  $awal7 := dsolve(\{x(0) = 0.3, y(0) = 0.4, sistem\}, fcns, type = numeric, method = classical) :$
- >  $awal8 := dsolve(\{x(0) = 0, y(0) = 0.4, sistem\}, fcns, type = numeric, method = classical) :$
- >  $awal9 := dsolve(\{x(0) = 0.4, y(0) = 0.3, sistem\}, fcns, type = numeric, method = classical) :$
- >  $awal10 := dsolve(\{x(0) = 0.15, y(0) = 0.2, sistem\}, fcns, type = numeric, method = classical) :$
- >  $gbawal1 := odeplot(awal1, [[x(t), y(t)]], -10..100, color = blue) :$
- >  $gbawal2 := odeplot(awal2, [[x(t), y(t)]], -10..70, color = blue) :$
- >  $gbawal3 := odeplot(awal3, [[x(t), y(t)]], -10..60, color = blue) :$
- >  $gbawal4 := odeplot(awal4, [[x(t), y(t)]], -10..65, color = blue) :$
- >  $gbawal5 := odeplot(awal5, [[x(t), y(t)]], -10..60, color = blue) :$
- >  $gbawal6 := odeplot(awal6, [[x(t), y(t)]], -10..60, color = blue) :$
- >  $gbawal7 := odeplot(awal7, [[x(t), y(t)]], -10..70, color = blue) :$
- >  $gbawal8 := odeplot(awal8, [[x(t), y(t)]], -10..65, color = blue) :$
- >  $gbawal9 := odeplot(awal9, [[x(t), y(t)]], -10..60, color = blue) :$
- >  $gbawal10 := odeplot(awal10, [[x(t), y(t)]], -10..70, color = blue) :$
- >  $display(gbawal1, gbawal2, gbawal3, gbawal4, gbawal5, gbawal6, gbawal7, gbawal8, gbawal9, gbawal10, dfp);$





### Titik Ekuilibrium $E_*$ Untuk Kasus $\frac{s_0}{s} < 1$

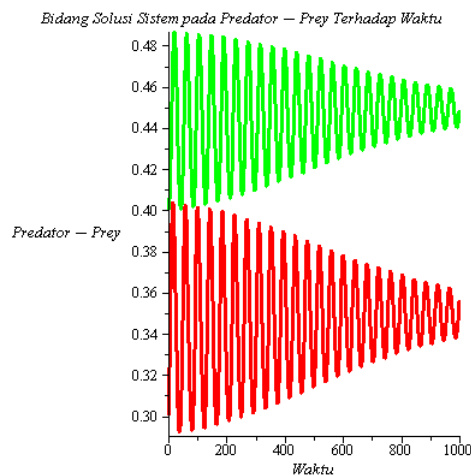
- > restart :
- > with(DEtools) :
- > with(plots) :
- >  $\alpha := 0.4; \beta := 1; c := 0.05; h := 0.1; m := 0.1;$ 
  - $\alpha := 0.4$
  - $\beta := 1$
  - $c := 0.05$
  - $h := 0.1$
  - $m := 0.1$
- >  $B := c^2 + \left(-\frac{2 \cdot \alpha}{\beta} + 2\right) \cdot c + \frac{\alpha^2}{\beta^2} - 2 \cdot \frac{\alpha}{\beta} - 4 \cdot h + 1;$ 
  - $B := 0.0225$
- >  $xbintang := \frac{1 - \frac{\alpha}{\beta} - c + \sqrt{B}}{2};$ 
  - $xbintang := 0.3500000000$
- >  $ybintang := \frac{2 \cdot m - \frac{\alpha}{\beta} - c + \sqrt{B} + 1}{2 \cdot \beta};$ 
  - $ybintang := 0.4500000000$
- >  $s_0 := c - \sqrt{B} + \frac{-\alpha^2 + \beta \cdot \alpha \cdot (-c + \sqrt{B} + 1)}{-\alpha \cdot \beta + \beta^2 \cdot (2 \cdot m - c + \sqrt{B} + 1)}$   

$$- \frac{2 \cdot h \cdot c}{\left(-\frac{\alpha}{\beta} + c + 1\right)^2 + \sqrt{B} \cdot \left(c - \frac{\alpha}{\beta} + 1\right) - 2 \cdot h};$$
  - $s_0 := 0.1798611111$
- >  $\frac{s_0}{s} = 0.9722 < 1 :$
- >  $s := \frac{s_0}{0.9722};$

```

> s := 0.185004228;
> ρ := s;
> x = 1/2 · (1 - c - α/β) + 1/2
 · √((1 - c - α/β)² - 4 · c · (α/β + h/c - 1));
> y = 1/β (m + (1/2 · (1 - c - α/β) + 1/2
 · √((1 - c - α/β)² - 4 · c · (α/β + h/c - 1)))));
> PD1 := diff(x(t), t) = x(t) · (1 - x(t) - α · y(t)/(m + x(t)) - h/(c + x(t)));
 PD1 := d/dt x(t) = x(t) (1 - x(t) - 0.4 · y(t)/(0.1 + x(t)) - 0.1/(0.05 + x(t)))
> PD2 := diff(y(t), t) = ρ · y(t) · (1 - β · y(t)/(m + x(t)));
 PD2 := d/dt y(t) = 0.185004228 · y(t) (1 - y(t)/(0.1 + x(t)))
> display({phaseportrait([PD1, PD2], [x(t), y(t)], t = 0..1000, [[x(0) = 0.3, y(0) = 0.4]],
 stepsize = 1, scene = [t, x(t)], labels = ['Waktu', 'Predator - Prey'], linecolour = [red]),
 {phaseportrait([PD1, PD2], [x(t), y(t)], t = 0..1000, [[x(0) = 0.3, y(0) = 0.4]],
 style = POINT, stepsize = 1, scene = [t, y(t)], labels = ['Waktu', 'Predator - Prey'], title =
 'Bidang Solusi Sistem pada Predator - Prey Terhadap Waktu', linecolour = [green])});

```



```

> dfp := dfieldplot([PD1, PD2], [x(t), y(t)], t = 0..100, x = -0.5..1, y =
 -0.5..1, title = 'Potret Fase Sistem pada Predator - Prey');

```

- >  $fcns := \{x(t), y(t)\} :$
- >  $sistem := PD1, PD2;$
- $sistem := \frac{d}{dt} x(t) = x(t) \left( 1 - x(t) - \frac{0.4y(t)}{0.1 + x(t)} - \frac{0.1}{0.05 + x(t)} \right),$
- $\frac{d}{dt} y(t) = 0.1850042286y(t) \left( 1 - \frac{y(t)}{0.1 + x(t)} \right)$
- >  $awal1 := dsolve(\{x(0) = 0.3, y(0) = 0.4, sistem\}, fcns, type = numeric, method = classical) :$
- >  $awal2 := dsolve(\{x(0) = 0.4, y(0) = 0.2, sistem\}, fcns, type = numeric, method = classical) :$
- >  $awal3 := dsolve(\{x(0) = 0.35, y(0) = 0.2, sistem\}, fcns, type = numeric, method = classical) :$
- >  $awal4 := dsolve(\{x(0) = 0.3, y(0) = 0.4, sistem\}, fcns, type = numeric, method = classical) :$
- >  $awal5 := dsolve(\{x(0) = 0.4, y(0) = 0.5, sistem\}, fcns, type = numeric, method = classical) :$
- >  $awal6 := dsolve(\{x(0) = 0.4, y(0) = 0.3, sistem\}, fcns, type = numeric, method = classical) :$
- >  $awal7 := dsolve(\{x(0) = 0.3, y(0) = 0.4, sistem\}, fcns, type = numeric, method = classical) :$
- >  $awal8 := dsolve(\{x(0) = 0, y(0) = 0.4, sistem\}, fcns, type = numeric, method = classical) :$
- >  $awal9 := dsolve(\{x(0) = 0.4, y(0) = 0.3, sistem\}, fcns, type = numeric, method = classical) :$
- >  $awal10 := dsolve(\{x(0) = 0.15, y(0) = 0.2, sistem\}, fcns, type = numeric, method = classical) :$
- >  $gbawal1 := odeplot(awal1, [[x(t), y(t)]], -10..100, color = blue) :$
- >  $gbawal2 := odeplot(awal2, [[x(t), y(t)]], -10..70, color = blue) :$
- >  $gbawal3 := odeplot(awal3, [[x(t), y(t)]], -10..60, color = blue) :$
- >  $gbawal4 := odeplot(awal4, [[x(t), y(t)]], -10..65, color = blue) :$
- >  $gbawal5 := odeplot(awal5, [[x(t), y(t)]], -10..60, color = blue) :$
- >  $gbawal6 := odeplot(awal6, [[x(t), y(t)]], -10..60, color = blue) :$
- >  $gbawal7 := odeplot(awal7, [[x(t), y(t)]], -10..70, color = blue) :$
- >  $gbawal8 := odeplot(awal8, [[x(t), y(t)]], -10..65, color = blue) :$
- >  $gbawal9 := odeplot(awal9, [[x(t), y(t)]], -10..60, color = blue) :$
- >  $gbawal10 := odeplot(awal10, [[x(t), y(t)]], -10..70, color = blue) :$
- >  $display(gbawal1, gbawal2, gbawal3, gbawal4, gbawal5, gbawal6, gbawal7, gbawal8, gbawal9, gbawal10, dfp);$

