# An Investigation in to Higher-order Interactions in The Lotka-Volterra Model

AIMS Mini-Project 1

Matthew Newton

Supervised by
Antonis Papachristodoulou

# 1 Abstract

Non-linear dynamical systems can describe the ways in which different species in an ecological network interact with each other. The field is well studied, however there are still large gaps in knowledge and understanding. The true nature of these interactions is very complex, however simplified models have been used to create dynamics that are easy to predict and fit real world interactions. Despite this, these models are still not accurate and it is of interest to make advancements to create better predictions as to how species behave. The Lotka-Volterra model describes the relationship between species through a quadratic relationship and is a well understood and commonly used model, as it is simple and provides accurate predictions. There are limitations to these models and it is in our interest to make them more complex to better model species interactions. In this project a higher order interaction in the form of a cubic relationship is added to the Lotka-Volterra model, to determine if it can improve the stability or even recover stability from an unstable system. The traditional Lotka-Volterra model with D-stable equilibrium admits a log-linear Lyapunov function. In this project it is proposed that a cubic Lotka-Volterra model can admit a log-linear-quadratic Lyapunov function. This project showed that provided the cubic Lotka-Volterra model meets certain conditions, it will satisfy this proposed log-linear-quadratic function. A Lyapunov function could be found for certain system that are not D-stable, the reasoning for this still remains an important research question.

# 2 Introduction

We live in a world of competition and cooperation, where multiple agents interact with one another to alter their behaviour. Whilst the laws of nature appear to fit well to the dynamical systems that we have created, they often provide cruel results, ones that cannot be solved analytically or become unstable computationally or hidden in complex dynamics. The models that are created often are good at minoring the behaviour seen in nature but can miss hidden interactions [**?**]. However, this field still continues to move forward, with new methods and simplifications that have found uses across many industries.

Whilst one can develop a mathematical model for any physical system, given sufficient information and understanding, sometimes extracting meaningful or useful results from it can be difficult. It is often useful to place models into different classes to facilitate analysis. The Lotka-Volterra model has found to fit into the category for many different physical systems [1], being most prominent in modeling population dynamics.

Higher order interactions have been proposed in many different forms to better match observations to model predictions [2]. Simple models tend not to reproduce the stability of larger systems and more natural models do not give large enough fluctuations in the states. In the field of ecology, it has been challenging to replicate results of coexistence of species in diverse environments from just purely pairwise interactions [3]. This project investigates cubic higher order interactions in the Lotka-Volterra model, determining if their stability can be altered from these interactions.

## 3 Background

### 3.1 Standard Lotka-Volterra Forms

Volterra proposed that the rate of change of a species population is of a quadratic form. The Lotka-Volterra model was first introduced in 1931 in ‘Leçons sur la Théorie Mathématique de la Lutte pour la Vie’ [4].

The general quadratic Lotka-Volterra model has well known dynamics and has been studied extensively in the literature [5]. It models the interaction of $n$ species and takes the form

$$\dot{x}_i = b_i x_i - \sum_{j=1}^{n} a_{ij} x_i x_j,\ (i = 1, \ldots, n)\,, \tag{1}$$

where $a_{ij}$ is the interaction coefficient between species $i$ and $j$ and $b_i$ is $i$'s growth rate. Denoting $X = \mathrm{diag}(x_i)$, this model can be rewritten as

$$\dot{x} = X(b - Ax). \tag{2}$$

There are three types of system of this general form that are defined by properties of $A$ [5]:

- Cooperative (resp. competitive) - if $a_{ij} \geq 0$ (resp. $\leq 0$) $\forall i \neq j$.
- Conservative - if there exists a diagonal matrix $D$ such that $D > 0$ and $AD$ is skew-symmetric.
- Dissipative - if there exists a diagonal matrix such that $D > 0$ and $AD \leq 0$. In this case, the matrix $A$ is called D-stable and the system is called ‘stably dissipative’.

It has been shown [1] that if the system is stably dissipative then there exists a global attractor and the dynamics of the attractor are Hamiltonian. It has also been shown [1] that if the interaction matrix $A$ is D-stable then there will be a Lyapunov function in the form

$$V(x) = \sum_{i=1}^{n} \alpha_i \left(x_i - q_i \log x_i\right) + K, \tag{3}$$

where $q_i$ is the equilibrium point for species $i$, $D = \mathrm{diag}(\alpha_i)$ and $K$ is a constant such that $V(q) = 0$.

Throughout this report, this Lyapunov function will be referred to as the log-linear Lyapunov function. Note that

$$\begin{aligned} \dot{V}(x) &= \sum_{i=1}^{n} \alpha_i \left(1 - \frac{q_i}{x_i}\right) \dot{x}_i, \\ &= \sum_{i=1}^{n} \alpha_i \left(x_i - q_i\right) \left(b_i - \sum_{j=1}^{n} a_{ij} x_i\right), \end{aligned}$$

which is negative definite around $x = q$ if $A^T D + DA > 0$.

### 3.2 Sum of Squares and Lyapunov Stability

Sum of squares (SOS) methods, in combination with Lyapunov theory, can be used to study stability of the equilibria of non-linear polynomial systems.

**Definition 1** *(SOS polynomial) A polynomial $p(x)$ admits a Sum of Squares (SOS) decomposition if there exists a set of polynomials $p_i(x)$, $i = 1, \ldots, M$ such that*

$$p(x) = \sum_{i=1}^{M} p_i^2(x). \tag{4}$$

It is clear that since $p(x)$ is composed of $p_i^2(x)$, then $p(x) \geq 0$.

**Proposition 1** *A polynomial $p(x)$ is SOS if and only if there exists a positive semi-definite matrix $Q$ such that*

$$p(x) = Z(x)^T Q Z(x),$$

*where $Z(x)$ is a vector containing monomials in $x$ of degree less than or equal* degree$(p(x))$.

Using this relationship, an SOS program can be converted into a semi-definite program (SDP), which can be solved with the many different SDP solvers that are freely available. The SDP solution is then converted into a solution of the SOS program.

How the method for constructing Lyapunov functions for a polynomial system works is fully covered in [6], but essentially it imposes a relaxation on the Lyapunov stability criteria by replacing polynomial non-negativity with a SOS condition. This is much more efficient than proving positivity of a polynomial, which is an NP hard problem [7]. The SOS condition implies the non-negativity condition, however it is a stricter condition.

Consider now a dynamical system of the form

$$\dot{x}(t) = f(x(t)), \tag{5}$$

where $x(t) \in \mathbb{R}^n$ and $f : \mathbb{R}^n \to \mathbb{R}^n$ is Lipschitz continuous. Assume without loss of generality that $f(0) = 0$. Definitions of stability can be found in [8]. The following result will be used throughout this report.

**Theorem 1** *Consider (5). Suppose there exists a function $V : \mathbb{R}^n \to \mathbb{R}$ such that*

- $V(0) = 0$,
- $V(x) > 0$,
- $-\frac{dV}{dt} = -\left(\frac{\partial V}{\partial x}\right) f(x) \geq 0$.

*Then $0$ is a stable equilibrium.*

**Proposition 2** *Consider (5) for which $f(x)$ is a polynomial. Suppose there exists a function $V : \mathbb{R}^n \to \mathbb{R}$ such that*

- $V(0) = 0$,

- *$V(x) - \phi(x)$ is SOS,*
- *$-\frac{\partial V}{\partial x} f(x)$ is SOS,*

*where $\phi(x) = \sum_{i=1}^{n} \sum_{j=1}^{d} \epsilon_{ij} x_i^{2j}$, such that $\sum_{j=1}^{m} \epsilon_{ij} \geq \lambda$ and $\lambda > 0$ [7]. The 0 is a stable equilibrium.*

Often in non-linear systems there are additional constraints on the states or the space that contains them. These can be defined as

- $a_i(x, u) \leq 0$ for $i = 1, \dots, N_1$,
- $b_j(x, u) = 0$ for $j = 1, \dots, N_2$,

where $x \in \mathbb{R}^n$ is the set of system states and $u \in \mathbb{R}^m$ is the set of auxiliary variables. To impose these constraints the derivative SOS constraint then becomes

$$-\frac{\partial V}{\partial x} f(x) + \sum_{i=1}^{N_1} p_i(x, u) a_i(x, u) + \sum_{j=1}^{N_2} q_j(x, u) b_j(x, u) \text{ is SOS,} \tag{6}$$

where $p_i(x, u)$ are SOS and $q_j(x, u)$ are free variables.

# 4 Simple Linear Model

Before investigating the nonlinear Lotka-Volterra model, it is worth analysing an even simpler linear system of the form

$$\dot{x} = Ax. \tag{7}$$

For this system to be stable, the real part of the eigenvalues of the matrix $A$ must be non-positive. A candidate Lyapunov function for this system takes the form $V(x) = x^T P x$ with $P$ being PSD. If the Lyapunov conditions are not satisfied then the system will be unstable and since the equilibrium is at the origin, it is not possible to stabilise the system using anything but the addition of linear terms. Hence, higher order terms will not have any impact on the stability of the zero equilibrium. It is also worth noting that the existence of a quadratic Lyapunov function for this system does not ensure that it is also D-stable, i.e. admit a diagonal Lyapunov function. The criteria for D-stability will be discussed further in a later section.

To off-set the equilibrium a constant birth rate can be added to the state space such that

$$\dot{x} = b - Ax. \tag{8}$$

This has equilibrium $q = A^{-1}b$, however shifting the state space such that the equilibrium is at the origin yields a new vector field of

$$\dot{x} = -Ax. \tag{9}$$

If this system is unstable, it is clear that a constant perturbation will not be able to stabilise the system.

## 5 General Lotka-Volterra Model

The general quadratic Lotka-Volterra model was defined in Equation (1). Consider a 2D quadratic vector field in the form

$$\begin{aligned} \dot{x} &= x(a - ax - by), \\ \dot{y} &= y(c - cy - dx). \end{aligned} \tag{10}$$

The non-zero equilibrium can be written as

$$q_x = \frac{c(a-b)}{ac-bd},\ q_y = \frac{a(c-d)}{ac-bd}. \tag{11}$$

A candidate Lyapunov function associated with the 2D quadratic vector field is

$$V = x - q_x \ln x + y - q_y \ln y + K, \tag{12}$$

where $K$ is a constant. Note that the equilibria must lie in the positive orthant, otherwise the system is not valid. It is impossible to have a non-positive population value - if the value were to be negative, the stable attractor would move the state into the positive orthant, as there can only be one stable equilibrium in this system.

Our aim here is to create a MATLAB program to automatically find the Lyapunov function given this general non-polynomial form. Effectively it will need to determine the coefficients of the Lyapunov function without the analytical information that they are equal to zero at the equilibrium. As mentioned previously, an SOS relaxation on the Lyapunov stability conditions will be used. To do this a SOS solver package called SOSTOOLS [6] is employed to deal with these constraints. For the solver to work it must be used in conjunction with an SDP solver, this program uses the SDP parser CVX [9] along with solver SeDuMi [10].

The Lyapunov function is parsed into MATLAB by declaring it in its general form, where $\beta_i$ are the coefficients that are to be determined

$$V = \beta_1 + \beta_2 x + \beta_3 y + \beta_4 \ln x + \beta_5 \ln y. \tag{13}$$

As stated in **Theorem 1**, the three Lyapunov constraints that need to be satisfied are

- $V(q) = 0$,
- $V(x)$ is PD,
- $-\dot{V}(x)$ is PSD.

As in **Proposition 2**, these constraints can be re-framed as SOS constraints

- $\beta_1 + \beta_2 q_x + \beta_3 q_y + \beta_4 \ln q_x + \beta_5 \ln q_y = 0$,
- $\beta_2, \beta_3 \geq 0.1$,
- $\nabla^2 V$ is SOS, which simplifies to $-\beta_4$ is SOS, $-\beta_5$ is SOS,
- $-\dot{V}(x) = -\frac{\partial V}{\partial x}\dot{x} - \frac{\partial V}{\partial y}\dot{y}$ is SOS.

The MATLAB program generates very accurate results, with the coefficients being equal to those found analytically to a specific accuracy. The function is plotted with $x$ and $y$ as the axes.

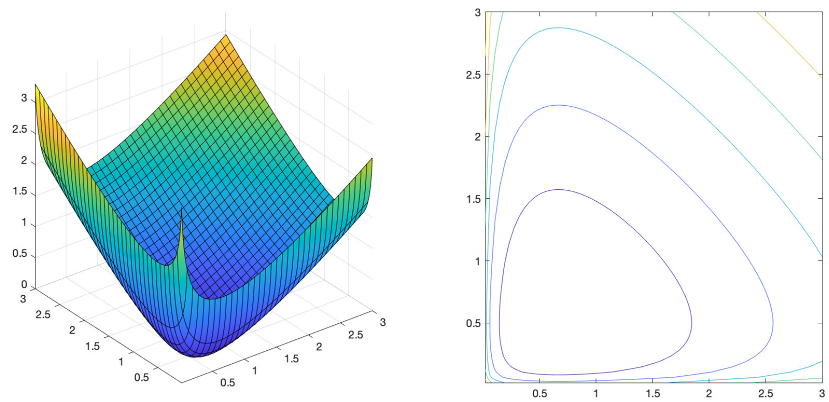

Figure 1: Surface and contour plot of Lyapunov function. $a = 3, b = 2, c = 4, d = 3$

The coefficients can be changed and as long as the equilibrium remains in the positive orthant, the program will continue to give the correct Lyapunov function. This program can be extended into higher dimensions and will continue to produce accurate results. An example 3D system is

$$A = \begin{bmatrix} 1.95 & 0.73 & -1.27 \\ 0.17 & 1.06 & -1.04 \\ 0.22 & 0.60 & 1.10 \end{bmatrix}, \; b = \begin{bmatrix} 2.00 \\ 0.44 \\ 0.80 \end{bmatrix}, \; q = \begin{bmatrix} 1.00 \\ 0.50 \\ 0.25 \end{bmatrix}. \tag{14}$$

The Lyapunov function is found in the correct form. Surface plots are created to show a slice of the function for each dimension at the equilibrium.

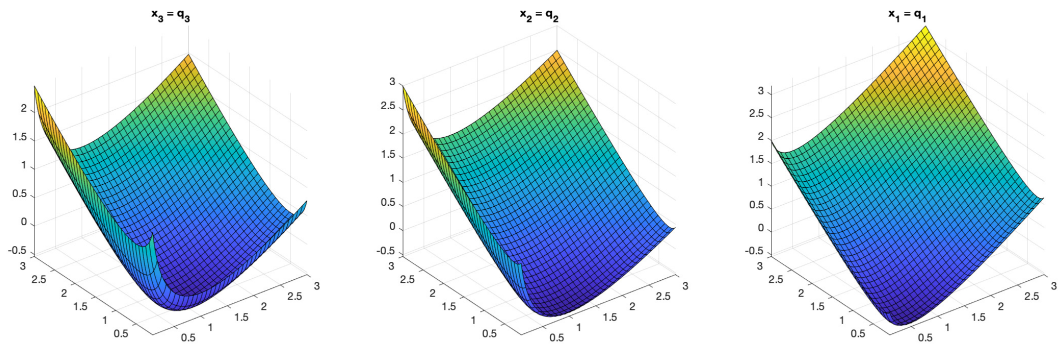


Figure 2: Surface plot of Lyapunov function for 3D system in Equation (14.)

# 6 Higher-order Lotka Volterra Model

The general Lotka-Volterra model stated before is quadratic, however higher-order terms can be added to create a cubic state space model in the form

$$\dot{x} = X\left(b - Ax - f(x)\right), \tag{15}$$

where $f(x)$ is a quadratic function.

Unlike the previous example, there is no general form for the candidate Lyapunov function. However, it is possible to propose a structure of a Lyapunov function and then check all of the stability criteria. If these are met, then the equilibrium is stable.

We are proposing a Lyapunov function in the form

$$V(x) = \sum_i \alpha_i \left(x_i - q_i \ln x_i\right) + (x-q)^T R \,(x-q) + K, \tag{16}$$

where $R$ is a matrix containing the weights for the quadratic function and $\alpha_i$ are constants. The linear and log summation will capture the linear and quadratic terms as in the previous example, and the quadratic term will 'cover' the cubic part of the state space. The vector field and Lyapunov function can be shifted such that the equilibrium is at the origin, this makes analysing the Lyapunov function conditions easier

$$\begin{aligned} \dot{x} &= \left(X + \mathrm{diag}(q)\right)\left(-Ax - f(x+q)\right), & (17)\\ V(x) &= \sum_i \alpha_i \left(x_i - q_i \ln\left(x_i + q_i\right)\right) + x^T R x + K. & (18)\end{aligned}$$

The Lyapunov stability criteria were stated in **Theorem 1**. The constant $K$ can be adjusted to ensure the Lyapunov function is zero at zero. For the positiveity condition to be true, $R$ must be PSD - we have already shown that the linear and log terms are PSD.

Let $R = 0$ for now. The time derivative condition can be evaluated by taking the first derivative w.r.t. time

$$\begin{aligned} \dot{V}(x) &= \sum_i \frac{\partial}{\partial x_i}\left(\alpha_i\left(x_i - q_i \ln\left(x_i + q_i\right)\right)\right)\dot{x}_i,\\ &= \sum_i \alpha_i \left(1 - \frac{q_i}{x_i + q_i}\right)\left(x_i + q_i\right)\left(b_i - (A(x+q))_i - f_i(x+q)\right),\\ &= \sum_i \alpha_i x_i \left(b_i - (A(x+q))_i - f_i(x+q)\right),\\ &= \sum_i \left(\alpha_i x_i \left(-(Ax)_i + f_i(q) - f_i(x+q)\right)\right).\end{aligned}$$

The quadratic term $f_i(x+q)$ can be written as a Taylor expansion, since all the third order or higher derivatives are equal to zero, then

$$f_i(x+q) = f_i(q) + \left.\frac{\partial f_i}{\partial x}\right|_{x=q} x + \frac{1}{2} x^T \nabla^2 f_i|_{x=q} x \tag{19}$$

and can be substituted into the Lyapunov equation

$$\sum_i \left(\alpha_i x_i f_i(x+q)\right) = \sum_i \alpha_i x_i \left(f_i(q) + \left.\frac{\partial f_i}{\partial x}\right|_{x=q} x + \frac{1}{2} x^T \nabla^2 f_i|_{x=q} x\right). \tag{20}$$

Since the zeroth order terms cancel, the first derivative of the Lyapunov function can then be written as

$$\dot{V}(x) = \sum_i \left( \alpha_i x_i \left( -(Ax)_i + \frac{\partial f_i}{\partial x}\Big|_{x=q} x + \frac{1}{2} x^T \nabla^2 f_i|_{x=q} x) \right) \right). \tag{21}$$

Rewritting this in index form

$$\dot{V}(x) = \sum_i \sum_j \sum_k \left( -\alpha_i x_i x_j a_{ij} + \alpha_i x_i x_j \frac{\partial f_i}{\partial x_j}\Big|_{x=q} + \frac{1}{2} \alpha_i x_j x_k \frac{\partial^2 f_i}{\partial x_j \partial x_k}\Big|_{x=q} \right). \tag{22}$$

The quadratic term can also be written in index form

$$f_i(x) = x^T Q_i x = \sum_j \sum_k x_j Q_{ijk} x_k. \tag{23}$$

The derivatives can be written in index form as

$$\frac{\partial f_i}{\partial x_j} = \sum_k Q_{ijk} x_k, \quad \frac{\partial^2 f_i}{\partial x_j \partial x_k} = Q_{ijk}. \tag{24}$$

The derivative condition then becomes

$$\dot{V}(x) = \sum_i \sum_j \sum_k (-\alpha_i x_i x_j a_{ij} + \alpha_i x_i x_j Q_{ijk} q_k + \alpha_i x_i x_j x_k Q_{ijk}) < 0. \tag{25}$$

This can be split into two conditions that guarantee that $-\dot{V}(x)$ is non-negative

$$\sum_i \sum_j \left( -\alpha_i x_i x_j a_{ij} + \alpha_i x_i x_j \sum_k Q_{ijk} q_k \right) < 0, \tag{26}$$

$$\sum_i \sum_j \sum_k (\alpha_i x_i x_j x_k Q_{ijk}) = 0. \tag{27}$$

From Equation (26) it is shown that adding the $Q_{ijk}/f_i(x)$ terms can improve the stability of the quadratic vector field. However, including these higher order interactions can shift the equilibrium position of the quadratic vector field and determining how this will alter the system behaviour is difficult to compute computationally.

Now assuming that $R$ is non-zero, the quadratic term in the Lyapunov function can be added and differentiated such that

$$\begin{aligned} \frac{\partial}{\partial t}\left(x^T R x\right) &= 2x^T R \dot{x}, \\ &= 2x^T R\left((X + \mathrm{diag}(q))(-Ax - f(x+q))\right). \end{aligned}$$

This constraint can be expanded out, however it produces complex terms which require analysis which is beyond the scope of this project. For now, the quadratic terms will be incorporated into the Lyapunov function when stated, however the mathematics will not be evaluated further.

# 7 Computing Stable Systems that are not D-Stable

The general Lotka-Volterra model is defined as stably dissipative if $A$ is D-stable (diagonally stable), which means there exists a $D$ such that $A^T D + DA < 0$. It was investigated whether it would be possible for a matrix to be linearly stable whilst not being D-stable. For the Lotka-Volterra model to be linearly stable the real parts of the eigenvalues of $\nabla \dot{x}|_q$ must be negative, which gives the condition

$$\text{real}\,(\text{eig}(\text{diag}(q)A)) < 0, \;\; q > 0, \;\; b > 0. \tag{28}$$

To find systems that satisfy these conditions, a MATLAB program was created to compare both conditions.

**Algorithm**

- Define number of states $n$.
- Repeat until counter example is found.
- Generate a random matrix $A$ of size $(n \times n)$.
- Test D-stability using CVX, $DA + A^T D < 0$ and $D > 0$.
- Define the equilibrium point as $q$.
- Calculate the eigenvalues of linearisation $\text{diag}(q)A$.
- Calculate birth rate from $A$ and $\text{diag}(q)$ through $b = -Aq$.
- If the CVX program is feasible and the real values of the eigenvalues are negative and the birth rates are all positive, then a counter example has been found.
- The matrices and vectors are stored.

To determine how common counter examples are the algorithm was repeated 10,000 times, the table below shows how many of these runs met the required conditions.

| Number of States | $n = 2$ | $n = 3$ | $n = 4$ | $n = 5$ | $n = 6$ |
|---|---|---|---|---|---|
| Number of Counterexamples | 285 | 164 | 48 | 10 | 1 |
| Probability of finding a counterexample | 0.0285 | 0.0164 | 0.0048 | 0.001 | 0.0001 |

Table 1: Table showing frequency of counterexamples found

From this table we can see that the number of counter examples decreases as the number of states increases, meaning they become increasingly rare as system dimensions gets larger. This is likely because the conditions for the parameters to be linearly stable but not D-stable becomes stricter as the size of the matrices increase. The number of parameters increase exponentially as the system states increase linearly, so it is expected that the number of counter examples will follow a similar trend.

The matrices that satisfied the counter examples are saved to be used with the SOS program in the results section.

# 8 Condition for Higher Order Terms to be D-Stable

Since the log-linear Lyapunov function does not work for systems that are not stably dissipative, the goal is to determine a Lyapunov function that will prove stability for systems of this type. In this project, the way this is attempted is to include cubic terms to the state space while still ensuring that the Lyapunov function proposed above is adequate.

Recall the condition in Equation (26)

$$\sum_i \sum_j \left( -\alpha_i x_i x_j a_{ij} + \alpha_i x_i x_j \sum_k Q_{ijk} q_k \right) < 0. \tag{29}$$

For this inequality to be true, it must satisfy the D-stability condition such that

$$\left( -A + \sum_k q_k Q_k \right)^T D + D \left( -A + \sum_k q_k Q_k \right) < 0, \tag{30}$$

where $D$ is a diagonal matrix. The terms can be expanded out

$$-AD - DA + \left( \sum_k q_k Q_k \right)^T D + D \left( \sum_k q_k Q_k \right) < 0. \tag{31}$$

The second condition

$$\sum_i \sum_j \sum_k Q_{ijk} x_i x_j x_k = 0. \tag{32}$$

Which means whatever higher order terms are added, they must ensure that the equilibrium stays in the same place. This is a very restrictive condition and likely makes the problem infeasible.

If the higher order terms are in the form

$$f_i = c_i x_j x_k,$$

then the 3rd Lyapunov condition becomes

$$\dot{V} = \sum_i \sum_j \sum_k \left( -\alpha_i x_i x_j A_{jk} - \alpha_i c_{ijk} x_i x_j x_k \right) < 0. \tag{33}$$

If $A$ is D-stable then stability can be guaranteed if

$$\sum_{ijk} \alpha_i c_{ijk} = 0, \ \forall i. \tag{34}$$

For example, if the higher order terms are added in the form

$$f_x = -c_x yz, \ f_y = -c_y xz, \ f_z = -c_z xy, \tag{35}$$

the Lyapunov condition becomes

$$\dot{V} = \sum_i \sum_j (-\alpha_i x_i x_j A_{jk}) - \alpha_x c_x xyz - \alpha_y c_y xyz - \alpha_z c_z xyz. \tag{36}$$

Therefore, if this system is to have a log-linear Lyapunov function, the following condition must hold,

$$c_x + c_y + c_z = 0. \tag{37}$$

As explained in the next section, the SOS program confirms that this is the case.

# 9 Stability Types and their Lyapunov Functions

Previously in this report the different stability types were discussed, in this section the stability types and their corresponding Lyapunov functions will be evaluated.

The Lotka-Volterra models can be summarised into the following categories:

1. Quadratic vector field
   (a) D-stable
   (b) Not D-stable but linearly stable at equilbria
   (c) Unstable
2. Cubic vector field
   (a) D-stable
   (b) Not D-stable but linearly stable at equilbria
   (c) Unstable

## 9.1 1(a): Quadratic vector field, D-stable

This has been shown to have a Lyapunov function with an analytic solution in the form $V = \sum_i \alpha_i (x_i - q_i \ln x_i)$. The SOS program should therefore always find the solution in this form. After testing this for many different systems that were D-stable, the SOS program would always find a Lyapunov function.

## 9.2 1(b): Quadratic vector field, not D-stable but linearly stable at equilbria

The log-linear Lyapunov function will not work as the system is not D-stable. There are two options for finding a Lyapunov function for this system, either add a quadratic term to the Lyapunov function or add a cubic term to the vector field such that the equilibrium of the system stays constant.

The first option involves proposing the Lyapunov function $V = \sum_i \alpha_i (x_i - q_i \ln x_i) + (x-q)^T R(x-q)$. The conditions in the SOS program are of the standard form and are used to determine if there is a feasible solution. Many different systems of this form were tested and for some of them Lyapunov functions were found. For the systems where a Lyapunov function was found, the log-linear Lyapunov function was tested as well. The SOS program was unable to find a Lyapunov function in this form as the SDP solver became infeasible. This is an exciting result as it shows that a Lyapunov function does exist for certain systems that are linearly stable but not D-stable. An example of this system is

$$A = \begin{bmatrix} -1.42 & -0.01 & 0.93 \\ -0.59 & -1.42 & -0.59 \\ 0.08 & -0.27 & -0.45 \end{bmatrix}, \; b = \begin{bmatrix} 1.19 \\ 1.45 \\ 0.17 \end{bmatrix}, \; q = \begin{bmatrix} 1.00 \\ 0.50 \\ 0.25 \end{bmatrix}. \tag{38}$$

Producing the Lyapunov function

$$\begin{aligned} V = {} & 1.0x + 1.0y + 1.0z - 1.0\log(x+1.0) - 0.5\log(y+0.5) - 0.25\log(z+0.25) + 0.3639x^2 \\ & + 0.1429xy - 0.1365xz + 0.1429xy + 0.4098y^2 + 0.2422yz + 0.2422yz - 0.1365xz + 0.9893z^2. \end{aligned} \tag{39}$$

This result was reproduce for more systems that are not D-stable but linear stable, however it would not work for all systems in this class. There are some cases where the log-linear Lyapunov function will work and some where neither the log-linear or log-linear-quadratic will work. The underlying reasons for why this is the case will not be discussed in this project, but remain an interested research question for the future.

The second option is to modify the state space to include higher order terms, initially terms that do not alter the equilibrium point of the system are added.

Currently the state space is in the form

$$\dot{x}_i = x_i \left(b_i - (Ax)_i\right). \tag{40}$$

It is shifted such that the equilibrium is at the origin

$$\dot{x}_i = (x_i + q_i)\left(-(Ax)_i\right). \tag{41}$$

Higher order terms can be added in the form $c_i x_j x_k$ so as not to move the equilibrium of interest. The state space then becomes

$$\dot{x}_i = (x_i + q_i)\left(-(Ax)_i + c_i x_j x_k\right). \tag{42}$$

The conditions for higher order terms were previously stated, it was speculated before setting up the SOS program that it would be infeasible as the constraints on what the higher order terms could be were too restrictive. The SOS program was set up for both the log-linear and log-linear-quadratic Lyapunov function. Various combinations of higher order terms were used but none of them could produce a Lyapunov function as the SDP was infeasible.

It was only possible to stabilise the system with higher order terms if it is already D-stable. This was still tested on the SOS program to compare against the conditions stated previously. All coefficients $(c_x, c_y, c_z)$ were analysed for integer values from $-5$ to 5, giving a total of 1331 possible systems. Each system was tested in the SOS program and it generated the conditions for the log-linear Lyapunov to be valid, only coefficients that satisfied $\sum_i c_i = 0$ would allow the program to be feasible.

It is also possible to add terms that alter the equilibrium, this means that the Lyapunov SOS constraints are relaxed, however the system is effectively changed from the original. For example, cubic terms that alter the equilibrium can be added such that

$$\dot{x}_i = (x_i + q_i)\left(-(Ax)_i + c_i(x_j + q_j)(x_k + q_k)\right). \tag{43}$$

Since the condition that $\sum_i c_i = 0$ no longer has to be satisfied, a larger range of higher order terms can be trialed. Both the log-linear and log-linear-quadratic candidate Lyapunov functions were trialed in the SOS program. For numerous cases the log-linear candidate Lyapunov function produced an infeasible program, whereas the log-linear-quadratic function would satisfy the constraints. An example of one of the systems is

$$A = \begin{bmatrix} -1.42 & -0.01 & 0.93 \\ -0.59 & -1.42 & -0.59 \\ 0.08 & -0.27 & -0.45 \end{bmatrix}, \ b = \begin{bmatrix} 1.19 \\ 1.45 \\ 0.17 \end{bmatrix}, \ c = \begin{bmatrix} -1.00 \\ -0.50 \\ 0.25 \end{bmatrix}, \ q_{new} = \begin{bmatrix} 1.22 \\ 0.02 \\ 0.59 \end{bmatrix}. \tag{44}$$

Producing the Lyapunov function

$$\begin{aligned} V = {} & 1.0x + 1.0y + 1.0z - 1.22log(x + 1.22) - 0.02log(y + 0.02) - 0.59log(z + 0.59) \\ & + 0.01x^2 + 0.07xy + 0.19xz + 0.07xy + 0.43y^2 + 1.15yz + 0.19xz + 1.15yz + 3.08z^2. \end{aligned} \tag{45}$$

## 9.3 1(c): Quadratic vector field, unstable

Since the system is unstable, it also will not be D-stable so the log-linear Lyapunov function will not work. Also as the system is unstable there will not be a Lyapunov function that is valid for that state space. Therefore, higher order terms must be added to the state space to alter the stability of the system. However, in this project we are only interested in terms that do not alter the equilibrium. So any higher order terms must keep the equilibrium constant. Since the system is already not linearly stable at the equilibrium, any higher order terms that are added will keep the linearisation the same and therefore will make no different to the stability. Therefore, there is no way to stability a system of this form with the methods use in this project. The SOS program was tested to see if any Lyapunov functions could be found, as expected the program could not find any valid functions. This is a positive result as it reinforces the validity of the program.

## 9.4 2(a): Cubic vector field, D-stable

This is for a system that already has the higher order cubic terms, given that the $A$ matrix is D-stable. What needs to be determined is what classes of systems fit into the log-linear and log-linear-quadratic Lyapunov functions. The conditions were suggested in the previous sections, however it still needs to be known whether the SOS program will find the corresponding Lyapunov functions.

In this project, a simplified cubic model with structure is used. As the number of variables scales exponentially with the size of $A$, we prose a 3D vector field with the structure

$$\begin{aligned} \dot{x} &= x(b - dx - a_1 y - a_2 z - cyz), \\ \dot{y} &= y(b - a_1 x - dy - a_3 z - cxz), \\ \dot{z} &= z(b - a_2 x - a_3 y - dz - cxy). \end{aligned} \tag{46}$$

The proposed Lyapunov function for this case is then

$$V = \alpha_x (x - q_x \ln x) + \alpha_y (y - q_y \ln y) + \alpha_z (z - q_z \ln z) + (x - q)^T R (x - q) + K. \tag{47}$$

However, the vector field can be shifted such that the equilibrium is at the origin. This makes analyzing the Lyapunov function easier and allows the SOS solver to be more stable.

$$\begin{aligned}
\dot{x} &= (x+q_x)\left[b - d(x+q_x) - a_1(y+q_y) - a_2(z+q_z) - c(y+q_y)(z+q_z)\right], \\
\dot{y} &= (y+q_y)\left[b - a_1(x+q_x) - d(y+q_y) - a_3(z+q_z) - c(x+q_x)(z+q_z)\right], \\
\dot{z} &= (z+q_z)\left[b - a_2(x+q_x) - a_3(y+q_y) - d(z+q_z) - c(x+q_x)(y+q_y)\right].
\end{aligned} \tag{48}$$

The Lyapunov function then becomes

$$V = \alpha_x\left(x - q_x \ln(x+q_x)\right) + \alpha_y\left(y - q_y \ln(y+q_y)\right) + \alpha_z\left(z - q_z \ln(z+q_z)\right) + x^T R x + K. \tag{49}$$

It was shown in 1(b) that certain higher order terms could produce a stable system with a log-linear Lyapunov function. For this system, the question is whether certain higher order terms can remain stable by producing a log-linear-quadratic Lyapunov function. This was initially done by parameter tuning to see if any immediate cases could be found. Setting $b = 1, d = 4, a_1 = 3, a_2 = 2, a_3 = 2, c = 1$ will produce the Lyapunov function

$$\begin{aligned}
V = {} & 0.498x + 0.498y + 0.486z - 0.0726\log(z+0.15) - 0.0489\log(x+0.098) \\
& - 0.0489\log(y+0.098) - 0.0868xy - 0.109xz - 0.109yz + 0.098x^2 + 0.098y^2 + 0.109z^2 - 0.365.
\end{aligned} \tag{50}$$

When the quadratic part of the Lyapunov function is taken away, the SOS program becomes infeasible. This means that an example of a cubic system needing a quadratic term in the Lyapunov function has been found. There are conditions that satisfy this class of three dimensional systems. However, even with a simplified state space equation as in this example, the simultaneous equations become too long and are not possible to solve analytically or computationally in a reasonable timeframe.

### 9.5 2(b): Cubic vector field, not D-stable but linearly stable at equilbria

If the cubic system is not D-stable then the log-linear Lyapunov function will not be valid. Since there are already cubic terms in the vector field and in this project we will not be looking at terms that are higher order than cubic, then adding more terms into the vector field is not an option. The log-linear-quadratic Lyapunov function can be trialed, however as this method did not work for the quadratic state space it is suggested that it will not work for this cubic state space.

### 9.6 2(c): Cubic vector field, unstable

Using the same argument as 1(c), it is not possible to stabilise a system of this form without control feedback.

## 10 Conclusion

The main efforts in this project were to determine the stability of various systems by finding a Lyapunov function. The Lyapunov stability criteria were used to analyse the constraints on a cubic Lotka-Volterra model to be stable for an equilibrium of interest. There were various cases that were analysed, depending on if the model is diagonally stable, linearly stable or cubic/quadratic, each of which has different stability criteria for the system parameters. Once the equations for the Lyapunov constraints were established, the Lyapunov stability criteria were reframed into SOS constraints. These could then be implemented into a SOS program using SOSTOOLS and an LMI solver [6], which was able to compute a Lyapunov function given a candidate structure. For a quadratic vector field, adding quadratic terms into the Lyapunov function produced a SDP that was feasible for certain system that were not D-stable. Also adding in cubic terms that altered the equilibrium would produce a log-linear-quadratic Lyapunov function. If the quadratic Lotka-Volterra model is diagonally stable then there is a log-linear Lyapunov function that can be proven analytically. If cubic terms are added then the same log-linear Lyapunov function can be used if the cubic terms satisfy certain criteria. Sometimes the log-linear Lyapunov function will not be sufficient and in which case quadratic terms must be added. The constraints on this case are strict and are stated in this report, however the classes of systems that this will be true is very restrictive and it may be hard to find such systems in the natural world.

This project has provided supplementary research into the study of Lyapunov functions related to the Lotka-Volterra model. Finding Lyapunov functions for cases where the system was not stably dissipative has provided further understanding into the non-linear behaviour of the system. Additionally, criteria for a Lyapunov function to exist for cubic Lotka-Volterra models has been established.

There is a lot of future work available in this area, the higher order interactions in this project were restricted to cubic terms, however it is possible that these terms would be quartic or even non-polynomial. Investigations into these terms could help produce more Lyapunov functions using SOS techniques.

## References


[1] Timon Idema. The behaviour and attractiveness of the lotka-volterra equations. 2005.

[2] Jacopo Grilli, György Barabás, Matthew Michalska-Smith, and Stefano Allesina. Higher-order interactions stabilize dynamics in competitive network models. *Nature*, 548, 07 2017.

[3] Jonathan Levine, Jordi Bascompte, Peter Adler, and Stefano Allesina. Beyond pairwise mechanisms of species coexistence in complex communities. *Nature*, 546, 06 2017.

[4] Vito Volterra. *Leçons sur la Théorie Mathématique de la Lutte pour la Vie.*

[5] Pedro Duarte, Rui L. Fernandes, and Waldyr M. Oliva. Dynamics of the attractor in the lotka–volterra equations. *Journal of Differential Equations - J DIFFERENTIAL EQUATIONS*, 149, 10 1998.

[6] S. Prajna, A. Papachristodoulou, and P. A. Parrilo. Introducing sostools: a general purpose sum of squares programming solver. In *Proceedings of the 41st IEEE Conference on Decision and Control, 2002.*, volume 1, pages 741–746 vol.1, Dec 2002.

[7] A. Papachristodoulou and S. Prajna. On the construction of lyapunov functions using the sum of squares decomposition. In *Proceedings of the 41st IEEE Conference on Decision and Control, 2002.*, volume 3, pages 3482–3487 vol.3, Dec 2002.

[8] Hassan K Khalil. *Nonlinear systems; 3rd ed.* Prentice-Hall, Upper Saddle River, NJ, 2002. The book can be consulted by contacting: PH-AID: Wallet, Lionel.

[9] Michael Grant and Stephen Boyd. Cvx: Matlab software for disciplined convex programming, version 2.1. http://cvxr.com/cvx, mar 2014.

[10] Jos F. Sturm. Using sedumi 1.02, a matlab toolbox for optimization over symmetric cones. *Optimization Methods and Software*, 11(1-4):625–653, 1999.