Simulation report on different predator-prey population models

Dominik Brandstetter (1605547)
Department of Physics, Faculty of Natural Sciences, Graz University
(Dated: November 17, 2020)

As part of the exercise "Methods of Modeling and Simulation" (20W PHM.204UB), I present a report on my simulation results for two different types of predator-prey population models, namely Lotka-Volterra and Strogatz types of models. Alongside the results of time-dependent simulations a fix-point and stability analysis via the Jacobians of the systems will be provided.

1. Introduction

As Alfred J. Lotka wrote in his 1920 paper, where he introduced the now as "Lotka-Volterra" known set of equations: "Periodic phenomena play an important role in nature, both organic and inorganic."[2]. Periodic behaviour between two opposite forces emerge from this astonishingly simple set of equations and thus it provides a starting point for simulations of all sorts of real-world phenomena. This model will be contrasted by the "Strogatz" model, an alternative, yet as we will find out equivalent, approach for simulating a non-linear dynamical system now depending on capacities.
Although those models are very generic and can be applied to many types of systems, we will look at them through the particular lense of populations of hare (prey) and lynx (predator) on an otherwise uninhabited island.

In section 2 we will briefly discuss both model types and introduce notation and variables. Additionally, we will look at the fix-point and stability analysis performed. In section 3 the methods of simulation and calculation will be outlined and the results of said simulation and analysis will be provided. Last but not least the results will be discussed in section 4.

FIG. 1: Picture of a lynx[3] (left; predator) and a hare[1] (right; prey; might be rabbit though). Despite not being in the same biotope in the images, we assume them to be the only species present on a deserted island. Lynxes eat hares but not the other way around.

2. Theory

2.1. Lotka-Volterra model

In the Lotka-Volterra model we define a source q_i and a sink s_i for the population x_i of the species i. Thus the equation of state for the species i can be simply stated as the difference between q_i and s_i

$$\frac{d}{dt}x_i = q_i - s_i \tag{1}$$

For the hare population x_h it is reasonable to expect a positive, exponential growth if no predator is present, thus the source q_h can be stated as

$$q_h = \alpha * x_h$$

with α being the growth rate and takes in the difference between natural deaths and births. The sink s_h for the hare population depends on the hare population x_h and the lynx population x_l correlated by impact factor β. β takes into account the number of deadly encounters

$$s_h = \beta * x_h * x_l$$

For the lynx population x_l we revert those terms, meaning we expect them to die out if no prey is present. The (conventionally positive) growth rate δ will be part of the sink s_l

$$s_l = \delta * x_l$$

Last, as the hare population decreases, we assume the lynx population to increase with each deadly encounter

$$q_l = \gamma * x_l * x_h$$

Here γ is, of course, related to β as it also takes the number of deadly encounters into consideration. Often $\beta = \gamma$ is assumed, however, they do not need to be equal, since we don't necessarily expect as many lynx to be born as hares were killed.

We can therefore state the equations of state for our Lotka-Volterra system as

$$\frac{d}{dt}x_h = \alpha * x_h - \beta * x_h * x_l$$
$$\frac{d}{dt}x_l = \gamma * x_l * x_h - \delta * x_l \quad (2)$$

Unfortunately, this is a system of non-linear differential equations and as such particularly hard to solve analytically. We will therefore restrain from doing so and perform a numerical simulation instead (see section 3).

2.2. Strogatz model

The Strogatz model is in fact just a Lotka-Volterra model in disguise. We start from the equations of state from the Lotka-Volterra system (2) and pull out the population x_h / x_l and growth rate α / δ

$$\frac{d}{dt}x_h = \alpha * \left(1 - \frac{\beta}{\alpha} * x_l\right) * x_h$$
$$\frac{d}{dt}x_l = -\delta \left(1 - \frac{\gamma}{\delta} * x_h\right) * x_l \quad (3)$$

Now we define

$$c_l = \frac{\alpha}{\beta}$$
$$c_h = \frac{\delta}{\gamma} \quad (4)$$

and thus the equations of state for the Strogatz model are

$$\frac{d}{dt}x_h = \alpha * \left(1 - \frac{x_l}{c_l}\right) * x_h$$
$$\frac{d}{dt}x_l = -\delta \left(1 - \frac{x_h}{c_h}\right) * x_l \quad (5)$$

Now we can actually see what those "new" parameters represent: they are the capacity of the other population. If the population of hare reaches its capacity c_h the growth rate for the lynx will vanish and vice versa.

2.3. Fix-Point analysis

A fix-point is characterised by a stable population of hare and lynx. This means we need to set the equations of state 0 and solve for the population.

$$\frac{d}{dt}x_h = \frac{d}{dt}x_l = 0 \rightarrow \{x_{h;fix}, x_{l;fix}\} \quad (6)$$

2.3.1. Lotka-Volterra model

$$\frac{d}{dt}x_h = 0 \Rightarrow \alpha * x_h = \beta * x_h * x_l$$
$$\frac{d}{dt}x_l = 0 \Rightarrow \gamma * x_l * x_h = \delta * x_l$$

The trivial solution is of course $\{0, 0\}$. However assuming a non-zero population of hare and lynx we get $\left\{\frac{\delta}{\gamma}, \frac{\alpha}{\beta}\right\}$.

2.3.2. Strogatz model

$$\frac{d}{dt}x_h = 0 \Rightarrow x_l = \frac{x_l}{c_l} * x_h$$
$$\frac{d}{dt}x_l = 0 \Rightarrow x_l = \frac{x_h}{c_h} * x_l$$

Again we have the trivial solution $\{0, 0\}$ but assuming a non-zero population also a second solution $\{c_h, c_l\}$, which is unsurprisingly the capacity limit for both populations.

2.4. Stability analysis

The stability of our systems is characterised by the eigenvalues of the Jacobian matrix J in the fix-points[4]

$$J = \left|\begin{pmatrix} \partial/\partial x_h * dx_h/dt & \partial/\partial x_l * dx_h/dt \\ \partial/\partial x_h * dx_l/dt & \partial/\partial x_l * dx_l/dt \end{pmatrix}\right|_{x_{h;fix}, x_{l;fix}} \quad (7)$$

The eigenvalues of any 2x2 matrix M are, as is easily computed,

$$\lambda_{\pm} = \frac{1}{2}\left(tr\{M\} \pm \sqrt{tr\{M\}^2 - 4 * det\{M\}}\right) \quad (8)$$

2.4.1. Lotka-Volterra model

The partial derivates are

$$\frac{\partial}{\partial x_h}\frac{dx_h}{dt} = \alpha - \beta * x_{l;fix}$$
$$\frac{\partial}{\partial x_l}\frac{dx_h}{dt} = -\beta * x_{h;fix}$$
$$\frac{\partial}{\partial x_h}\frac{dx_l}{dt} = \gamma * x_{l;fix}$$
$$\frac{\partial}{\partial x_l}\frac{dx_l}{dt} = \gamma * x_{h;fix} - \delta \quad (9)$$

Calculating the trace and determinate of our Jacobian for the non trivial fix-point, however, yields

$$\begin{aligned} tr\{J\} &= \alpha - \beta * x_{l;fix} + \gamma * x_{h;fix} - \delta \\ &= \alpha - \beta * \frac{\alpha}{\beta} + \gamma * \frac{\delta}{\gamma} - \delta \\ &= 0 \end{aligned} \tag{10}$$

$$\begin{aligned} det\{J\} &= (\alpha - \beta * x_{l;fix}) * (\gamma * x_{h;fix} - \delta) \\ &\quad + (\beta * x_{h;fix}) * (\gamma * x_{l;fix}) \\ &= \alpha\delta \end{aligned} \tag{11}$$

which means that the eigenvalues

$$\begin{aligned} \lambda_{\pm} &= \pm\frac{1}{2}\left(\sqrt{-4\alpha\delta}\right) \\ &= \pm i\sqrt{\alpha\delta} \end{aligned} \tag{12}$$

of a Lotka-Volterra model will be purely imaginary and thus the system will fall into an undamped oscillation.[4] For the trivial fix-point $\{0, 0\}$ the trace

$$\begin{aligned} det\{J\} &= \alpha - \beta * x_{l;fix} + \gamma * x_{h;fix} - \delta \\ &= \alpha - \delta \end{aligned} \tag{13}$$

is non-zero. Calculating the determinate

$$\begin{aligned} det\{J\} &= (\alpha - \beta * x_{l;fix}) * (\gamma * x_{h;fix} - \delta) \\ &\quad + (\beta * x_{h;fix}) * (\gamma * x_{l;fix}) \\ &= -\alpha\delta \end{aligned} \tag{14}$$

shows that the eigenvalues

$$\begin{aligned} \lambda_{\pm} &= \frac{1}{2}\left(\alpha - \delta \pm \sqrt{\alpha^2 - 2\alpha\delta + \delta^2 + 4\alpha\delta}\right) \\ &= \frac{1}{2}\left(\alpha - \delta \pm (\alpha + \delta)\right) \\ &= (\alpha, -\delta) \end{aligned} \tag{15}$$

are both real and will have a different sign. This means that this fix-point is a saddle point and as such not a stable convergence point.[4]

2.4.2. Strogatz model

As we showed above, this model is mathematically equivalent to the Lotka-Volterra model. This means the results for the stability analysis will also apply for this model.

3. Simulation

As discussed above the equations of state are non-linear differential equations and this work will make no attempt on solving them analytically. The integration will be done numerically and to avoid a large numerical error, a variable time step Runge-Kutta 4 solver will be employed.
All code has been written in Python 3.8.5, the RK4 algorithm has been provided by the SciPy package 1.5.2 (scipy.integrate.solve_ivp) and the plotting has been done with MatPlotLib 3.3.1. Additionally, the NumPy package 1.19.1 has been used for calculating the determinate and the trace and the cmath built-in module handled the negative square roots. The code, excluding code for plotting, can be found in the appendix B.
The integrator used is not symplectic, thus, the population is strictly speaking not periodic and the population of hare and lynx will differ from cycle to cycle. However due to the high-order Runge-Kutta with non-fixed time steps ($dt <= 0.1\ a.u.$) this error is negligible and not visible in any of the data.

For this work, four systems were tested. The first three are classic Lotka-Volterra models and the fourth one is a Strogatz one. System 2 and 3 are the same system with different initial populations. All plots can be found in appendix A.

3.1. System 1

System 1 is classic Lotka-Volterra model with a growth rate $\alpha = 0.5$, $\beta = \gamma = 0.008$ and $\delta = 0.8$. The initial population is $\{50, 25\}$ and it was integrated in the interval $t = [0, 50]\ a.u.$.
The non trivial fix-point is $\{100, 62.5\}$ and the eigenvalues of the Jacobian at this fix-point are $\lambda_{\pm} = \pm 0.6324$.

3.2. System 2 & System 3

System 2 and 3 are also classic Lotka-Volterra models with a growth rate $\alpha = 0.3$, $\beta = 0.025$, $\gamma = 0.0015$ and $\delta = 0.2$. The initial population is $\{150, 12\}$ for system 2 and $\{40, 10\}$ for system 3. It was integrated in the interval $t = [0, 100]\ a.u.$.
The non trivial fix-point is $\left\{133.\dot{3}, 12\right\}$ and the eigenvalues of the Jacobian at this fix-point are $\lambda_{\pm} = \pm 0.2449$.

3.3. System 3

System 3 is a Strogatz model with a growth rate $\alpha = \beta = 0.04$ and capacities $c_h = 50$ and $c_l = 10$. The initial population is $\{10, 2\}$ and it was integrated in the interval $t = [0, 500]\ a.u.$.

The non trivial fix-point is $\{50, 10\}$ and the eigenvalues of the Jacobian at this fix-point are $\lambda_{\pm} = \pm 0.03\dot{9}$.

4. Discussion

As discussed above, all Lotka-Volterra-like models will experience an undamped oscillation around the non-trivial fix-point (if the system doesn't start in either fix-point). Thus, as was to be expected, all tested systems have purely imaginary eigenvalues which differ only in sign.
Interesting to note for all systems is, that contrary to what one might think, the hare population continues to increase after the point of lowest lynx population. The populations are not completely asynchronous by half a period, but less.

4.1. System 1

In terms of frequency, this model is by far the quickest. It rapidly goes through cycles of high population and low population.

4.2. System 2 & 3

Those systems show how the ratio of hare to lynx at the beginning can influence the system for all time. Even though system 2 starts with more hare, system 3 will reach much higher hare populations. On the contrary, it also reaches a much lower population, whereas it is much more stable in system 2.

4.3. System 4

In terms of frequency, this model is by far the slowest. This system experiences, compared to the other ones, much longer periods of low hare population. This is because once the hare population reaches its capacity the growth rate for lynx will fall to 0. This will cause the lynx population to fall rapidly.
Since the capacity for one population controls the growth of the other, the name capacity can be questioned. However, looking at the fix-point for the system, the capacities represent the maximum stable populations.

---

[1] ErikaWittlieb. 2017 (accessed October 17, 2020).

[2] A. J. Lotka. Analytical note on certain rhythmic relations in organic systems. *Proceedings of the National Academy of Sciences*, 6(7):410–415, 1920.

[3] skeeze. 2015 (accessed October 17, 2020).

[4] A. K. Steiner. *Methods of Modeling and Simulation*, 2020.

Appendix A

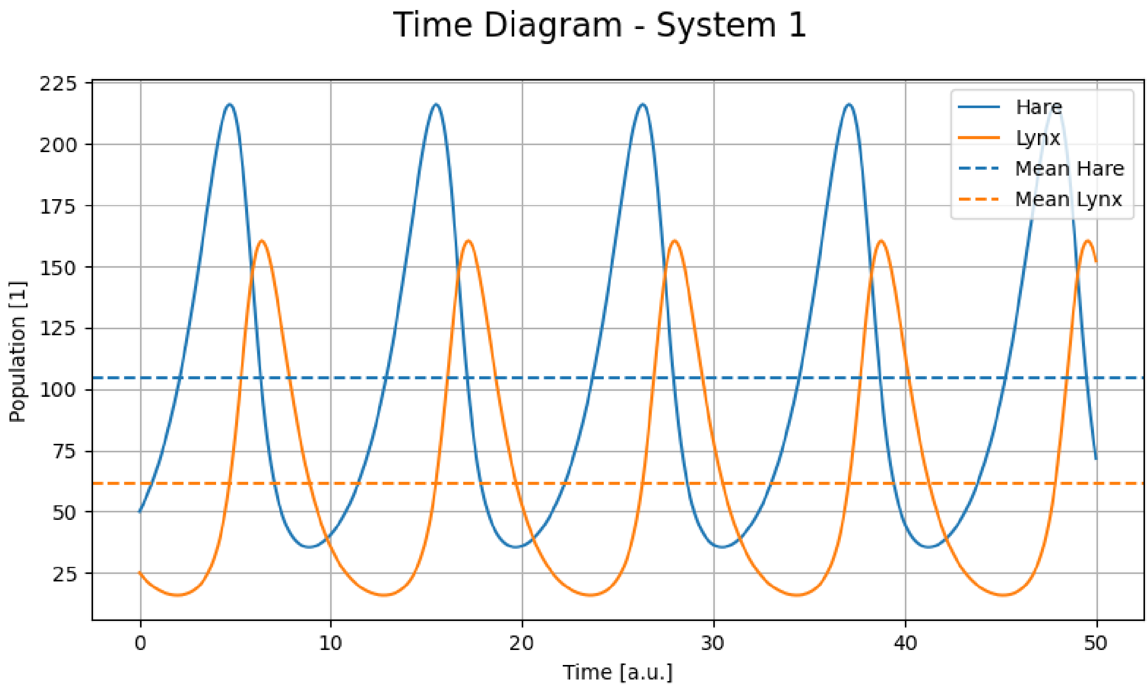

FIG. 2: Time diagram for the system 1 with the following parameters: $\alpha = 0.5$; $\beta = \gamma = 0.008$; $\delta = 0.8$ and the following initial population: $\{50, 25\}$. The hare population over time in blue; its mean value in dashed blue. The lynx population in orange; its mean value in dashed orange.

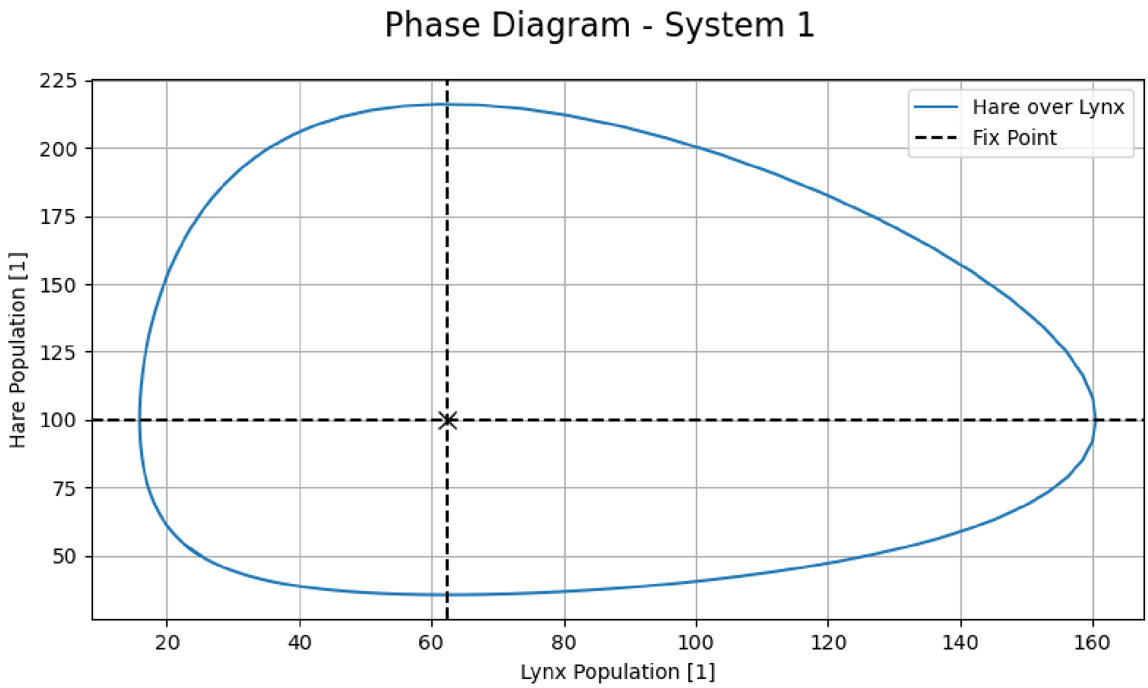

FIG. 3: Phase diagram for the system 1. The hare over lynx ratio in blue and the non trivial fix-point $\{100, 62.5\}$ in black.

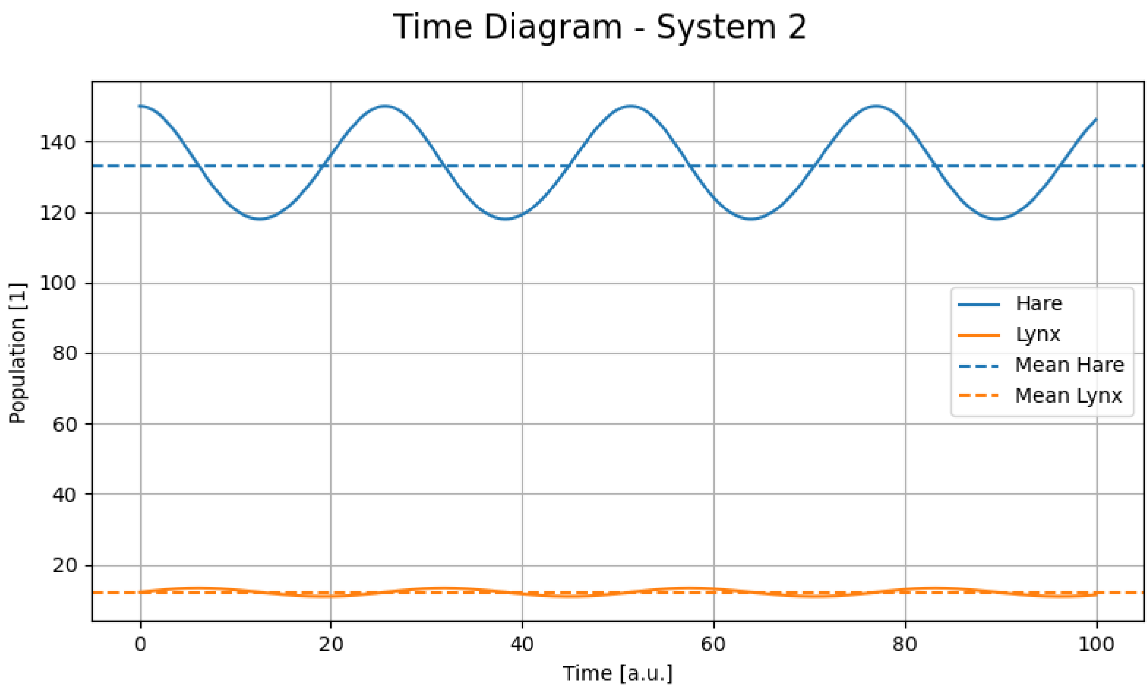

FIG. 4: Time diagram for the system 2 with the following parameters: $\alpha = 0.3$; $\beta = 0.025$; $\gamma = 0.0015$; $\delta = 0.2$ and the following initial population: $\{150, 12\}$. The hare population over time in blue; its mean value in dashed blue. The lynx population in orange; its mean value in dashed orange.

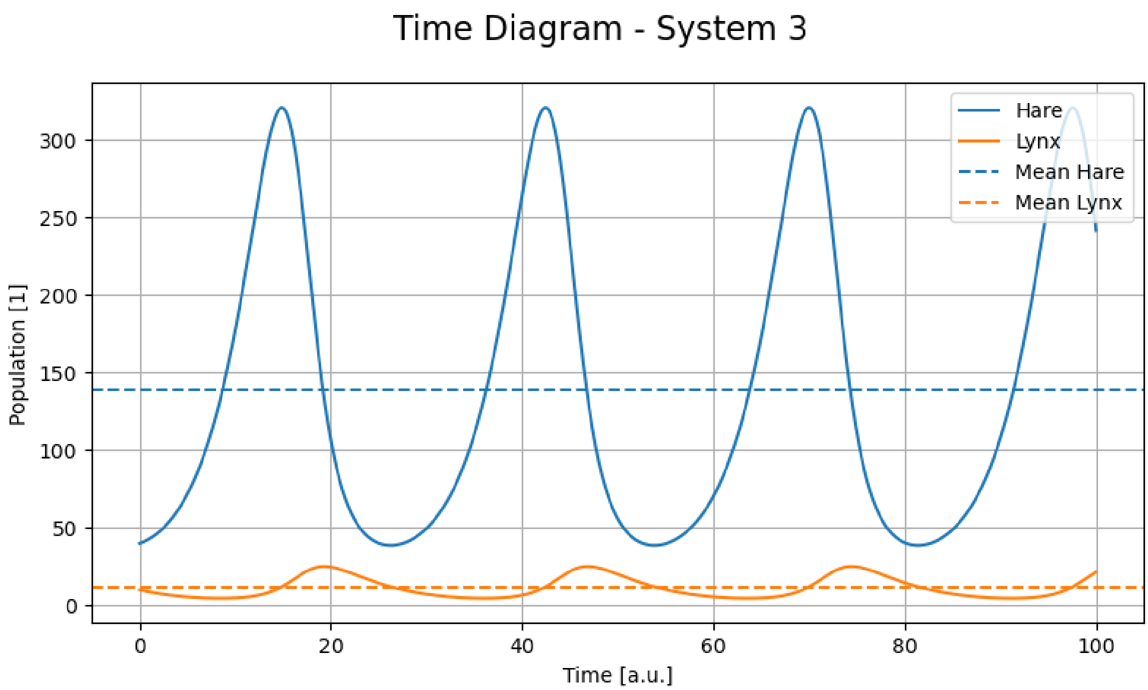

FIG. 5: Time diagram for the system 3 with the following parameters: $\alpha = 0.3$; $\beta = 0.025$; $\gamma = 0.0015$; $\delta = 0.2$ and the following initial population: $\{40, 10\}$. The hare population over time in blue; its mean value in dashed blue. The lynx population in orange; its mean value in dashed orange.

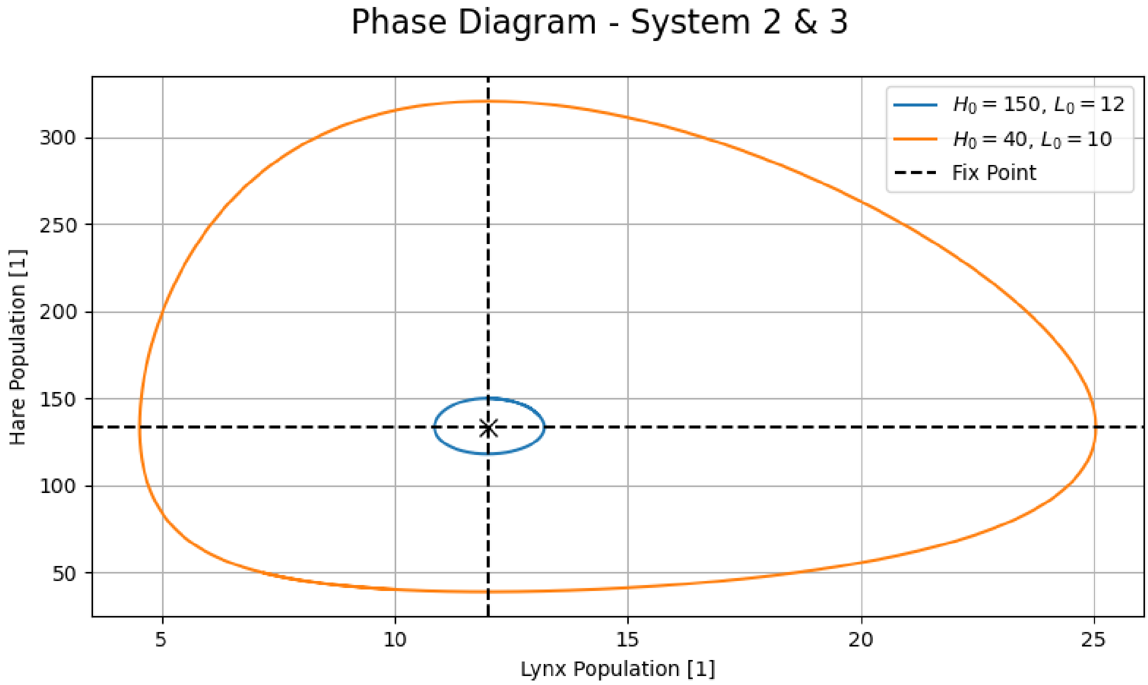

FIG. 6: Phase diagram for the system 1. The hare over lynx ratio in blue and the non trivial fix-point $\{133.\dot{3}, 12\}$ in black.

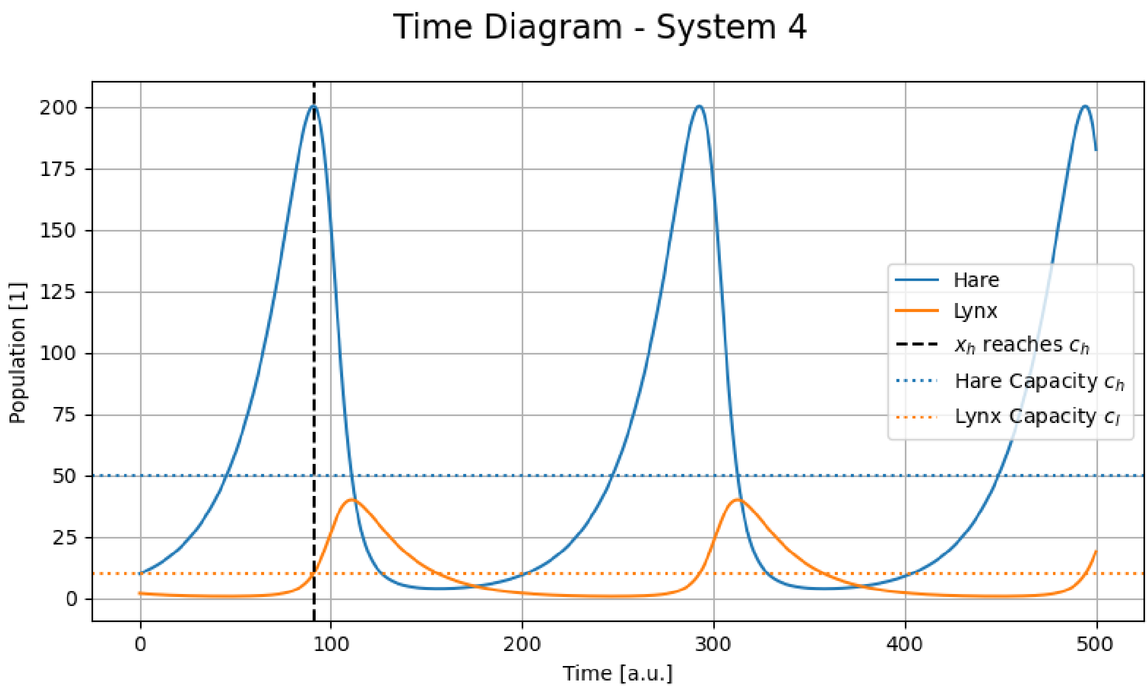

FIG. 7: Time diagram for the system 4 with the following parameters: $\alpha = \beta = 0.04$; $c_h = 50$; $c_l = 10$ and the following initial population: $\{10, 2\}$. The hare population over time in blue; its mean value in dashed blue. The lynx population in orange; its mean value in dashed orange. The first time the hare population x_h reaches its capacity c_h is marked in dashed black.

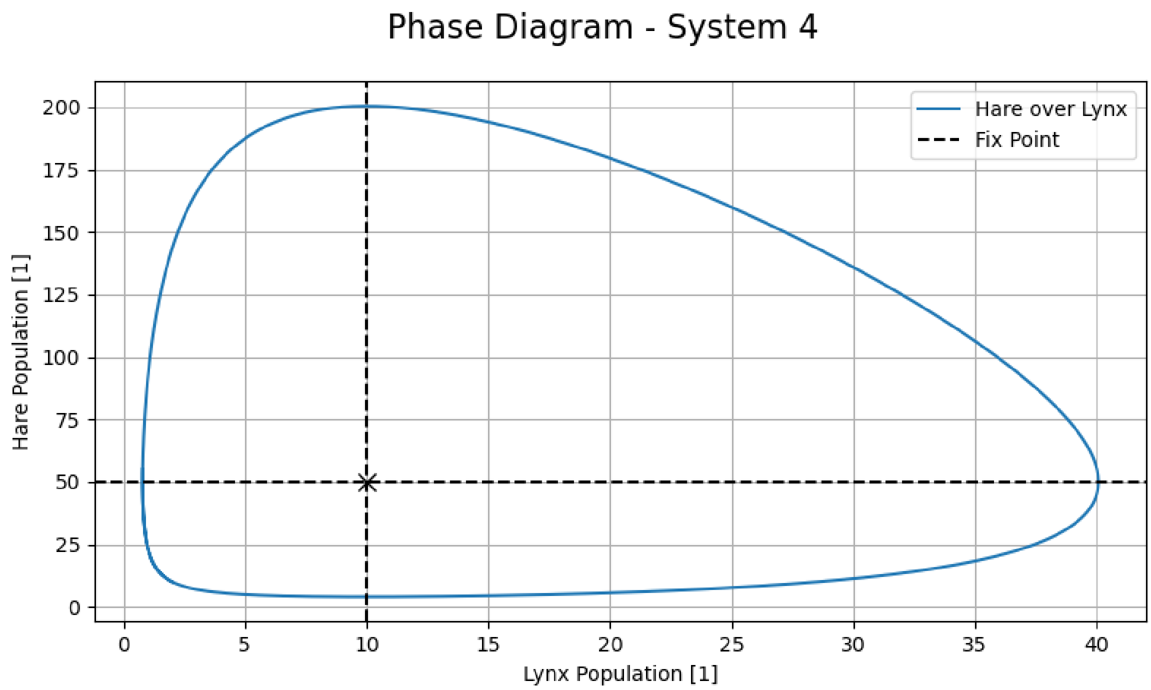

FIG. 8: Phase diagram for the system 1. The hare over lynx ratio in blue and the non trivial fix-point $\{50, 10\}$ in black.

Appendix B

```
# Python Import
import cmath
from abc import abstractmethod
from dataclasses import dataclass

# Third Party Import
import numpy as np
from numpy import linalg
from scipy.integrate import solve_ivp


class PopulationModel():

    def integrate(self, t_0, t_end, max_dt=0.1):

        solution = solve_ivp(self._integration_func,
                             [t_0, t_end],
                             [self.prey_0, self.predator_0],
                             max_step=max_dt)

        return solution.t, solution.y

    def is_stable(self, jacobian):

        trace = np.trace(jacobian)
        det = linalg.det(jacobian)
        root = cmath.sqrt(trace**2 - 4 * det)

        eigenvalue_1 = (trace + root) / 2
        eigenvalue_2 = (trace - root) / 2

        return eigenvalue_1, eigenvalue_2

    @abstractmethod
    def get_fix_points(self):
        pass

    @abstractmethod
    def _integration_func(self, t, y):
        pass


@dataclass
class LotkaVolterraModel(PopulationModel):
    '''
    d/dt prey = alpha*prey - beta*prey*predator
    d/dt predator = gamma*prey*predator - delta*predator
    '''

    alpha: float
    beta: float
    gamma: float
    delta: float
    prey_0: float = 0
    predator_0: float = 0
```

```
    def get_fix_points(self):

        predator_fix_point = self.alpha / self.beta
        prey_fix_point = self.delta / self.gamma

        return [[0, 0], [prey_fix_point, predator_fix_point]]

    def is_stable(self):

        prey, predator = self.get_fix_points()[1]

        a = self.alpha - self.beta * predator
        b = -self.beta * prey
        c = self.gamma * predator
        d = self.gamma * prey - self.delta

        return super().is_stable([[a, b], [c, d]])

    def _integration_func(self, t, y):

        y_new = np.zeros(2)

        y_new[0] = self.alpha * y[0] - self.beta * y[0] * y[1]
        y_new[1] = self.gamma * y[0] * y[1] - self.delta * y[1]

        return y_new


@dataclass
class StrogatzModel(PopulationModel):
    '''
    d/dt prey = alpha*(1 - predator/predator_capacity)*prey
    d/dt predator = -beta*(1 - prey/prey_capacity)*predator
    '''

    alpha: float
    beta: float
    prey_capacity: float
    predator_capacity: float
    prey_0: float = 0
    predator_0: float = 0

    def get_fix_points(self):

        return [[0, 0], [self.prey_capacity, self.predator_capacity]]

    def is_stable(self):

        prey, predator = self.get_fix_points()[1]

        a = self.alpha * (1 - predator / self.predator_capacity)
        b = -self.alpha * prey / self.predator_capacity
        c = self.beta * predator / prey
        d = -self.beta * (1 - prey / self.prey_capacity)

        return super().is_stable([[a, b], [c, d]])
```

```
def _integration_func(self, t, y):

    y_new = np.zeros(2)

    y_new[0] = self.alpha * (1 - y[1] / self.predator_capacity) * y[0]
    y_new[1] = -self.beta * (1 - y[0] / self.prey_capacity) * y[1]

    return y_new
```