



Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich



Master Thesis
Master of Advanced Studies in Finance at the
Chair of Entrepreneurial Risks

Semi-analytical solution of a Generalized Delay Logistic Equation

Antoine Beuchat

Professor: Prof. Dr. Didier Sornette

March 7, 2010

Abstract

In this master we show how to obtain a semi-analytical solution to the generalized delay logistic equation $dx/dt = x(1 - x/(a + bx(t - \tau)))$ via a method of steps. The results are supported by numerical computations. In addition, the semi-analytical solution is used as a basis to a new regime classification for the case $a = 0$. We complement the set of regimes described in [9] with diagrams using the generalized carrying capacity as the resource limit. Since we develop new approaches to the study of the generalized logistic equation, we hope that they might be a source of inspiration for economical, population categorization or fitting problems.

Contents

Abstract	i
1. Introduction	1
2. Logistic Equations	3
2.1. Standard logistic equation	3
2.2. Generalized delay logistic equation (GDLE)	7
3. Step-wise construction	9
4. Regime classification	15
5. Discussion	27
A. Numerical sim. with Mathematica	31
B. Numerical simulations with R	33
List of Figures	45
Bibliography	47

1

Introduction

The delay equation studied in this master was firstly introduced in [9]. The solution of the delay equation allows to describe punctuated evolution of complex systems in a variety of applications among which the variation of demand and supply in the economy [2], development of human societies and the growth of firms. These systems are characterized by periods of stagnation followed by period of radical changes [1, 5, 4].

The emphasis of this work was originally to obtain an example that could support our new model, however in the course of the master a new graphical representation as well as a new analytical solution were found shifting our original intention.

The master is organized as follows. Chapter 2 introduces the standard logistic equation as well as the generalized delay logistic equation (GDLE). The introduction consists in the definition of the logistic equations and a geometrical guide to construct the standard logistic equation sharpening our intuition regarding our model. Since the non-linear delay equations are complex in their behaviour, Chapter 3 develops a graphical construction of the solution of the GDLE. The construction consist in a multi-step approach suggested by the nature of our model. The multi-step approach leads to the novel semi-analytical solution supporting the stability analysis advanced by Yukalov, Yukalova and Sornette [9]. Finally Chapter 4 reviews the scheme of stability analysis and complements it by a topological view.

The results in this study are supported by numerical integration of the

GDLE. The integration of the GDLE was done via an iterative procedure ensuring that the successive approximations of the GDLE solution yielded to a satisfactory local error estimate, see Appendix B. We implemented the solver based on the software R [7]. For future work on fitting problems of the GDLE an algorithm was also written based on the Levenberg-Marquardt method. The method finds candidate solutions to a nonlinear least squares algorithm. The aim is to find an optimal (ideally the optimal) solution from on a multi-dimensional parameter space. Fitting results on synthetic data are found in Appendix B. In addition to our solvers, the semi-analytical solution allowed us to do numerical experiments changing the set of parameters of our model dynamically. The dynamic experiments were performed with the software Mathematica [8], see Appendix A.

2

Logistic Equations

In order to have a flavor of the behavior of the generalized delay logistic equation (GDLE) the well known logistic equation will be introduced. Then, a generalized form of the logistic equation with a time-delay in the carrying capacity is presented based on the general model advanced by Yukalov, Yukalova and Sornette [9]. Since the non-linear delayed equations are difficult to study, we develop an intuitive graphical construction of the solution of the GDLE. The construction with a step approach will be presented in the next chapter.

2.1. Standard logistic equation

The logistic function also frequently called *S*-shape is the solution of the following autonomous first order non-linear differential equation.

$$\frac{dN(t)}{dt} = rN(t) \left(1 - \frac{N(t)}{K} \right), \quad t \geq 0 \quad (2.1)$$

where $N(t)$ is characterizing the physical system while the coefficient r and K could be interpreted as the parameters controlling the growth rate and the carrying capacity of the environment, respectively. Throughout our study t is the time, and therefore it is always positive. Setting the initial condition $N(0) := N_0 \geq 0$ the solution of the system is

$$N(t) = \frac{KN_0 e^{rt}}{K + N_0(e^{rt} - 1)} \quad (2.2)$$

To develop some intuition of the dynamics of Eq. (2.1) a direction field is presented in Fig. 1 for a particular parameterization.

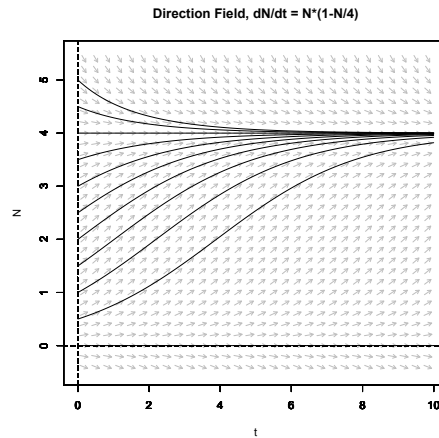


Fig. 1.: Direction field of the Verhulst equation (2.1) and integral curves for different initial conditions. In this figure the carrying capacity is $K = 4$ and the initial conditions are from the bottom to the top $N_0 = \{0, 0.5, 1, \dots, 5\}$.

The rate of change is constituted by the sum of a linear and a quadratic term, the linear term is greater than the quadratic term when $N(t) < K$, the opposite holds true.

$$rN(t) - \frac{rN^2(t)}{K} > 0 \quad \Rightarrow \quad N(t) < K \quad (2.3)$$

For $N(t) \ll K$, the logistic curve grows exponentially, since the linear term in 2.2 dominates. The limiting case is the simplest population growth model described by the equation

$$\frac{dN}{dt} = rN(t) \quad (2.4)$$

For $N(t)$ close to K the term $(1 - N(t)/K) \rightarrow 0$ and the rate of change tends to zero. This situation is visualized by the plateau approaching K in Fig 1. Finally for $N(t) \gg K$ the quadratic term dominates and the limiting case is

$$\frac{dN}{dt} = -\frac{rN^2(t)}{K} \quad (2.5)$$

The logistic curve behaves as K/rt for t close to zero.

Formally, the separation (if any) between regimes of growth and the decay or equivalently convexity and concavity is found via the inflection points of Equation 2.1. Let t^* be the time at which the logistic equation

has an inflection point. The time t^* satisfies $N''(t^*) = 0$ and in the epsilon neighborhood of t^* , $N''(t^* + \epsilon)$ and $N''(t^* - \epsilon)$ has opposite signs. $N''(t)$ is given by

$$N''(t) := \frac{d^2N(t)}{dt^2} = r^2N(t) \left(1 - \frac{N(t)}{K}\right) \left(1 - \frac{2N(t)}{K}\right), \quad t \geq 0 \quad (2.6)$$

A study of signs reveals that the point $N(t) = K/2$ is the inflection point, therefore t^* satisfies $N(t^*) = K/2$, leading to

$$t^* = -\frac{1}{r} \ln \left(\frac{N_0}{K - N_0} \right), \quad t^* \geq 0 \quad (2.7)$$

To satisfy $t^* \geq 0$ the logarithm term should satisfy $0 \leq \left(\frac{N_0}{K - N_0}\right) \leq 1$ leading to a new condition $N_0 < K/2$. In summary, the separation of regimes in (2.1) exists only for the initial conditions $N_0 < K/2$. Note that in Fig. 1 for $N_0 < K/2 = 2$ the inflection point separating two regimes can be visualized. Furthermore, the time t^* for $N_0 = 2$, $r = 1$, $K = 4$ is $t^* \simeq 1.1$.

2.1.1. Geometrical construction

Before introducing the generalized delay logistic equation and its construction steps, the construction of the logistic equation advanced by Verhulst (2.1) is presented. Recall that the evolution of the logistic equation (2.1) is fully characterized by the initial condition N_0 and its two parameters r and K . The solution for the physical system for $N_0 < K/2$ can be constructed as follows (see Fig. 2)

1. Draw the constant carrying capacity K
2. Draw the limiting case $N(t) = N_0 e^{rt}$ obtained from (2.4)
3. Draw the limiting case $N(t) = K - e^{-rt}(K/N_0 - 1)$. The case is obtained from (2.4) and noting that the logistic equation is symmetric under rotation around the inflection point. More precisely, applying to (2.4) the following transformation $t \rightarrow 2t^* - t$ and $N \rightarrow 2N(t^*) - t$, where t^* is the inflection point.

For small t start at N_0 and follow the first limiting case corresponding to the exponential growth up to the inflection point. At the inflection point and for higher t follow the second limiting case corresponding to the negative exponential decay. In the case of $K/2 < N_0 < K$ follow the negative exponential decay (see Fig. 3).

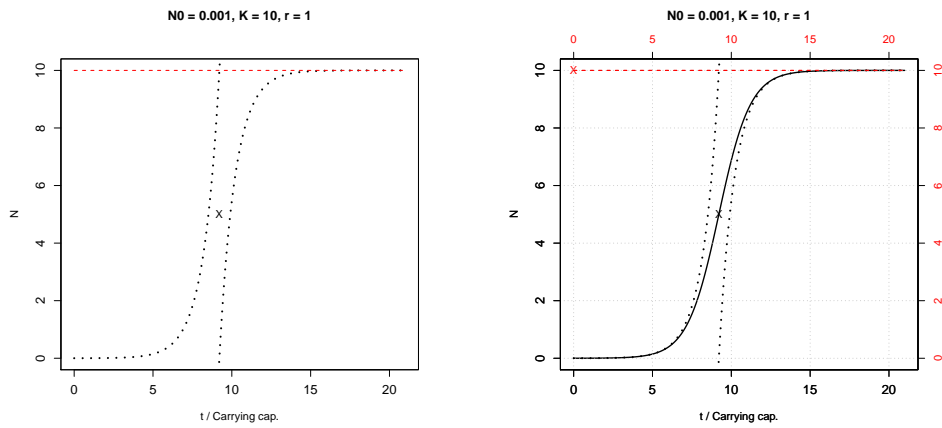


Fig. 2.: Construction of the logistic equation, case $N_0 < K/2$, in dotted line the carrying capacity. The first graph shows the limiting cases and the inflection point. The parameters of (2.1) are $K = 10, r = 1$ and $N_0 = 0.001$.

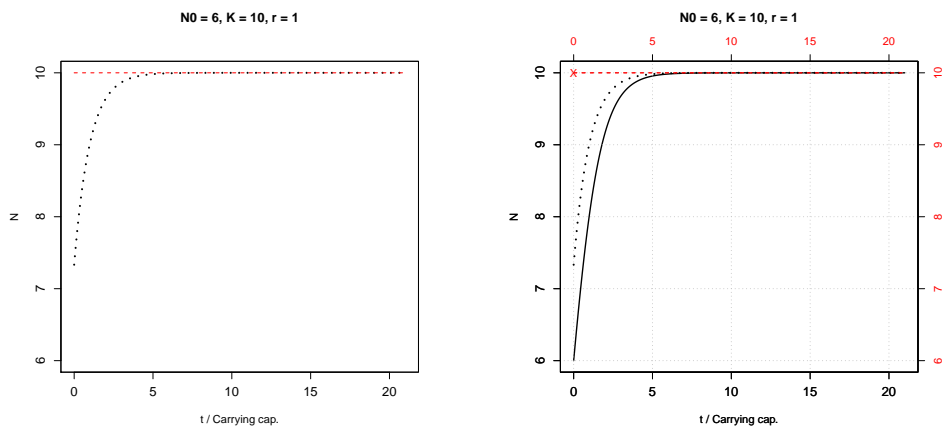


Fig. 3.: The logistic equation advanced by Verhulst, case $K/2 < N_0 < K$, in dotted line the constant carrying capacity

2.2. Generalized delay logistic equation (GDLE)

In this section a generalized form of the logistic equation with a time-delay in the carrying capacity is presented based on the general model advanced by Yukalov, Yukalova and Sornette [9]. The general model is

$$N'(t) := \frac{dN(t)}{dt} = \gamma N(t) - \frac{CN(t)^2}{K(t)}, \quad t > 0 \quad (2.8)$$

with initial function

$$N(t) = f(t), \quad t < 0 \quad (2.9)$$

and initial value

$$N(0) = N_0, \quad t = 0 \quad (2.10)$$

where $K(t) := A + BN(t - \tau)$ is the generalized carrying capacity. $K(t)$ is a linear function of the physical system, the two contributions are A , interpreted as the original carrying capacity and B , interpreted as the factor that controls the current carrying capacity as a proportion of the historical development. The term γ can be interpreted as the net birth rate with respect to the death or equivalently the net gain rate versus the loss. The term C represents the net competition versus the cooperation. The two terms regulate the two causes of development, γ controls the individual composing the system while C controls the interactions or the feedback to the system. We restrict our study to the following parameterization,

$$-\infty < B < \infty, \quad A \geq 0, \quad C > 0, \quad \gamma > 0, \quad \text{and} \quad \tau > 0$$

The general model in (2.8) is simplified by dimensionless transformations and parameter reduction techniques [9]. In short, we define a new variable $x(t)$ that is the normalized development $N(t)$ by an effective rate in order to obtain a relative development $x(t) := N(t)/N_{eff}$. Additionally the two terms γ and C are absorbed with the constants A and B . This transformation redefines new constants a and b , respectively, furthermore a new time scale s is chosen, below the transformations

$$\frac{dx(t)}{dt} \frac{1}{\gamma} = x(t) - \frac{x^2(t)}{\frac{\gamma}{CN_{eff}}K(t)} = x(t) - \frac{x^2(t)}{\frac{\gamma}{CN_{eff}}A + \frac{\gamma}{C}Bx(t - \tau)}, \quad t > 0 \quad (2.11)$$

with the new time scale $s = t\gamma$ and defining

$$\begin{aligned} a &:= \frac{\gamma}{CN_{eff}}A \\ b &:= \frac{\gamma}{C}B \\ k(s) &:= a + bx(s - \tau) \end{aligned} \quad (2.12)$$

the equation reduces to

$$\frac{dx(s)}{ds} = x(s) - \frac{x^2(s)}{a + bx(s - \tau)} = x(s) - \frac{x^2(s)}{k(s)}, \quad s > 0 \quad (2.13)$$

with initial function

$$x(s) = f(s), \quad s < 0 \quad (2.14)$$

and initial value

$$x(0) = f(0) = x_0, \quad s = 0 \quad (2.15)$$

The general model is obtained from Eq. (2.8) with a time dependent carrying capacity, however we can focus our study on Eq. (2.13) without loss of generality. In the following sections the new time scale s will be replaced with t .

3

Step-wise construction

In this section a guide to 'construct' the solution of Eq. (2.13) will be provided. The construction will follow a method of steps. The steps will be of length τ , we will stop at the second step since the other steps can easily be inferred by recursion. The method of steps is motivated by the semi-analytical solution within each step and the graphical interpretation of the dynamics versus its carrying capacity. In the first section of this chapter details on the analytical motivation are provided, then the system is resolved semi-analytically and finally new insights on the regime classification are shown for a specific parameterization ($a = 0$).

3.0.1. Analytical motivation

The analytical rationale for using the method of steps is that for any $t \in [n\tau, (n+1)\tau]$, for all positive integers n , the term $x(t - \tau)$ is known. Therefore Eq. (2.13) within each interval reduces to a logistic equation with a time varying carrying capacity $k(t)$. For example in Eq. (2.13) within the interval $[0, \tau]$ the term $x(t - \tau)$ is known and corresponds to the initial function $f(t)$.

More precisely, the function $f(t)$ defined on $[-\tau, 0]$ allows to define the evolution of the delayed system in (2.13) on the new interval $[0, \tau]$ as the solution of the following first order non-linear differential equation

$$\frac{dx_1(t)}{dt} = x_1(t) - \frac{x_1^2(t)}{a + bf(t)}, \quad t \in [0, \tau] \quad (3.1)$$

with initial value $x_1(0) = x(0) = f(0)$. The following step consists in

constructing the solution in the interval $[\tau, 2\tau]$ by using the solution of x_1 defined in $[0, \tau]$. The solution is given by

$$\frac{dx_2(t)}{dt} = x_2(t) - \frac{x_2^2(t)}{a + bx_1(t - \tau)}, \quad t \in [\tau, 2\tau] \quad (3.2)$$

with initial value $x_2(\tau) = x(\tau) = x_1(\tau)$. The remaining steps can be performed by recursion.

$$dx_n(t)/dt = x_n(t) - x_n^2(t)/(a + bx_{n-1}(t - \tau)), \quad t \in [(n - 1)\tau, n\tau] \quad (3.3)$$

Finally, the solution of the delayed system (2.13) at a given time t is the collection of n trajectories, where n satisfies $(n - 1)\tau \leq t \leq n\tau$. We are now capable to solve our system by successive integration as it will be seen below.

3.0.2. Semi-analytical solution

In this section it will be shown that the method of steps allows to find the solution of (2.13) in a semi-analytical form. We first start solving the model advanced by Yukalov, Yukalova and Sornette [9] (GDLE) for the regime "gain and competition" where the initial function is constant over the initial interval, namely

$$f(t) = x_0 = \text{constant}, \quad t \in [-\tau, 0] \quad (3.4)$$

Then, another initial function that simplifies the semi-analytical form is proposed

$$f(t) = (e^t - a)/b, \quad t \in [-\tau, 0] \quad (3.5)$$

In the constant initial function case (GDLE) it will be demonstrated that the semi-analytical form for the two first steps can be simplified in a closed form expression.

To find the closed form solution of (2.13) in the first step it is sufficient to recall that at the initial stage for $t \in [-\tau, 0]$ the solution of the system is the constant x_0 . Therefore (3.1) reduces to the standard logistic equation below,

$$x_1(t) = \frac{x_0 k_1 e^t}{k_1 + x_0 (e^t - 1)}, \quad t \in [0, \tau] \quad (3.6)$$

where $k_1 = a + bx(t - \tau) = a + bx_0$, as defined in (2.12).

For the following steps where $t > \tau$ the carrying capacity $k_n(t)$ is not

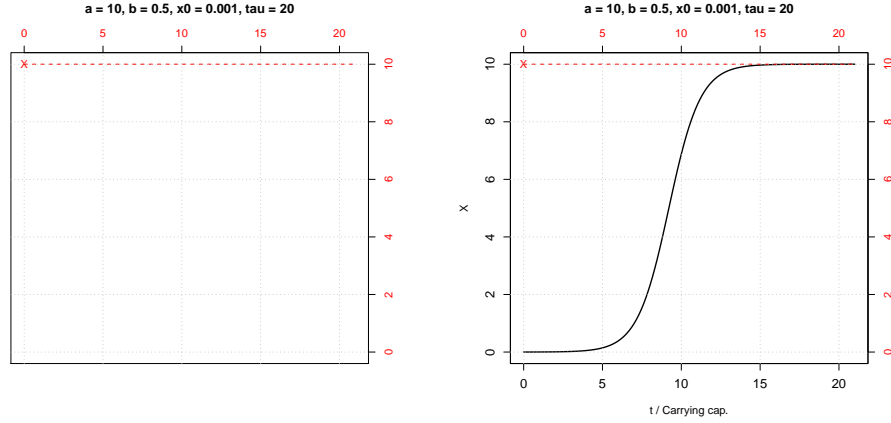


Fig. 4: Construction of the first step of the generalized delay logistic equation. (left) Carrying capacity in dotted line $k_1 = a + b * x_0 \simeq 10$ for $t \in [0, \tau]$. (right) The logistic equation in solid line.

constant anymore but it is a linear transformation of $x_{n-1}(t)$, namely

$$k_n(t) = a + bx_{n-1}(t - \tau), \quad t \in [(n-1)\tau, n\tau]$$

The solution of the delayed system (2.13) of the n -th trajectory reduces to the following autonomous first-order non-linear differential equation

$$\begin{cases} dx_n/dt = x_n(t) - x_n^2(t)/(k_n(t)), & t \in [(n-1)\tau, n\tau] \\ x_n(t) = x_n(\tau) = x_{n-1}(\tau), & t = \tau \end{cases} \quad (3.7)$$

The n -th differential equation can be solved for $t \in [(n-1)\tau, n\tau]$ by dividing both sides of (3.7) by $x_n^2(t)$, leading to

$$\frac{dx_n(t)}{dt} \frac{1}{x_n^2} = \frac{1}{x_n} - \frac{1}{k_n(t)}$$

by direct substitution of $v = 1/x_n$ and noting that $\frac{dv}{dt} = -\frac{1}{x_n^2} \frac{dx_n}{dt}$, we obtain

$$v' + v = \frac{1}{k_n(t)}$$

we choose the integrating factor e^t to finally obtain

$$x_n(t) = \frac{e^t}{\int_{(n-1)\tau}^t \frac{e^s}{k_n(s)} ds + C_n}, \quad \text{with } C_n = \frac{e^{(n-1)\tau}}{x_{n-1}((n-1)\tau)}, \quad t \in [(n-1)\tau, n\tau] \quad (3.8)$$

where C_n is the constant of integration determined with the continuity condition $x_n((n-1)\tau) = x_{n-1}((n-1)\tau)$. Note that (3.8) holds for arbitrary

initial functions.

Equipped with the new semi-analytical solution (3.8) for arbitrary initial functions, the second interval of the generalized delay logistic equation (GDLE) becomes analytically solvable. In the first interval $[0, \tau]$ the integration constant is $C_1 = 1/x_0$ and the solution is given in (3.6), see Fig. 4. In the second interval $[\tau, 2\tau]$ the integration constant is $C_2 = e^\tau/x_1(\tau)$, leading to the solution below, see Fig. 5

$$x_2(t) = \frac{x_1(\tau)K_2e^{(t-\tau)}}{K_2 + x_1(\tau)\left(e^{(t-\tau)} - 1 + b\frac{k_1(k_1-x_0)}{x_0K_2}\ln(A(t))\right)}, \quad t \in [\tau, 2\tau] \quad (3.9)$$

with

$$A(t) := \frac{e^{(t-\tau)}x_0K_2 + a(k_1 - x_0)}{k_1^2} \quad \text{and} \quad K_2 := a + b(a + bx_0) = \lim_{t \rightarrow \infty} x_2(t) \quad (\neq k_2(t))$$

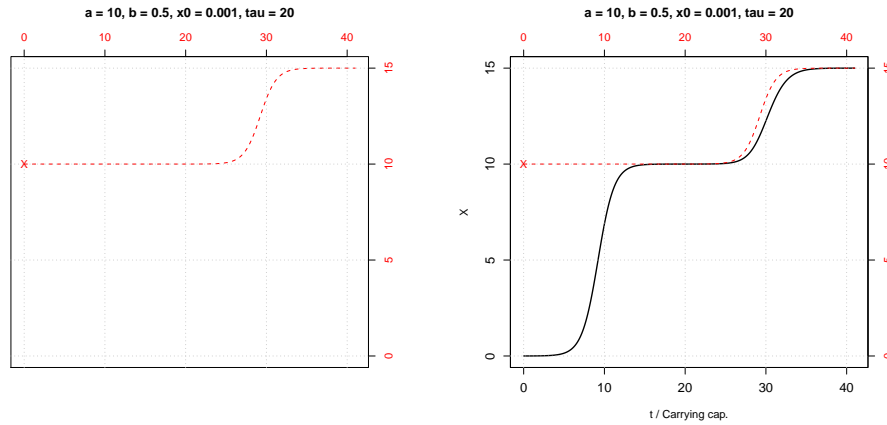


Fig. 5.: Construction of the second step of the generalized delay logistic equation. (left) Carrying capacity in dotted line $k_1 = a + b * x_0 \simeq 10$ for $t \in [0, \tau]$ and $k_2(t) = a + b * x_1(t - \tau)$ for $t \in [\tau, 2\tau]$. (right) The solution of the generalized delay logistic equation in solid line.

Note the similarity with (3.6), the only extra term is the factor containing the logarithm $b\frac{k_1(k_1-x_0)}{x_0K_2}\ln(A(t))$. The extra term is zero when $K_2 \neq 0$ and either $b = 0$, $k_1 = a + bx_0 = 0$, $k_1 = x_0$ or $A(t) = 1$. The first case $b = 0$ the delay is removed and we return to the standard logistic equation, the second case $k_1 = 0$ and third case correspond to the two trivial solutions $x_2(t) = x_1(t) = x(t) = 0$ and $x_2(t) = x_1(t) = x(t) = x_0$, respectively. Finally, the fourth case $A(t)$ is always equals to one for $t = \tau$, therefore in the second interval for $t \approx \tau$ the solution behaves

like a logistic equation with a new carrying capacity K_2 . However, as t becomes closer to 2τ , $A(t)$ might move far from one, therefore breaking the structural similarity with the logistic equation.

More generally, instead of choosing a constant initial function $f(t) = x_0$ one could have chosen an arbitrary initial function. The natural candidate is the exponential function that would simplify the integrand in (3.10). Let us chose $f(t) = (e^t - a)/b$ or equivalently $k_1 = e^t$, the evolution of the system is exponential at its origins for $t \leq \tau$. Recall that the first step is

$$x_1(t) = \frac{e^t}{\int_0^t \frac{e^s}{k_1} ds + C_1}, \quad \text{with } C_1 = \frac{1}{f(0)}, \quad (3.10)$$

the solution in the interval defined in $[0, \tau]$ is given by

$$x_1(t) = \frac{e^t}{t + (1 - a)/b} \quad (3.11)$$

Note that the solution in the first interval is not a logistic equation as in the case where $k_1 = a + bx_0$.

The former results might suggest that we could study the different regimes by focusing on the semi-analytical solution 3.8, below a functional approach.

Functional approach and insights from the semi-analytical solution

In order to study the system (3.8) from a functional viewpoint one could transform it into a simpler equivalent expression. In order to simplify the notation from (3.8) and without loss of generality the index n is removed and the coefficient C_n is replaced by $1 = f(0)$, leading to the following equation,

$$x(t) = \frac{e^t}{\int_0^t \frac{e^s}{k(s)} ds + 1} \quad (3.12)$$

Let us define $X(t) := \frac{e^t}{x(t)}$ and $Y(t) := \frac{e^t}{k(t)}$ for $x(t)$ and $k(t)$ non zeros. Solving equation (3.12) is equivalent to solve

$$dX/dt = Y(t), \quad \text{where } Y(t) = \frac{e^t}{k(t)} = \frac{e^t}{a + b \frac{e^{t-\tau}}{X(t-\tau)}} \quad (3.13)$$

For the particular case $a = 0$ or $t \rightarrow \infty$, the equation reduces to

$$\frac{dX}{dt} = \frac{X(t - \tau)}{be^{-\tau}},$$

which has a solution in terms of the Lambert functions W_k . The characteristic equation is

$$r = \frac{1}{be^{-\tau}}e^{-r\tau}, \quad \text{with solutions} \quad r_k = \frac{1}{\tau}W_k\left(\frac{\tau}{be^{-\tau}}\right), \quad (3.14)$$

The solution to the system (3.13) is given by (refer to [3])

$$X(t) = C_0e^{-t/\tau} + C_1e^{r_0t} + C_2e^{r_1t} + C_3e^{rt} + \sum_{k=1}^{\infty} e^{p_k t} [C_{1k}\cos(q_k t) + C_{2k}\sin(q_k t)] \quad (3.15)$$

where $r_k = p_k \pm iq_k$ are complex solutions of (3.14) and C_{i_k} , $i = 1, 2$ are arbitrary constants while C_j , $j = 0, 1, 2, 3$ depend on e^τ/b and τ as follows,

The first case is for $e^\tau/b < -1/(\tau e) \iff -\tau e^{\tau+1} < b < 0$

$$X(t) = \sum_{k=1}^{\infty} e^{p_k t} [C_{1k}\cos(q_k t) + C_{2k}\sin(q_k t)]$$

The second case for $-1/(\tau e) < e^\tau/b < 0 \iff b < -\tau e^{\tau+1}$

$$X(t) = C_1e^{r_0t} + C_2e^{r_1t} + \sum_{k=1}^{\infty} e^{p_k t} [C_{1k}\cos(q_k t) + C_{2k}\sin(q_k t)]$$

where C_1, C_2 are arbitrary and r_0, r_1 are real roots.

The last case is for $e^\tau/b > 0 \iff b > 0$

$$X(t) = C_3e^{rt} + \sum_{k=1}^{\infty} e^{p_k t} [C_{1k}\cos(q_k t) + C_{2k}\sin(q_k t)]$$

where C_3 is arbitrary and r are the real root. Finally, the solution of (3.12) is given by

$$x(t) = \frac{e^t}{X(t)}$$

In summary, we can characterize the system, in the particular case $a = 0$, with the trivial exponential functions. The first and the third case are the only relevant to us since the second case is unrealistic; $b \ll -1$ would lead to a rapid destruction of the resources (recall $\tau > 0$).

4

Regime classification

For a particular set of parameters, we have seen how to construct the two first steps of the generalized delay logistic equation in Fig. 4 and Fig. 5, the constructions of the solutions in each step was guided by the carrying capacity. In other words at the n -th step $x_n(t)$ was constructed considering the carrying capacity $k_n(t)$. We concluded that the construction of the generalized delay logistic equation is not as simple as in 2.1.1 because the carrying capacity is a function of time and the evolution of the system. However, to grasp the similarities with the standard logistic all the qualitatively classified regimes in Yukalov, Yukalova and Sornette [9] will be presented and complemented by a topological study. This new perspective will open the door of a new qualitative classification appropriate for population development.

The general analysis of the generalized delay logistic equation when gain prevails over loss and competition prevails over cooperation leads to different qualitative solutions depending on the system of parameters. Below we follow the same classification and nomenclature as in Yukalov, Yukalova and Sornette [9]. The general logistic equation is given by

$$\frac{dx(t)}{dt} = x(t) - \frac{x^2(t)}{a + bx(t - \tau)}, \quad t > 0 \quad (4.1)$$

with initial function

$$x(t) = x_0, \quad t < 0 \quad (4.2)$$

and initial value

$$x(0) = f(0) = x_0, \quad t = 0 \quad (4.3)$$

The system has two stationary solutions $x_{s1}^* = 0$ and $x_{s2}^* = \frac{a}{1-b}$, the first solution is unstable for $a > 0, \tau > 0$ and any b , however the second solution is stable in the following parameter regions [9]:

$$\begin{aligned} a \geq 0, & \quad 0 < b < 1, & \quad \tau \geq 0, & \quad or \\ a > x_0|b|, & \quad -1 < b < 0, & \quad \tau \geq 0, & \quad or \\ a > 0, & \quad b < -1, & \quad t < \tau_0, & \end{aligned} \quad (4.4)$$

where $\tau_0 = \frac{1}{\sqrt{b^2-1}} \arccos(1/b)$. In addition, the point $x_{s2}^* = \frac{a}{1-b}$ becomes a stable center for

$$a > 0, \quad b < -1, \quad \tau = \tau_0 \quad (4.5)$$

In the following section an exhaustive classification is presented complemented by system-carrying capacity diagrams. The trajectories represented in the diagrams are always $x_1(t)$ and $x_2(t)$ constituting the collection of the two first trajectories of $x(t)$ in the two first intervals (refer to Chapter 3).

Punctuated unlimited growth

The regime classification punctuated unlimited growth corresponds to the parameterization

$$a \geq 0, \quad b \geq 1, \quad \tau \geq 0$$

the carrying capacity driven by the parameter b is non-decreasing, the parameters imply $k_1 = a + bx_0 > x_0$, in other words the system starts below its initial carrying capacity. $x(t)$ grows towards infinity by steps of duration τ .

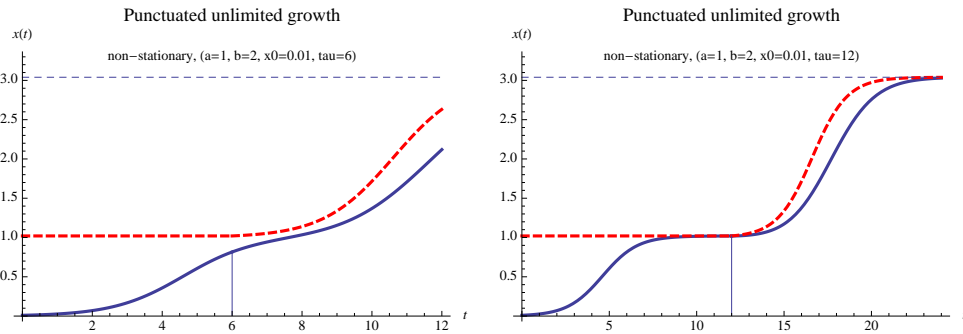


Fig. 6.: In solid line the trajectory of the system for $t \in [0, 2\tau]$. In thick dotted line the carrying capacity $k_1 \approx 1$ and $k_2(t) = a + b * x_1(t - \tau)$. The horizontal dotted line corresponds to $x(t) = K_2$ and the vertical segment is $t = \tau$. (left) strictly increasing growth, (right) non-decreasing growth.

Two sub-regime appear in Fig. 6, on the left regime the system approaches the carrying capacity without stabilizing while in the right regime a plateau is clearly visible interrupting the growth. The thin dashed line is $x(t) = K_2$ defined in (3.9) and is analogous to the standard carrying capacity but for the second interval.

If the system is used to model population growth, then the population in this regime would follow a sigmoid approaching the growing carrying capacity inherited from the past generations. The factor b controlling the carrying capacity production leads to the scenario of unlimited population growth.

Punctuated growth to a stationary level

Punctuated growth to a stationary level is characterized by the following parameterization

$$a > (1 - b)x_0, \quad 0 \leq b < 1, \quad \tau \geq 0$$

the carrying capacity is again non-decreasing, the parameters imply that the system starts below its initial carrying capacity and its stationary point, $x_0 < k_1 < x_{s2}^* = a/(1 - b)$. $x(t)$ grows towards the stationary solution $x(t) = x_{s2}^*$.

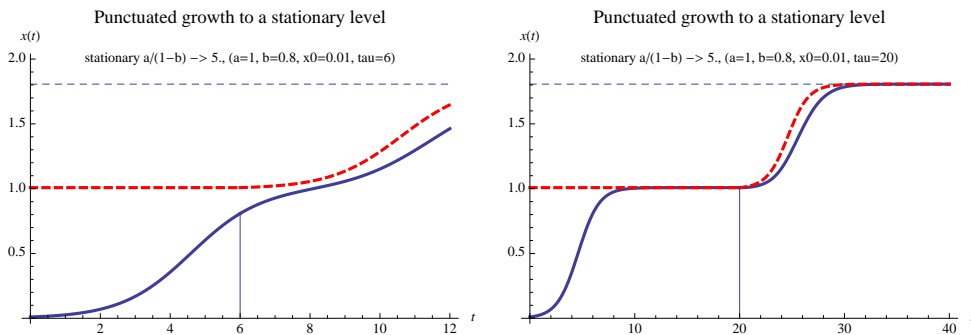


Fig. 7.: In solid line the trajectory of the the regime punctuated growth to a stationary level for $t \in [0, 2\tau]$. In dotted line the carrying capacity.

In the population growth model the population in this regime would follow a sigmoid approaching a long term equilibrium $x_{s2}^* = a/(1 - b)$ being the long term carrying capacity. The carrying capacity is reduced at each step due to the factor $b < 1$ controlling the carrying capacity production.

Punctuated decay to a stationary level

The regime punctuated decay to a stationary level corresponds to the parameterization

$$0 \leq a \leq (1 - b)x_0, \quad 0 \leq b < 1, \quad \tau \geq 0$$

the carrying capacity driven by the parameter b is non-increasing, the parameters imply that the system starts above the stationary level and its initial carrying capacity, $0 < k_1 = a + bx_0 < x_{s2}^* < x_0$. $x(t)$ decays towards a stationary solution $x(t) = x_{s2}^* = a/(1 - b)$. Fig. 16 shows the behavior of the trajectories $x_1(t)$ and $x_2(t)$.

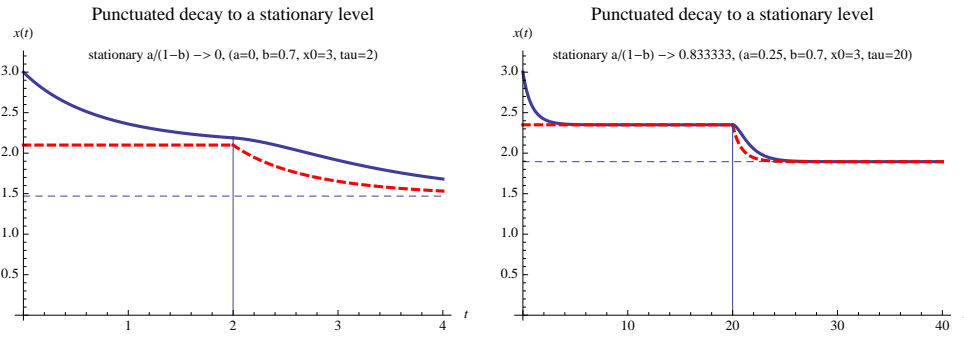


Fig. 8.: In solid line the trajectory of the the regime punctuated decay to a stationary level for $t \in [0, 2\tau]$. In dotted line the carrying capacity.

In the population growth model the population in this regime would be permanently over-shooting its resources leading to a decay in population size approaching a long term equilibrium $x_{s2}^* = a/(1 - b)$. The carrying capacity in this model is destroyed with time $b < 1$.

Punctuated alternation to a stationary level

The regime classification punctuated alternation to a stationary level corresponds to following the parameterization

$$a > |b|x_0, \quad -1 \leq b < 0, \quad \tau \geq 0$$

the carrying capacity oscillates and two sub-cases arise depending on the initial behavior of the system either grow or decay. When $x_0 < x_{s2}^* < k_1$ then $x(t)$ grows initially, see top of Fig. 9. On the other hand when $x_0 > x_{s2}^* > k_1 > 0$ then $x(t)$ decays initially, see bottom of Fig. 9. $x(t)$ tends in both cases towards the stationary solution $x(t) = x_{s2}^* = a/(1 - b)$ through a sequence of alternations.

The population in these cases oscillates around the limit and by time to time overshoots its carrying capacity in a reversible way. The carrying capacity in this model is destroyed throughout the time $-1 \leq b < 0$.

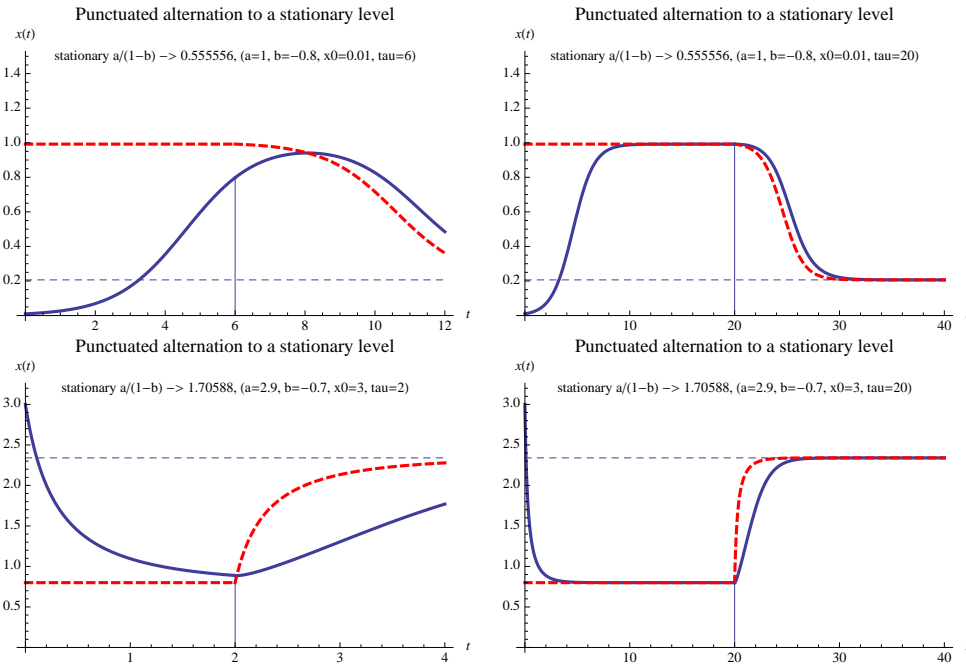


Fig. 9.: In solid line the trajectory of the regime punctuated alternation to a stationary level for $t \in [0, 2\tau]$. In dotted line the carrying capacity. (top) Initial growth, (bottom) initial decay.

Punctuated alternation to a finite time death

The regime punctuated alternation to a finite time death corresponds to the parameterization

$$a > |b|x_0, \quad b < -1, \quad \tau > \tau_0$$

when $x_0 < k_1$ then $x(t)$ grows initially, see Fig. 10. On the other hand when $x_0 > k_1 > 0$ then $x(t)$ decays initially. $x(t)$ in both cases alternates until $x(t)$ becomes zero.

In the population model this regime corresponds to a population extinction. The carrying capacity in this model is destroyed throughout the time $b < -1$.

Everlasting non-decaying oscillations

The regime punctuated alternation to a finite time death corresponds to the parameterization

$$a > |b|x_0, \quad b < -1, \quad \tau = \tau_0$$

$x(t)$ oscillates without attenuation around x_{s2^*} ,

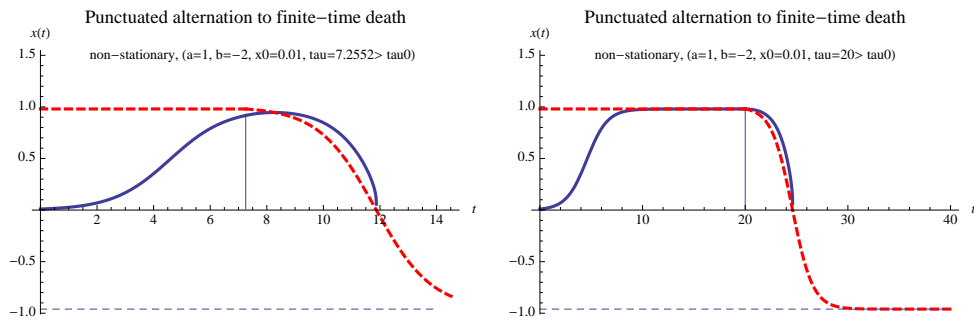


Fig. 10.: In solid line the trajectory of the the regime punctuated alternation to a finite time death for $t \in [0, 2\tau]$. In dotted line the carrying capacity.

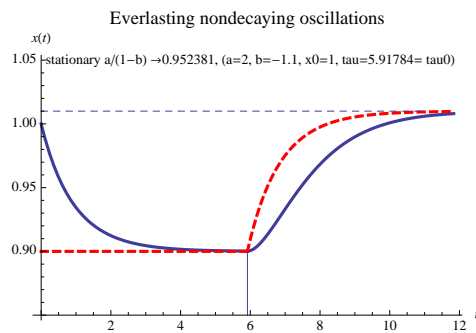


Fig. 11.: In solid line the trajectory of the the regime everlasting non-decaying oscillations for $t \in [0, 2\tau]$. In dotted line the carrying capacity.

For the population model, the delay τ_0 cuts the growth or the decline of the population before the destructive factor $b < -1$ is influential.

Oscillatory approach to a stationary level

The regime oscillatory approach to a stationary level corresponds to the parameterization

$$a > |b|x_0, \quad b < -1, \quad \tau < \tau_0$$

when $x_0 < k_1$ then $x(t)$ grows initially, see Fig. 12. On the other hand when $x_0 > k_1 > 0$ then $x(t)$ decays initially. $x(t)$ in both cases tends towards the stationary solution $x(t) = x_{s2}^* = a/(1-b)$ through a sequence of alternations.

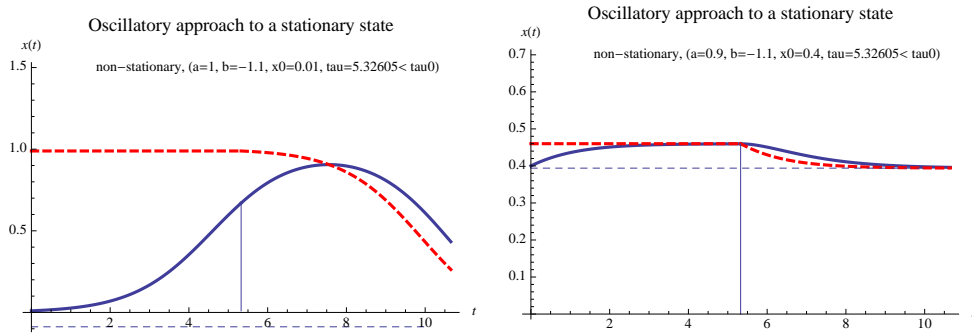


Fig. 12.: In solid line the trajectory of the the regime oscillatory approach to a stationary level for $t \in [0, 2\tau]$. In dotted line the carrying capacity.

This classification corresponds to a population overshoot followed by oscillations.

Growth to a fixed finite-time singularity

This regime is one out of three regimes where the initial carrying capacity k_1 is negative. Only the growth to a fixed finite-time singularity is exposed for illustrative purposes. However, the three cases are not relevant when studied in parallel to their carrying capacities. Indeed, the three cases share an initial negative carrying capacity ($k_1 < 0$), leading to the following system

$$\frac{dx_1(t)}{dt} = x_1(t) + \frac{x_1^2(t)}{|k_1|}, \quad t \in [0, \tau]$$

the system is not assimilable to a logistic equation anymore, therefore the term carrying capacity is not applicable and the diagrams system-carrying capacity are of no use.

The regime growth to a fixed finite-time singularity corresponds to the parameterization

$$a < |b|x_0, \quad b < 0, \quad \tau > t_c$$

where t_c is a finite time singularity obtained from the first step (3.6). Let t_c be the time at which $x_1(t_c) \rightarrow \pm\infty$, then t_c satisfies

$$t_c = \{t > 0 : k_1 + x_0(e^t - 1) = 0\} \iff t_c = \ln\left(1 - \frac{k_1}{x_0}\right)$$

With a sufficiently large time-lag τ but not necessarily smaller than t_c , the system diverges at the singularity t_c .

4.0.3. Qualitative classification

The system-carrying capacity diagrams suggest that a new classification can be inferred from the different approaches of the system to their carrying capacity. This qualitative classification seems natural from a population dynamics viewpoint. Indeed, it is natural to view the population growth as a physical system limited by a physical capacity.

In order to classify the different regimes we will restrain to the first interval $t \in [0, \tau]$ using the factor R defined as the proportion of the carrying capacity with respect to the population at the end of the cycle. From (3.6) we define

$$R := \frac{k_1}{x_1(\tau)} = e^{-\tau} \left(\frac{k_1}{x_0} - 1 \right) + 1, \quad (\tau > 0, x_0 > 0) \quad (4.6)$$

Having R as a measure of population reserves, five classification arise

R > 1: "satiety"

The population grows approaching the carrying capacity from below without consuming all the physical resources and with some resources margin. The condition $R > 1$ is satisfied for

$$x_0 < k_1, \quad \tau > 0, \quad x_0 > 0$$

The initial value x_0 is therefore below the carrying capacity k_1 , as illustrated in Fig. 13.

The reserves at the end of the cycle are not completely consumed by the population before the next resource regeneration cycle. The population grows with almost no stagnation, figuratively, we call this regime "satiety".

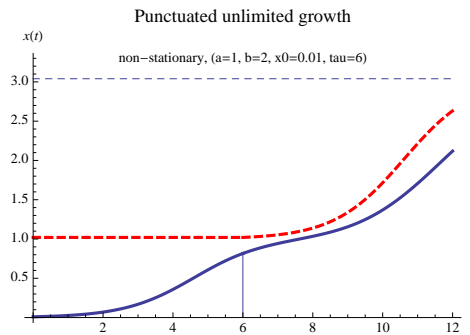


Fig. 13.: Population approaching the carrying capacity from below. Population not consuming all the physical resources

R = 1: "gluttony"

The population approaches the carrying capacity during a growth or decline phase. In the neighborhood of τ the physical resources equal the carrying capacity. The condition $R = 1$ is satisfied for

$$\tau \rightarrow +\infty \quad \text{or} \quad x_0 = k_1, \quad \tau > 0, \quad x_0 > 0$$

these conditions suggest the population never consumes entirely its resources within the first interval, since $x_0 = k_1$ leads to the trivial solution $x(t) = x_0 = k_1$ and $\tau \rightarrow +\infty$ is excluded. However, for $\tau \gg 1$ we can appreciate the creation of a population saturation (plateau) very close to its carrying capacity as illustrated in Fig. 14 and Fig. 15.

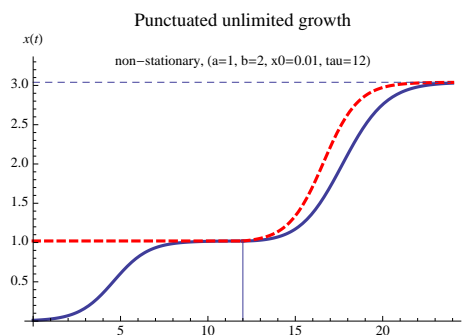


Fig. 14.: $R \approx 1$ with $e^{-\tau} \approx 0$, the population approaches the carrying capacity from below consuming almost all the physical resources until the next cycle of resources regeneration.

If the reserves at the end of the cycle are entirely consumed the population becomes constant and no regeneration of the resources follows. This is classified as "gluttony"

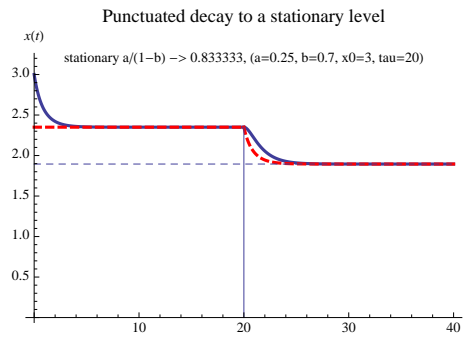


Fig. 15.: $R \approx 1$ with $e^{-\tau} \approx 0$, the population decline approaching the carrying capacity from above.

$0 < R < 1$: "starvation"

The population declines approaching the carrying capacity from above. The condition $0 < R < 1$ is satisfied for

$$0 < k_1 < x_0, \quad \tau > 0, \quad x_0 > 0$$

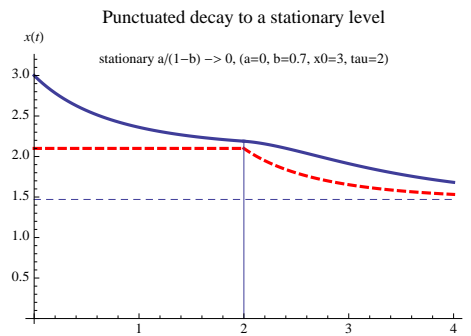


Fig. 16.: In solid line the trajectory of the regime punctuated decay to a stationary level for $t \in [0, 2\tau]$. In dotted line the carrying capacity.

No reserves are available and this regime can be interpreted as a population extinction. The regime is labelled "starvation".

$R = 0$, trivial solution $x(t) = 0$

The condition $R = 0$ is satisfied for

$$k_1 = 0 \quad x_0 = k_1, \quad \tau > 0, \quad x_0 > 0$$

leading to the trivial solution $x(t) = k_1 = 0$.

R < 0, not applicable

The condition $R < 0$ is satisfied for

$$x_0 < \frac{k_1}{1 - e^\tau}, \quad \tau > 0, \quad x_0 > 0$$

As a consequence of the conditions above the carrying capacity must be negative ($k_1 < 0$), therefore as mentioned earlier the parallel study population versus carrying-capacity is not applicable.

5

Discussion

In this master we showed how to obtain a semi-analytical solution to the GDLE via a method of steps. The results were supported by numerical computations either via numerical integration or analytical solutions in the two first intervals. In addition, the semi-analytical solution was used as a basis to a new regime classification for the special case $a = 0$. Then, we complemented the set of regimes described in [9] with diagrams using the generalized carrying capacity as the resource limit. The diagrams suggested a new view on the classification regimes that was topological. A first topological classification attempt was done via a newly defined resource factor R . However, the classification with R was not fully satisfactory due to the narrow view on solely the first interval. Indeed, the R factor did not take into consideration the full dynamics of the system. Even if R allowed to know the population approach to the carrying capacity at the specific point τ it did not allow to know the behavior of the population beyond τ . For instance, for $R > 1$ it was not possible to distinguish the three different regimes "Punctuated growth to a stationary level", "Oscillatory approach to a stationary level" or "Punctuated alternation to a finite time death". For future work and to tackle the limitations of R , one could explore another measure complementing R by assessing the trend of the reserves beyond τ . The trend could be captured by another ratio G representing the ratio of the reserves in the two first intervals, G could be defined as the following ratio

$$G := \frac{K_2/x_2(2\tau)}{k_1/x_1(\tau)}, \quad (\tau > 0, x_0 > 0) \quad (5.1)$$

where $x_2(t)$, K_2 are defined in (3.9) and $x_1(t)$, k_1 are given in (3.6). After some simplifications noting that $x_1(\tau) = x_2(\tau)$, we obtain the simplified expression

$$G = e^{-\tau} \left[\frac{K_2}{k_1} + x_1(\tau) \left[\frac{e^\tau - 1}{k_1} + b \left[\frac{1}{x_0} \frac{k_1}{K_2} - \frac{1}{K_2} \right] \ln[A(2\tau)] \right] \right] \quad (5.2)$$

where

$$A(2\tau) = \frac{x_0 K_2}{k_1 k_1} e^\tau + \frac{a}{k_1} \left(1 - \frac{x_0}{k_1} \right)$$

Similarly to R , the ratio G is measuring the trend of the reserves over the two time intervals, when G is greater than one it is indicating a sustained increase or decrease in reserves for $R > 0$ or $R < 0$, respectively. While when G is negative it indicates an increasing or decreasing oscillation in the reserves for $G < -1$ or $G > -1$, respectively.

Future work would involve studying the ratio above as well as finding analytically the points at which the carrying capacity and the system crosses in order to classify the regimes with respect to their intersections.

Aknowledgements

I wish to acknowledge Prof. Dr. Didier Sornette at the Chair of Entrepreneurial Risks (ETHZ) for his valuable comments and abundant ideas, I appreciated in particular his availability during the preparation of this master. I also want to express my appreciation to the team in the Chair of Entrepreneurial Risk whose comments were invaluable.



Numerical sim. with Mathematica

Dynamic parameterization (2 steps)

```
1 (** Integrating from scratch (slower) **)
2 C1 := 1/y0;
3 f1[t_, a_, b_, y0_] = (* For t in [0,Tau] *)
4   Exp[t]*(Integrate[Exp[t]/(a + b*y0), {t, 0, t},
5     Assumptions -> t >= 0] + C1)^-1 (* Use InputForm / TeXForm *)
6
7 C1 := 1/y0;
8 C2 := Exp[tau]/f1[tau, a, b, y0]
9 f2[t_, a_, b_, y0_, tau_] = (* For t in [Tau,2Tau] *)
10  Exp[t]/(Integrate[Exp[t]/(a + b*f1[t - tau, a, b, y0]), {t, tau, t},
11    Assumptions -> tau > 0 && t >= 0] + C2)
12
13 (** Already integrated (faster) **)
14 f1[t_, a_, b_, y0_] := E^t/(y0^(-1) + (-1 + E^t)/(a + b*y0)) (* For t in [0,Tau] *)
15 f2[t_, a_, b_, y0_, tau_] := (* For t in [Tau,2Tau] *)
16  E^t/(y0^(-1) + (-1 + E^tau)/(a +
17    b*y0) + ((E^t - E^tau)*y0*(a + a*b + b^2*y0) -
18    b^2*E^tau*(a^2 + a*(-1 + 2*b)*y0 + (-1 + b)*b*y0^2)*
19    Log[E^tau*(a + b*y0)^2] +
20    b^2*E^tau*(a^2 + a*(-1 + 2*b)*y0 + (-1 + b)*b*y0^2)*
21    Log[a^2*E^tau + a*((1 + b)*E^t + (-1 + b)*E^tau)*y0 +
22    b^2*E^t*y0^2])/(y0*(a + a*b + b^2*y0)^2)
23
24 k1[a_, b_, y0_] := a + b*y0;
25 K2[a_, b_, y0_] := a + b*(a + b*y0);
26 args[a_, b_, x0_, tau_] = "(a=" <> ToString[a] <> ", b=" <> ToString[b] <> ",
27   x0=" <> ToString[y0] <> ", tau=" <> ToString[tau] <> ")";
28
29 Manipulate[Show[
30   Plot[{f1[t, a, b, y0], k1[a, b, y0]}, {t, 0, tau},
31     PlotStyle -> {Directive[Thick], Directive[Thick, Dashed, Red]},
32     PlotRange -> Full, AxesLabel -> {t, y[t]},
33     PlotLabel -> Style[args[a, b, y0, tau], 10]],
34   Plot[{K2[a, b, y0]}, {t, 0, 2*tau},
35     PlotStyle -> {Directive[Dashed, Green]}, PlotRange -> Full],
36   Plot[{f2[t, a, b, y0, tau], a + b*f1[t - tau, a, b, y0]}, {t, tau, 2*tau},
37     PlotStyle -> {Directive[Thick],
38       Directive[Thick, Dashed, Red]}, PlotRange -> Full],
39   PlotRange -> All], {{tau, 10, "Tau"}, 1, 100, 1}, {{a, 0, "a"}, 0,
40   20}, {{b, 2, "b"}, -5, 5}, {{y0, .1, "y0"}, 0, 20},
41   SaveDefinitions -> True, FrameLabel -> "Logistic DDE"]
```


B

Numerical simulations with R

Numerical solution of GDLE for a general parameterization set

```
1 N = 10^3
2 T = seq(from=0, to=12, length.out=N)
3 M = 9 # number of parameters
4 A = seq(2, 0, length.out = M)
5 B = seq(-1.5, 1.5, length.out = M)
6 Sigma = expand.grid(c(1,-1),c(1,-1))
7 Sigma.lab = c("gain / competition", "loss / competition", "gain / cooperation", "loss / cooperation")
8 tau = 6
9 tau.idx = which(min(abs(T - tau)) == abs(T - tau))
10
11 for(sig in 1:nrow(Sigma))
12 {
13     png(filename = paste("plots/Rplot", "sig_", sig, ".png", sep = ""), width = 4800, height = 3600)
14     layout(matrix(c(rep(1, M), rep(2:(M*M+1))), nr = M + 1, byrow = TRUE), heights = c(0.3, rep(1, M-1)))
15     plot(1, axes = F, frame.plot = F, ylab = "", xlab = "")
16     text(x=1,y=1,Sigma.lab[sig], cex = 5)
17
18     print(Sigma.lab[sig])
19
20     for(a in A) for(b in B)
21     {
22         par(mar = c(5,4,6.5,4)+0.1)
23
24         # initial conditions
25         x0 = .5
26         y0 = a + b*x0
27
28         if((a < x0*abs(b)) & (b<0))
29         {
30             plot(0, 0, type = "n", main = "", xlab = "", ylab = "", cex.axis = 1.5, axes = F)
31             next
32         }
33
34         X = K = R = rep(NA, N) # node is a/(1-b) = 1.807229
35
36         # s < tau
37         X[1:(tau.idx+1)] = x0*y0*exp(T[1:(tau.idx+1)]) / (y0 + x0*(exp(T[1:(tau.idx+1)])-1))
38         R[1:(tau.idx)] = diff(X[1:(tau.idx+1)]) / ( diff(T[1:(tau.idx+1)]) * X[1:(tau.idx)] *Sigma[sig,1])
```

```

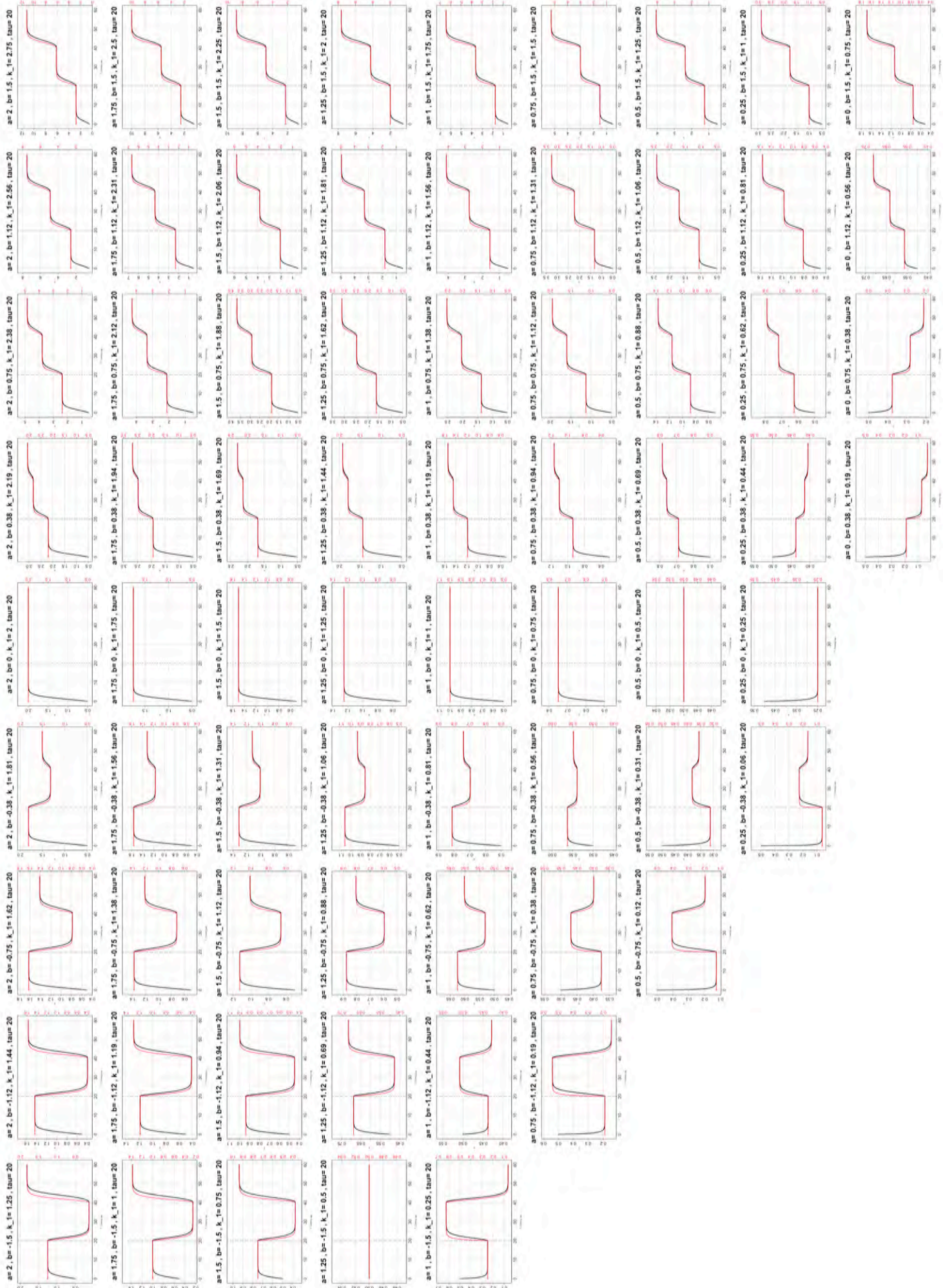
39     K[1:(tau.idx)] = a + b*x0
40
41     # s > tau
42     for(s in (tau.idx+1):(N-1))
43     {
44         stau = s - tau.idx
45         K[s] = (a + b*X[stau])
46         R[s] = (1 - Sigma[sig,2]/Sigma[sig,1]*X[s]/K[s]) # effective reproduction rate
47         X[s+1] = X[s] + diff(T)[s]*X[s]*Sigma[sig,1]*R[s]
48     }
49     X[N] = X[N-1]; R[N] = R[N-1]; K[N] = K[N-1]
50     out = rep(FALSE, N)
51     if(any(K <= 0) | any(X <= 0) )
52         out[round(min(which(K < 0.1)[1], which(X < 0.1)[1], na.rm = TRUE)*.999):N] = T
53     idx = is.na(X) | abs(X) == Inf | is.na(K) | abs(K) == Inf | out
54
55     lab = paste(paste("a=", round(a, digits = 2), ", b=", round(b, digits = 2),
56                 ", k_1=", round(a+b*x0, digits = 2), ", tau=", tau))
57
58     if(length(X[!idx])>0)
59     {
60         ylim = c(.9*min(X[!idx], K[!idx]), 1.1*max(X[!idx], K[!idx]))
61         if(.9*min(X[!idx], K[!idx]) < 0) ylim = range(0, ylim[-1])
62         plot(T[!idx], X[!idx], type = "l", main = lab, cex.main = 3.5, lwd = 3.5,
63             xlab = "T / Carrying cap.", ylab = "X", cex.axis = 2,
64             xlim = range(0,T[!idx]), ylim = ylim)
65         abline(v = T[tau.idx], lty = 2, lwd = 1.5)
66         grid(col = "grey", lwd = 2)
67
68         par(new = TRUE, mar = c(5,4,6.5,4)+0.1)
69         plot(T[!idx], K[!idx], col = 2, type = "l", xaxt="n", yaxt="n",
70             ylab = "", xlab = "", lwd = 2.5, xlim = range(0,T[!idx]), ylim = ylim)
71         points(T[!idx][1], K[!idx][1], col = 2, pch = "X")
72         axis(side = 4, col.axis = 2, cex.axis = 2)
73     } else plot(0, 0, type = "n", main = "", xlab = "", ylab = "", cex.axis = 1.5, axes = F)
74
75     }
76     dev.off()

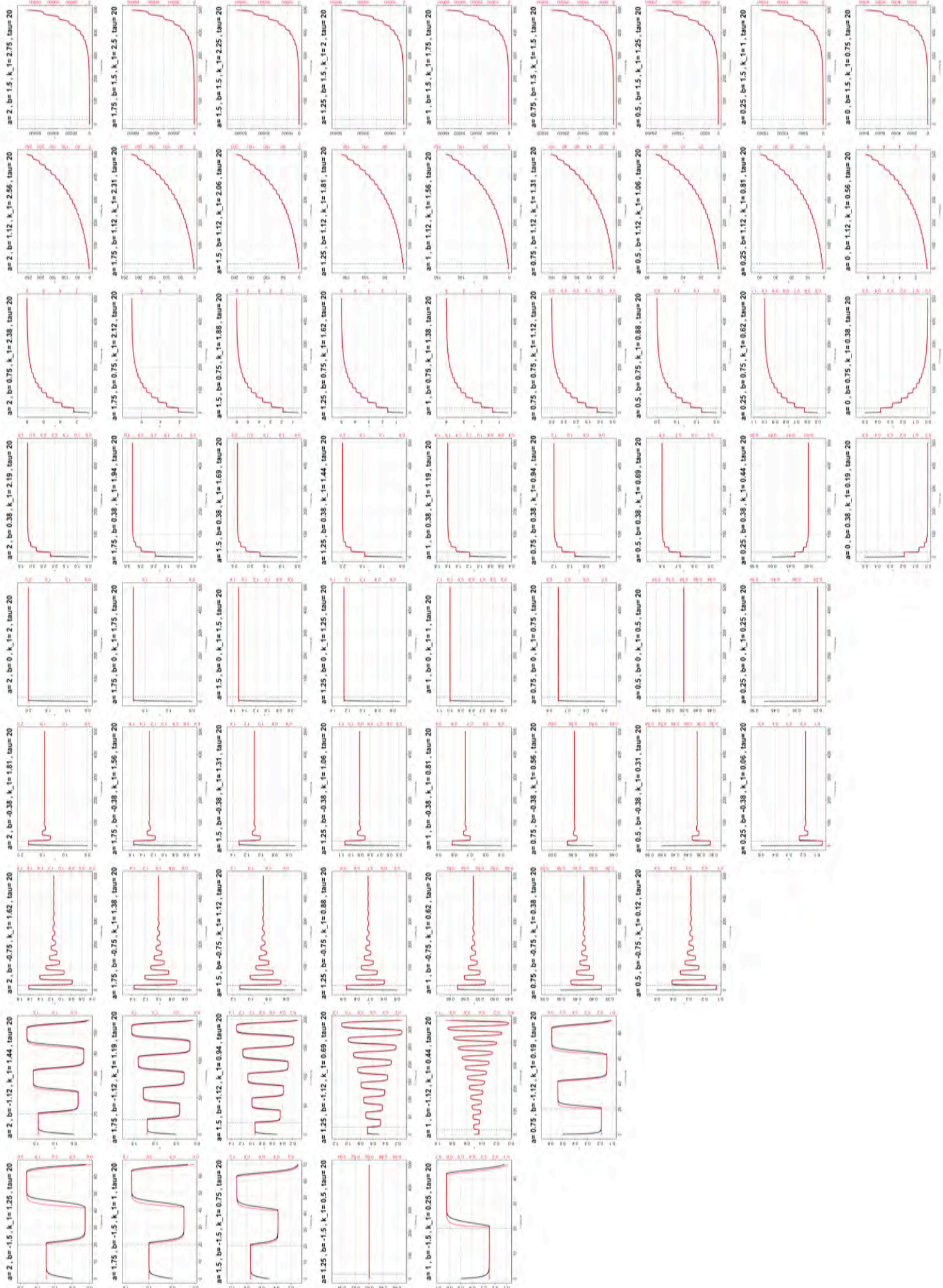
```

The code above produces the graphs below. The graphs represent the variety of different solutions excluding the solutions where $a < |b|x_0$, due to the negative carrying capacity. In this study we restrict to the case 'gain and competition', however the code can be used for all the cases advanced in Sornette [9].

gain / competition







Functions used for the graphs generation in this master

```

1 # utility functions
2 normalize <- function(x) (x-min(x))/(max(x) - min(x))
3 directionField<-function(
4   dydx,xlim=c(-1,1),ylim=c(-1,1),nxarrows=20,nyarrows=20,arrowHead=NA,
5   title=paste("Direction Field, dy/dx =", dydx),scale=1)
6 {
7   xinc = diff(xlim)/(nxarrows*2)
8   yinc = diff(ylim)/(nyarrows*2)
9   if(is.na(arrowHead)) arrowHead = par("pin")[1]/100
10  plot(0,xlim=xlim,ylim=ylim,type="n",xlab="x",ylab="y",main=title)
11  for(x in seq(xlim[1]+xinc, xlim[2]-xinc, length = nxarrows))
12  {
13    for(y in seq(ylim[1]+yinc, ylim[2]-yinc, length = nyarrows))
14    {
15      stan = atan(eval(parse(text=dydx)))
16      arrows(x-xinc*cos(stan)*scale, y-yinc*sin(stan)*scale,
17            x+xinc*cos(stan)*scale, y+yinc*sin(stan)*scale, length = arrowHead)
18      abline(h = 0, lty = 2, lwd = 0.1); abline(v = 0, lty = 2, lwd = 0.1)
19    }
20  }
21 }
22 simulateLogistic <- function(a, b, x0, r=1, tau, T, sig = 1, showAll = F)
23 {
24   if(tau >= max(T)) stop("tau too big")
25
26   Sigma = expand.grid(c(1,-1),c(1,-1))
27   Sigma.lab = c("gain / competition", "loss / competition",
28               "gain / cooperation", "loss / cooperation")
29
30   N = length(T)
31   tau.idx = which(min(abs(T - tau)) == abs(T - tau))[1]
32
33   # initial conditions
34   y0 = a + b*x0
35   X = K = R = rep(NA, N) # node is a/(1-b) = 1.807229
36
37   # s < tau
38   X[1:(tau.idx+1)] = (x0*y0*exp(r*T[1:(tau.idx+1)]) / (y0 + x0*(exp(r*T[1:(tau.idx+1)]-1)))
39   R[1:(tau.idx)] = diff(X[1:(tau.idx+1)]) / (diff(T[1:(tau.idx+1)]) * X[1:(tau.idx)] *Sigma[sig,1])
40   K[1:(tau.idx)] = a + b*x0
41
42   # s > tau
43   for(s in (tau.idx+1):(N-1))
44   {
45     stau = s - tau.idx
46     K[s] = (a + b*X[stau])
47     R[s] = r*(1 - Sigma[sig,2]/Sigma[sig,1]*X[s]/K[s]) # effective reproduction rate
48     X[s+1] = X[s] + diff(T)[s]*X[s]*Sigma[sig,1]*R[s]
49   }
50
51   X[N] = X[N-1]; R[N] = R[N-1]; K[N] = K[N-1]
52   idx.x = is.na(X) | abs(X) == Inf
53
54   if(!showAll) return(X)
55   else return(list(X = X, K = K))
56 }
57
58 plotLogisticDelay <- function(par, t, N = 10^3, log = F)
59 {
60   logY = ifelse(log, "y", "")
61
62   T = seq(min(t), max(t), length.out = N)
63   tau.idx = which(min(abs(T - par$tau)) == (T - par$tau))
64
65   out = simulateLogistic(a = par$a, b = par$b, x0 = par$x0, r = par$r,
66     tau = par$tau, T, sig = 1, showAll = TRUE)
67   X = out$X; K = out$K
68   idx.x = is.na(X) | abs(X) == Inf
69   idx.k = is.na(K) | abs(K) == Inf
70 }

```

```

71     if(length(X[!idx.x])>0 & length(K[!idx.k])>0)
72     {
73         ylim = range(X[!idx.x], K[!idx.k])
74         plot(T[!idx.x], X[!idx.x], type = "l", main = "", cex.main = 3, lwd = 2,
75             xlab = "t / Carrying cap.", ylab = "X", cex.axis = 1.2, log = logY, ylim = ylim)
76         par(new = TRUE)
77         plot(T[!idx.k], K[!idx.k], col = 2, lty = 2, type = "l", xaxt="n", yaxt="n", log = logY,
78             ylab = "", xlab = "", lwd = 1.5, ylim = ylim)
79         points(T[1], K[1], col = 2, pch = "X")
80         grid(col = "grey")
81         abline(v = T[tau.idx], lty = 3, lwd = 1)
82         axis(side = 4, col.axis = 2, cex.axis = 1)
83         axis(side = 3, col.axis = 2, cex.axis = 1)
84         print("xlim, ylim")
85         print(range(T[!idx.x]))
86         print(ylim)
87     }
88     print("Summary X / K")
89     print(summary(X))
90     print(summary(K))
91 }
92 plotLogisticDelaySteps <- function(par, t, N = 10^3, log = F, show = "all")
93 {
94     logY = ifelse(log, "y", "")
95
96     T = seq(min(t), max(t), length.out = N)
97     tau.idx = which(min(abs(T - par$tau)) == (T - par$tau))
98
99     out = simulateLogistic(a = par$a, b = par$b, x0 = par$x0, r = par$r, tau = par$tau,
100        T, sig = 1, showAll = TRUE)
101     X = out$X;      K = out$K
102     idx.x = is.na(X) | abs(X) == Inf
103     idx.k = is.na(K) | abs(K) == Inf
104
105     if(length(X[!idx.x])>0 & length(K[!idx.k])>0)
106     {
107         ylim = range(X[!idx.x], K[!idx.k])
108         if(show == "all")
109         {
110             plot(T[!idx.x], X[!idx.x], type = "l", main = "", cex.main = 3, lwd = 2,
111                 xlab = "t / Carrying cap.", ylab = "X", cex.axis = 1.2, log = logY, ylim = ylim)
112             par(new = TRUE)
113             plot(T[!idx.k], K[!idx.k], col = 2, lty = 2, type = "l", xaxt="n", yaxt="n", log = logY,
114                 ylab = "", xlab = "", lwd = 1.5, ylim = ylim)
115             points(T[1], K[1], col = 2, pch = "X")
116         } else if(show == "X") {
117             plot(T[!idx.x], X[!idx.x], type = "l", main = "", cex.main = 3, lwd = 2,
118                 xlab = "t / Carrying cap.", ylab = "X", cex.axis = 1.5, log = logY, ylim = ylim)
119         } else if(show == "K") {
120             plot(T[!idx.k], K[!idx.k], col = 2, lty = 2, type = "l", xaxt="n", yaxt="n", log = logY,
121                 ylab = "", xlab = "", lwd = 1.5, ylim = ylim)
122             points(T[1], K[1], col = 2, pch = "X")
123         }
124         grid(col = "grey")
125         abline(v = T[tau.idx], lty = 3, lwd = 1)
126         axis(side = 4, col.axis = 2, cex.axis = 1)
127         axis(side = 3, col.axis = 2, cex.axis = 1)
128         print("xlim, ylim")
129         print(range(T[!idx.x]))
130         print(ylim)
131     }
132
133     print("Summary X / K")
134     print(summary(X))
135     print(summary(K))
136 }
137 plotExp2 <- function(par, t, Np, ylim)
138 {
139     T = seq(min(t), max(t), length.out = Np)
140     N = par$N0 * exp(par$r*T)
141     N2 = par$K - (par$K - par$N0)^2/par$N0 * exp(-par$r*T)
142     K = rep(par$K, length(N))
143     plot(T, N2, type = "l", main = "", cex.main = 3, lwd = 3, lty = 3,
144         xlab = "t / Carrying cap.", ylab = "N", cex.axis = 1.2, ylim = ylim)

```

```

145     par(new = TRUE)
146     plot(T, K, col = 2, lty = 2, type = "l", xaxt="n", yaxt="n",
147         ylab = "", xlab = "", lwd = 1.5, ylim = ylim)
148   }
149   plotExp <- function(par, t, Np, ylim)
150   {
151     T = seq(min(t), max(t), length.out = Np)
152     N = par$N0 * exp(par$r*T)
153     N2 = par$K - (par$K - par$N0)^2/par$N0 * exp(-par$r*T)
154     K = rep(par$K, length(N))
155     plot(T, N, type = "l", main = "", cex.main = 3, lwd = 3, lty = 3,
156         xlab = "t / Carrying cap.", ylab = "N", cex.axis = 1.2, ylim = ylim)
157     lines(T, N2, lwd = 3, lty = 3)
158     par(new = TRUE)
159     plot(T, K, col = 2, lty = 2, type = "l", xaxt="n", yaxt="n",
160         ylab = "", xlab = "", lwd = 1.5, ylim = ylim)
161   }
162   plotVerhulst <- function(par, t, Np = 10^3, show = "all")
163   {
164
165     T = seq(min(t), max(t), length.out = Np)
166
167     getLogistic <- function(par, t)
168     {
169       N0 = par$N0; K = par$K; r = par$r
170       K * N0 * exp(r*t) / ( K + N0*( exp(r*t) - 1 ) ) # model logistic function
171     }
172
173     N = getLogistic(par, T); K = rep(par$K, length(N))
174
175     ylim = range(N, K)
176     if(show == "all") {
177       plot(T, N, type = "l", main = "", cex.main = 3, lwd = 2,
178         xlab = "t / Carrying cap.", ylab = "N", cex.axis = 1.2, ylim = ylim)
179       par(new = TRUE)
180       plot(T, K, col = 2, lty = 2, type = "l", xaxt="n", yaxt="n",
181         ylab = "", xlab = "", lwd = 1.5, ylim = ylim)
182       points(T[1], K[1], col = 2, pch = "X")
183     } else if(show == "inflection") {
184       plot(T, N, type = "l", main = "", cex.main = 3, lwd = 2,
185         xlab = "t / Carrying cap.", ylab = "N", cex.axis = 1.2, ylim = ylim)
186       par(new = TRUE)
187       plot(T, K, col = 2, lty = 2, type = "l", xaxt="n", yaxt="n",
188         ylab = "", xlab = "", lwd = 1.5, ylim = ylim)
189       points(T[1], K[1], col = 2, pch = "X")
190       ti = -1/par$r*log(par$N0/(par$K-par$N0))
191       points(ti, par$K/2, col = 1, pch = "X")
192     } else if(show == "N") {
193       plot(T, N, type = "l", main = "", cex.main = 3, lwd = 2,
194         xlab = "t / Carrying cap.", ylab = "N", cex.axis = 1.5, log = logY, ylim = ylim)
195     } else if(show == "K") {
196       plot(T, K, col = 2, lty = 2, type = "l", xaxt="n", yaxt="n", log = logY,
197         ylab = "", xlab = "", lwd = 1.5, ylim = ylim)
198       points(T[1], K[1], col = 2, pch = "X")
199     }
200     grid(col = "grey")
201     axis(side = 4, col.axis = 2, cex.axis = 1)
202     axis(side = 3, col.axis = 2, cex.axis = 1)
203     print("xlim, ylim")
204   }
205   getLogisticDelay <- function(par, t)
206   {
207     N = 5*10^2
208     T = seq(min(t), max(t), length.out = N)
209     tau = par$tauFact * (max(t) - min(t)) + min(t)
210     x = simulateLogistic(a = par$a, b = par$b, x0 = par$x0, r = par$r, tau = tau, T, sig = 1)
211     idx = unlist(lapply(t, function(x,T) which(min(abs(T - x)) == abs(T - x))[1], T = T)) # T[idx]
212     x[idx]
213   }

```

Optimization algorithm for fitting data to GDLE

The algorithm below is based on a modification of the Levenberg-Marquardt algorithm that minimize the sum of squares between the data and the model [6, 7]. We use heuristic initial conditions randomly generated to find an (local) optimal parameterization.

```

1 myOptim <- function(x,t,par,nseeds,fun,nprint=1,sdScale=10,maxiter=10^2,parCoverage=T)
2 {
3   # Antoine Beuchat
4   # Optimization based on nls.lm and random seeds
5   # IN: x: dependent variable, t: indep. variable,
6   # IN: par: list of parameters to pass to the function 'fun', nseeds: number of initial conditions guesses,
7   # IN: fun: model to fit function, nprint: 0 no print, 1 printing steps, sdScale: scale parameter
8   #      tuning the uncertainty of the seeds
9   # OUT: optimization result from nls.lm, plot with the fit
10
11   library(minpack.lm)
12
13   residFun <- function(p, t, observed, fun) observed - fun(p,t) # residual function
14
15   # starting values for parameters
16   res = list()
17   parAll = as.data.frame(matrix(0, nrow = nseeds, ncol = length(par),
18     dimnames = list(1:nseeds, names(par))), stringsAsFactors = F)
19
20   for(i in 1:nseeds)
21   {
22     parStart.red <- lapply(par, function(x,sdScale) x+rnorm(1,sd=sdScale*abs(x)), sdScale = sdScale)
23     if(!is.null(parStart.red$tauFact)) parStart.red$tauFact = par$tauFact # do not change it
24     parAll[i, ] = unlist(parStart.red)
25
26     # perform fit nls
27     res[[i]] <- nls.lm(par=parStart.red, fn = residFun, t = t, observed = x, fun = fun,
28       control = nls.lm.control(nprint = nprint, maxiter = maxiter)) # summary(res[[1]])
29
30   }
31
32   idx.exitOk = rep(T,nseeds) # unlist(lapply(res, function(x) x["info"] == 1 | x["info"] == 2 | x["info"] == 9))
33
34   if(any(idx.exitOk))
35   {
36     resOk = res[idx.exitOk]
37     idx.min = unlist(lapply(resOk, function(x) x["deviance"]))
38     if(nprint != 0) print(paste("deviance:", idx.min))
39
40     idx.min = idx.min[!is.na(idx.min)]
41     if(length(idx.min) < 1) stop("Only NAs deviations!")
42
43     # if the optim. does not exit properly summary(res) wil lead to an error
44     resOpt = resOk[[which(idx.min == min(idx.min))[1]]] # summary(res)
45
46     # plot model evaluated at final parameter estimates
47     x.sim = fun(as.list(coef(resOpt)), t)
48     if(any(!is.finite(x.sim))) ylim = NULL
49     else ylim = range(x.sim,x)
50     plot(t, x.sim, type = "l", col=2, lwd=2, ylim = ylim)
51     points(t, x, pch = "+", cex = 0.5)
52     grid()
53
54     if(parCoverage)
55     {
56       dev.new()
57       stripchart(parAll, vertical = T)
58     }
59
60     return(resOpt)
61
62   } else {
63     print("ERROR")
64   }
65 }

```

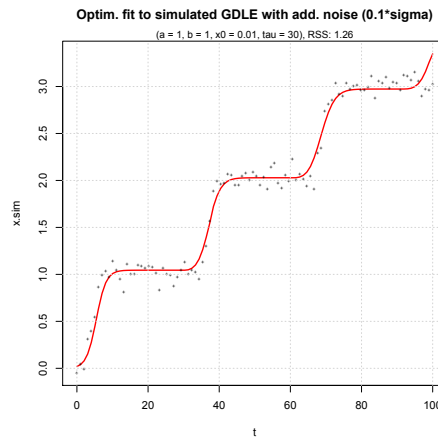
To clarify the usage of the optimization code above, we fit our GDLE model to a simulated GDLE with additive and multiplicative noise, respectively. The results are stable and promising for future works involving fitting problems. Even by adding a white noise with standard deviation of 10% of the system, the fit performs well with an $RSS = 1.25$. The fit for the multiplicative noise performs well but with a higher $RSS = 3.53$. The multiplicative noise consists in multiplying a white noise with mean 1 and standard deviation of 10% of the system standard deviation.

Synthetic GDLE with additive noise

```

1 # performance with simulated example
2 # trial how to recover the parameters x0, r, K when the t scale changes
3 par = list(a=1,b=1,x0=.01,r=1,tauFact=0.3)
4 N = 10^2
5 t = seq(0, 100, length.out = N)
6 tau = par$tauFact * (max(t) - min(t)) + min(t)
7 x = simulateLogistic(a = par$a, b = par$b, x0 = par$x0, r = par$r, tau = tau, t, sig = 1) # plot(t, x)
8 set.seed(1)
9 x.noisy = x + rnorm(length(x), sd = sd(x)*.1) # plot(x.noisy)
10 nseeds = 3
11 coef(myOptim(x.noisy,t,par,nseeds,fun=getLogisticDelay,sdScale=2))
12 title("Optim. fit to simulated GDLE with add. noise (0.1*sigma)")
13 mtext("(a = 1, b = 1, x0 = 0.01, tau = 30), RSS: 1.26", 3, cex = 0.9)}

```

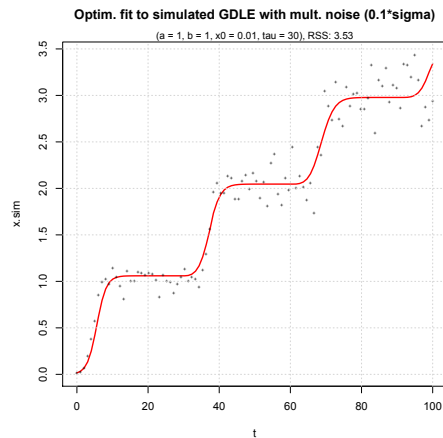


Synthetic GDLE with multiplicative noise

```

1
2 if(F) # performance with simulated example
3 {
4     # trial how to recover the parameters x0, r, K when the t scale changes
5     par = list(a=1,b=1,x0=.01,r=1,tauFact=0.3)
6     N = 10^2
7     t = seq(0, 100, length.out = N)
8     tau = par$tauFact * (max(t) - min(t)) + min(t)
9     x = simulateLogistic(a = par$a, b = par$b, x0 = par$x0, r = par$r, tau = tau, t, sig = 1) # plot(t, x)
10    set.seed(1)
11    x.noisy = x*rnorm(length(x), sd = sd(x)*.1, mean = 1) # plot(x.noisy)
12    nseeds = 3
13    coef(myOptim(x.noisy,t,par,nseeds,fun=getLogisticDelay,sdScale=2))
14    title("Optim. fit to simulated GDLE with mult. noise (0.1*sigma)")
15    mtext("(a = 1, b = 1, x0 = 0.01, tau = 30), RSS: 3.53", 3, cex = 0.9)
16 }

```



List of Figures

1.	Direction field of the Verhulst equation (2.1) and integral curves for different initial conditions. In this figure the carrying capacity is $K = 4$ and the initial conditions are from the bottom to the top $N_0 = \{0, 0.5, 1, \dots, 5\}$	4
2.	Construction of the logistic equation, case $N_0 < K/2$, in dotted line the carrying capacity. The first graph shows the limiting cases and the inflection point. The parameters of (2.1) are $K = 10, r = 1$ and $N_0 = 0.001$	6
3.	The logistic equation advanced by Verhulst, case $K/2 < N_0 < K$, in dotted line the constant carrying capacity	6
4.	Construction of the first step of the generalized delay logistic equation. (left) Carrying capacity in dotted line $k_1 = a + b * x_0 \simeq 10$ for $t \in [0, \tau]$. (right) The logistic equation in solid line.	11
5.	Construction of the second step of the generalized delay logistic equation. (left) Carrying capacity in dotted line $k_1 = a + b * x_0 \simeq 10$ for $t \in [0, \tau]$ and $k_2(t) = a + b * x_1(t - \tau)$ for $t \in [\tau, 2\tau]$. (right) The solution of the generalized delay logistic equation in solid line.	12
6.	In solid line the trajectory of the system for $t \in [0, 2\tau]$. In thick dotted line the carrying capacity $k_1 \simeq 1$ and $k_2(t) = a + b * x_1(t - \tau)$. The horizontal dotted line corresponds to $x(t) = K_2$ and the vertical segment is $t = \tau$. (left) strictly increasing growth, (right) non-decreasing growth.	16
7.	In solid line the trajectory of the the regime punctuated growth to a stationary level for $t \in [0, 2\tau]$. In dotted line the carrying capacity.	17
8.	In solid line the trajectory of the the regime punctuated decay to a stationary level for $t \in [0, 2\tau]$. In dotted line the carrying capacity.	18
9.	In solid line the trajectory of the the regime punctuated alternation to a stationary level for $t \in [0, 2\tau]$. In dotted line the carrying capacity. (top) Initial growth, (bottom) initial decay.	19

10.	In solid line the trajectory of the the regime punctuated alternation to a finite time death for $t \in [0, 2\tau]$. In dotted line the carrying capacity.	20
11.	In solid line the trajectory of the the regime everlasting non-decaying oscillations for $t \in [0, 2\tau]$. In dotted line the carrying capacity.	20
12.	In solid line the trajectory of the the regime oscillatory approach to a stationary level for $t \in [0, 2\tau]$. In dotted line the carrying capacity.	21
13.	Population approaching the carrying capacity from below. Population not consuming all the physical resources	23
14.	$R \approx 1$ with $e^{-\tau} \approx 0$, the population approaches the carrying capacity from below consuming almost all the physical resources until the next cycle of resources regeneration. . .	23
15.	$R \approx 1$ with $e^{-\tau} \approx 0$, the population decline approaching the carrying capacity from above.	24
16.	In solid line the trajectory of the the regime punctuated decay to a stationary level for $t \in [0, 2\tau]$. In dotted line the carrying capacity.	24

Bibliography

- [1] N. Eldredge. *Time frames*. Simon and Schuster New York, 1985.
- [2] A. Fowler. Asymptotic analysis of the delayed logistic equation when the delay is large. *IMA Journal of Applied Mathematics*, 28:41–9, 1982.
- [3] M. Ghil, I. Zaliapin, and S. Thompson. A delay differential model of ENSO variability: parametric instability and the distribution of extremes. *Nonlinear Processes in Geophysics*, 15(3):417–433, 2008.
- [4] S. Gould. *Natural Hist.*, 86:12–16, 1977.
- [5] S. Gould. *Science*, 216:380–387, 1982.
- [6] J. Mor. The Levenberg-Marquardt algorithm: implementation and theory. *Lecture notes in mathematics*, 630:105–116, 1977.
- [7] R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2009. ISBN 3-900051-07-0.
- [8] I. Wolfram Research. *Mathematica Edition: Version 7.0*. Champaign, Illinois, 2008.
- [9] V. I. Yukalov, E. P. Yukalova, and D. Sornette. Punctuated evolution due to delayed carrying capacity. *Physica D (1752-1767)*, Aug 2009.