



Universiteit
Leiden
The Netherlands

Tree plantation

Linden, P. van der

Citation

Linden, P. van der. (2008). *Tree plantation*.

Version: Not Applicable (or Unknown)

License: [License to inclusion and publication of a Bachelor or Master thesis in the Leiden University Student Repository](#)

Downloaded from: <https://hdl.handle.net/1887/3596834>

Note: To cite this publication please use the final published version (if applicable).

P. van der Linden

Tree Plantation

Bachelorscriptie, 24 september 2008

Scriptiebegeleider: dr. F.M. Spijksma



Mathematisch Instituut, Universiteit Leiden

Contents

1	The Model	1
1.1	Kolmogorov forward equations	1
1.2	Laplace Transform	3
1.2.1	Formal solutions for a plantation with 2 trees	5
1.2.2	Inverse Laplace	6
2	The eigenvalue method	10
2.1	The method	10
2.2	Plantation with 2 trees revisited	11
2.3	Numerical solutions	12
2.3.1	$N = 2$	13
2.3.2	$N = 30$	13
3	Quasi-stationary distribution	15
3.1	The idea	15
3.2	Numerical solutions for the QSD	16
4	Conclusion	19
	Appendices	21
A		21
B		22
C		24
D		26

Chapter 1

The Model

The model I am going to study is a model of a tree plantation. These trees have a chance of getting infected with a disease by insects or when they're exposed to leaves of a sick tree. This can be modeled by a continuous time Markov chain under suitable assumptions.

When modeling the plantation, I will consider a plantation with N trees, for N fixed. Infection of trees by insects, which occurs randomly, has probability $0 < \alpha < 1$. The rate of infection by spread of infected leaves is considered to be $0 < \beta < 1$.

When a tree is found to be infected, it gets cut down and it is replaced by a nursery seedling. The process of finding trees to be infected is considered to be a Poisson process with parameter λ . Nursery seedlings can't be infected by leaves at the nursery, because of protection nets. However nursery seedlings can get infected by insects, so when a nursery seedling replaces an infected tree, it too has a chance of being already infected. This is a very small probability $0 < p = 1 - q < 1$.

The evolution of the number of infected trees through time can be modeled as a continuous time Markov chain $\{X(t); t \geq 0\}$. $X(t)$ denotes the amount of infected trees at time $t (t \geq 0)$, with $X(0) = a \geq 1$. To be more precise, it can be viewed as a birth-death process with birthrates $\alpha + j\beta$ and death rates λq .

1.1 Kolmogorov forward equations

In this section I am going to derive the Kolmogorov forward equations for the model as described above. To do this I will first define the probabilities $p_j(t)$. These $p_j(t)$ are the probabilities of having j infected trees at time t , given that there are a infected trees at time 0.

With the $p_j(t)$ defined, I will now be able to derive the Kolmogorov forward equations. For that I will take a look at the relations between the probabilities $p_j(t + \delta t)$ and $p_j(t)$ for a small time interval of length δt . The probability of having j infected trees at time $t + \delta t$ starting with i infected trees is given by:

$$\sum_{i=1}^N P(X(t + \delta t) = j, X(t) = i) = \sum_{i=1}^N P(X(t + \delta t) = j | X(t) = i) P(X(t) = i) \quad (1.1)$$

Because the probability of two events occurring in a small time interval of length δt is very

small, I may omit quantities of order $o(\delta t)$. Now the summation can be written as follows:

$$\begin{aligned}
\sum_{i=1}^N P(X(t + \delta t) = j | X(t) = i) P(X(t) = i) \\
&= P(X(t + \delta t) = j | X(t) = j - 1) P(X(t) = j - 1) \\
&+ P(X(t + \delta t) = j | X(t) = j) P(X(t) = j) \\
&+ P(X(t + \delta t) = j | X(t) = j + 1) P(X(t) = j + 1).
\end{aligned} \tag{1.2}$$

Writing this summation out and omitting the quantities of order $o(\delta t)$ leads to the following system of equations:

$$\begin{aligned}
p_0(t + \delta t) &= p_0(t)(1 - \alpha\delta t) + p_1(t)\lambda q\delta t \\
&\vdots \\
p_j(t + \delta t) &= p_{j-1}(t)(\alpha + \beta[j - 1])\delta t \\
&+ p_j(t)(1 - \lambda q\delta t - [\alpha + \beta j]\delta t) + p_{j+1}(t)\lambda q\delta t \\
&\vdots \\
p_0(t + \delta t) &= p_{N-1}(t)(\alpha + \beta[N - 1])\delta t + p_N(t)(1 - \lambda q\delta t).
\end{aligned} \tag{1.3}$$

From this point it's easy to get to the Kolmogorov forward equations. The first thing I will do, is subtracting $p_j(t)$ from every equation for $j = 0, \dots, N$. Next I will divide by δt and I will take the limit for $\delta t \rightarrow 0$. This leads to the Kolmogorov forward equations:

$$\begin{aligned}
\frac{dp_0(t)}{dt} &= -\alpha p_0(t) + \lambda q p_1(t) \\
\frac{dp_1(t)}{dt} &= \alpha p_0(t) - (\lambda q + \alpha + \beta) p_1(t) + \lambda q p_2(t) \\
&\vdots \\
\frac{dp_j(t)}{dt} &= (\alpha + (j - 1)\beta) p_{j-1}(t) - (\lambda q + \alpha + \beta j) p_j(t) + \lambda q p_{j+1}(t) \\
&\vdots \\
\frac{dp_N(t)}{dt} &= (\alpha + (N - 1)\beta) p_{N-1}(t) - \lambda q p_N(t).
\end{aligned} \tag{1.4}$$

Now putting the left-hand side of the equations 0 leads to the equations for the finite $(N + 1)$ -state birth-death process with birth-parameters $\alpha + j\beta$ and death-parameter λq . For $\alpha > 0$ the distribution of $X(t)$ converges to the stationary distribution for $t \rightarrow \infty$. The stationary probabilities are denoted by $\pi_j = P(X(\infty) = j)$, meaning that in the long run the probability of having j infected trees is π_j . The stationary distribution $\pi = (\pi_j)$ for $j = 0, 1, \dots, N$, satisfies the following system of linear equations:

$$\begin{aligned}
0 &= -\alpha\pi_0 + \lambda q\pi_1 \\
0 &= (\alpha + (j - 1)\beta)\pi_{j-1} - (\lambda q + \alpha + j\beta)\pi_j + \lambda q\pi_{j+1} \quad j = 1, \dots, N - 1 \\
0 &= (\alpha + (N - 1)\beta)\pi_{N-1} - \lambda q\pi_N.
\end{aligned} \tag{1.5}$$

The solutions to this system are easily found by expressing π_j in terms of π_0 . π_0 can be determined from the normalizing equation $\sum_{j=0}^N \pi_j = 1$. The solutions of equations (1.5) are:

$$\begin{aligned}\pi_1 &= \frac{\alpha}{\lambda q} \pi_0 \\ \pi_2 &= \frac{\alpha(\beta + \alpha)}{(\lambda q)^2} \pi_0 \\ &\vdots \\ \pi_j &= \frac{\alpha(\beta + \alpha) \cdots (\alpha + \beta[j - 1])}{(\lambda q)^j} \pi_0,\end{aligned}\tag{1.6}$$

where

$$\pi_0 = \frac{1}{1 + \frac{\alpha}{\lambda q} + \frac{\alpha(\alpha+\beta)}{(\lambda q)^2} + \cdots + \frac{\alpha(\alpha+\beta)\cdots(\alpha+\beta[j-1])}{(\lambda q)^N}}.\tag{1.7}$$

1.2 Laplace Transform

In this section, I will use the Laplace transform to obtain a formal solution of the equations in (1.4). So let $\hat{p}_j(s)$ be the Laplace transform:

$$\hat{p}_j(s) = \int_0^\infty e^{-st} p_j(t) dt \quad Re(s) > 0, j = 0, 1, \dots, N.\tag{1.8}$$

Assuming that $X(0) = a$ and transforming the left-hand side of the first equation of (1.4) gives

$$\begin{aligned}\int_0^\infty \frac{dp_0(t)}{dt} e^{-st} dt &= [p_0(t)e^{-st}]_{t=0}^{t=\infty} + s \int_0^\infty p_0(t) e^{-st} dt \\ &= -p_0(0) + s\hat{p}_0(s).\end{aligned}\tag{1.9}$$

Transforming the right-hand side of the same equation gives

$$\int_0^\infty -\alpha p_0(t) e^{-st} + \lambda q p_1(t) e^{-st} dt = -\alpha\hat{p}_0(s) + \lambda q\hat{p}_1(s).\tag{1.10}$$

If $a \neq 0$, then $p_0(0) = 0$ and these two equations combine to

$$\lambda q\hat{p}_1 = (s + \alpha)\hat{p}_0.$$

If $a = 0$, then $p_0(0) = 1$ and the two equations combine to

$$\lambda q\hat{p}_1 - 1 = (s + \alpha)\hat{p}_0.$$

Next I will transform the j' th equation. This way I will find a general system of transformation with N trees. Transforming the left-hand side, yields, similarly to (1.9)

$$\int_0^\infty \frac{dp_j(t)}{dt} e^{-st} dt = -p_j(0) + s\hat{p}_j(s).$$

Transforming the right-hand side yields:

$$\begin{aligned} \int_0^\infty (\alpha + (j-1)\beta)p_{j-1}(t)e^{-st} - (\lambda q + \alpha + \beta j)p_j(t)e^{-st} + \lambda q p_{j+1}(t)e^{-st} dt \\ = (\alpha + (j-1)\beta)\hat{p}_{j-1} - (\lambda q + \alpha + j\beta)\hat{p}_j + \lambda q \hat{p}_{j+1} \end{aligned} \quad (1.11)$$

When $a \neq j$ these transformations combine to

$$(s + \alpha + \lambda q + j\beta)\hat{p}_j = (\alpha + (j-1)\beta)\hat{p}_{j-1} + \lambda q \hat{p}_{j+1},$$

and if $a = j$, then $p_j(0) = 1$, $p_j(t) = p_a(t)$ and the two equations combine to

$$(s + \alpha + \lambda q + a\beta)\hat{p}_a - 1 = (\alpha + (a-1)\beta)\hat{p}_{a-1} + \lambda q \hat{p}_{a+1}.$$

I only need to transform the last equation (the N -th equation) to complete the system. Transforming the left-hand side:

$$\begin{aligned} \int_0^\infty \frac{dp_N(t)}{dt} e^{-st} dt &= [p_N(t)e^{-st}]_{t=0}^{t=\infty} + s \int_0^\infty p_N(t)e^{-st} dt \\ &= -p_N(0) + s\hat{p}_N(s) \end{aligned} \quad (1.12)$$

Transforming the right-hand side

$$\begin{aligned} \int_0^\infty (\alpha + (N-1)\beta)p_{N-1}(t)e^{-st} - \lambda q p_N(t)e^{-st} dt \\ = (\alpha + (j-1)\beta)\hat{p}_{N-1}(s) - \lambda q \hat{p}_N(s) \end{aligned} \quad (1.13)$$

When $a \neq N$ these transformations combine to

$$(s + \lambda q)\hat{p}_N = (\alpha + (N-1)\beta)\hat{p}_{N-1} + \lambda q \hat{p}_N.$$

If $a = N$, then $p_N(0) = 1$ and the two equations combine to

$$(s + \lambda q)\hat{p}_N - 1 = (\alpha + (N-1)\beta)\hat{p}_{N-1} + \lambda q \hat{p}_N.$$

Thus, after having applied the Laplace transform to the Kolmogorov forward equations for a plantation with N trees, I've obtained a general linear system of N equations. The formal solutions of (1.4) can then be derived. Given $X(0) = a, a \neq 0, N$, this system is:

$$\begin{aligned} (s + \alpha)\hat{p}_0 &= \lambda q \hat{p}_1, \\ (s + \alpha + \lambda q + a\beta)\hat{p}_a - 1 &= (\alpha + (a-1)\beta)\hat{p}_{a-1} + \lambda q \hat{p}_{a+1}, \\ (s + \alpha + \lambda q + j\beta)\hat{p}_j &= (\alpha + (j-1)\beta)\hat{p}_{j-1} + \lambda q \hat{p}_{j+1}, \\ (s + \lambda q)\hat{p}_N &= (\alpha + (N-1)\beta)\hat{p}_{N-1} - \lambda q \hat{p}_N. \end{aligned} \quad (1.14)$$

Here $j = 1, \dots, a-1, a+1, \dots, N-1$.

1.2.1 Formal solutions for a plantation with 2 trees

In this section I will consider a plantation with 2 trees. For this model I will be studying the problem I've introduced earlier. The Kolmogorov forward equations (1.4) for the plantation with 2 trees are the following

$$\begin{aligned}\frac{dp_0(t)}{dt} &= -\alpha p_0(t) + \lambda q p_1(t) \\ \frac{dp_1(t)}{dt} &= \alpha p_0(t) - (\lambda q + \alpha + \beta) p_1(t) + \lambda q p_2(t) \\ \frac{dp_2(t)}{dt} &= (\alpha + \beta) p_1(t) - \lambda q p_2(t).\end{aligned}$$

First I am interested in the stationary distribution, because when I am searching for the formal solutions of $p_j(t)$, I can use it to check correctness of the formal solutions. Putting the left-hand side to 0, I will find the stationary distribution from (1.6) and (1.7). This yields:

$$\pi_0 = \frac{1}{1 + \frac{\alpha}{\lambda q} + \frac{\alpha(\alpha+\beta)}{(\lambda q)^2}} = \frac{(\lambda q)^2}{(\lambda q)^2 + \alpha(\alpha + \beta) + \alpha \lambda q},$$

as well as π_1 and π_2

$$\begin{aligned}\pi_1 &= \frac{\alpha}{\lambda q} \pi_0 = \frac{\alpha \lambda q}{(\lambda q)^2 + \alpha(\alpha + \beta) + \alpha \lambda q} \\ \pi_2 &= \frac{\alpha(\alpha+\beta)}{(\lambda q)^2} \pi_0 = \frac{\alpha(\alpha + \beta)}{(\lambda q)^2 + \alpha(\alpha + \beta) + \alpha \lambda q}.\end{aligned}$$

Now I've found the stationary distribution, I will transform the Kolmogorov forward equations with the Laplace transform. By applying the transform, I can make 3 choices for the starting values of the number of infected trees. I will transform the Kolmogorov forward equations for all three starting values, getting the following three systems (by 1.15):

Given that $X(0) = 0$

$$\begin{aligned}(s + \alpha)\hat{p}_0 - 1 &= \lambda q \hat{p}_1 \\ (s + \alpha + \lambda q + \beta)\hat{p}_1 &= \alpha \hat{p}_0 + \lambda q \hat{p}_2 \\ (s + \lambda q)\hat{p}_2 &= (\alpha + \beta)\hat{p}_1\end{aligned}\tag{1.15}$$

given that $X(0) = 1$

$$\begin{aligned}(s + \alpha)\hat{p}_0 &= \lambda q \hat{p}_1 \\ (s + \alpha + \lambda q + \beta)\hat{p}_1 - 1 &= \alpha \hat{p}_0 + \lambda q \hat{p}_2 \\ (s + \lambda q)\hat{p}_2 &= (\alpha + \beta)\hat{p}_1\end{aligned}\tag{1.16}$$

and given that $X(0) = 2$

$$\begin{aligned}(s + \alpha)\hat{p}_0 &= \lambda q \hat{p}_1 \\ (s + \alpha + \lambda q + \beta)\hat{p}_1 &= \alpha \hat{p}_0 + \lambda q \hat{p}_2 \\ (s + \lambda q)\hat{p}_2 - 1 &= (\alpha + \beta)\hat{p}_1.\end{aligned}\tag{1.17}$$

1.2.2 Inverse Laplace

Next I will calculate the formal solutions $p_j(t)$ for all three possibilities. I will compare these and I will take a look at the problem for this simple example. But before I will do that, I will introduce an inverse Laplace transform lemma, which I'll be needing to undo the Laplace transform and to obtain $p_j(t)$, $j = 0, 1, 2, \dots, N$. This theorem comes from [2] and is the following.

Lemma 1 *Let g be holomorphic except for a finite number of poles at a_1, \dots, a_n , and suppose that there exist constants M and k such that*

$$|g(p)| \leq M|p|^{-k} \quad \text{for } |p| \text{ large}$$

Then for $t > 0$ and $\sigma > \text{Re } a_j$ ($j = 1, \dots, n$),

$$\frac{1}{2\pi i} \lim_{R \rightarrow \infty} \int_{\sigma - iR}^{\sigma + iR} g(p)e^{pt} dp = \sum_{j=1}^n \text{res}\{g(p)e^{pt}; a_j\}.$$

This lemma guarantees that the inverse transform equals the sum of residues at poles a_1, \dots, a_n as long $|g(p)| \leq M|p|^{-k}$ for large p . This makes it a lot easier to find $p_j(t)$, $j = 1, 2, \dots, N$.

Next I will rewrite the systems (1.15), (1.16) and (1.17) and then I will apply the lemma. The first system I will rewrite, is (1.16) with $X(0) = 1$. For this system it's clear that:

$$\hat{p}_1 = \frac{s + \alpha}{\lambda q} \hat{p}_0 \tag{1.18}$$

$$\hat{p}_2 = \frac{\alpha + \beta}{s + \lambda q} \hat{p}_1 = \frac{(\alpha + \beta)(s + \alpha)}{\lambda q(s + \lambda q)} \hat{p}_0. \tag{1.19}$$

This follows from the first and the third equations of the system. Now looking at the second equation we get:

$$\lambda q \hat{p}_2 = \frac{(s + \lambda q + \alpha + \beta)(s + \alpha)}{\lambda q} \hat{p}_0 - \alpha \hat{p}_0 - 1.$$

Multiplying by λq yields

$$(\lambda q)^2 \hat{p}_2 = (s + \lambda q + \alpha + \beta)(s + \alpha) \hat{p}_0 - \lambda q \alpha \hat{p}_0 - \lambda q.$$

Combining this with (1.19) gives:

$$\frac{(\alpha + \beta)(s + \alpha)\lambda q}{s + \lambda q} \hat{p}_0 = (s + \lambda q + \alpha + \beta)(s + \alpha) \hat{p}_0 - \lambda q \alpha \hat{p}_0 - \lambda q,$$

which yields

$$\begin{aligned} [(s + \lambda q)(s + \lambda q + \alpha + \beta)(s + \alpha) - \alpha \lambda q(s + \lambda q) - (\alpha + \beta)(s + \alpha)\lambda q] \hat{p}_0 \\ = \lambda q(s + \lambda q). \end{aligned} \tag{1.20}$$

This immediately gives the following expression for \hat{p}_0

$$\begin{aligned}\hat{p}_0 &= \frac{\lambda q(s + \lambda q)}{s(s^2 + s(2\alpha + 2\lambda q + \beta) + \alpha(\lambda q + \alpha + \beta) + (\lambda q)^2)} \\ &= \frac{\lambda q(s + \lambda q)}{s(s + A + B)(s + A - B)},\end{aligned}$$

where $A = \alpha + \frac{\beta}{2} + \lambda q$ and $B = \sqrt{(\alpha + \beta)\lambda q + \frac{\beta^2}{4}}$.

Now I can apply the inverse Laplace lemma and obtain the formal solution for $p_0(t)$, $p_1(t)$ and $p_2(t)$. By virtue of this lemma, I should calculate the residues at the poles. For $\hat{p}_0(s)$ these poles are $0, -(A + B)$ and $-(A - B)$. So for $p_0(t)$

$$\begin{aligned}p_0(t) &= \lim_{s \rightarrow 0} s\hat{p}_0(s)e^{st} + \lim_{s \rightarrow -(A+B)} (s + A + B)\hat{p}_0(s)e^{st} + \lim_{s \rightarrow -(A-B)} (s + A - B)\hat{p}_0(s)e^{st} \\ &= \frac{(\lambda q)^2}{A^2 - B^2} + \frac{\lambda q(\lambda q - (A + B))}{2B(A + B)}e^{-(A+B)t} - \frac{\lambda q(\lambda q - (A - B))}{2B(A - B)}e^{-(A-B)t}.\end{aligned}$$

Now from (1.18) and (1.19) \hat{p}_1 and \hat{p}_2 can be transformed. Since the inverse transformations for \hat{p}_1 and \hat{p}_2 are similarly obtained as \hat{p}_0 , I will merely give $p_1(t)$ and $p_2(t)$ and I will not explicitly derive these:

$$\begin{aligned}p_1(t) &= \frac{\alpha\lambda q}{A^2 - B^2} \\ &+ \frac{(\alpha - (A + B))(\lambda q - (A + B))}{2B(A + B)}e^{-(A+B)t} \\ &- \frac{(\alpha - (A - B))(\lambda q - (A - B))}{2B(A - B)}e^{-(A-B)t}\end{aligned}\tag{1.21}$$

$$\begin{aligned}p_2(t) &= \frac{\alpha(\alpha + \beta)}{A^2 - B^2} + \frac{(\alpha + \beta)(\alpha - (A + B))}{2B(A + B)}e^{-(A+B)t} \\ &- \frac{(\alpha + \beta)(\alpha - (A - B))}{2B(A - B)}e^{-(A-B)t}.\end{aligned}\tag{1.22}$$

To check if these solutions make any sense, it is convenient to take a look at what happens, when the limits are taken for t to infinity. Because then we should end up in the stationary distribution. This means for the formal solutions, that the exponents will go to zero. A quick look at $p_0(t), p_1(t)$ and $p_2(t)$ confirms this.

Now left are the systems with $X(0) = 0$ and $X(0) = 2$. These are slightly more difficult because of the -1 in the first and the third equations, respectively. In case $X(0) = 2$, it doesn't make it much more difficult, but in case $X(0) = 0$, I will not obtain \hat{p}_0 first, but instead \hat{p}_2 . I will first take a look at the case with $X(0) = 2$.

The first and the last equations from (1.17) imply

$$\begin{aligned}\hat{p}_1 &= \frac{\alpha + s}{\lambda q}\hat{p}_0 \\ \hat{p}_2 &= \frac{\alpha + \beta}{s + \lambda q}\hat{p}_1 + \frac{1}{s + \lambda q} = \frac{(\alpha + s)(\alpha + \beta)}{(s + \lambda q)\lambda q}\hat{p}_0 + \frac{1}{(s + \lambda q)}.\end{aligned}$$

From this, \hat{p}_0 is easily derived (with the same steps as before):

$$\begin{aligned}\hat{p}_0 &= \frac{(\lambda q)^2}{s(s^2 + s(2\alpha + 2\lambda q + \beta) + \alpha(\lambda q + \alpha + \beta) + (\lambda q)^2)} \\ &= \frac{(\lambda q)^2}{s(s + A + B)(s + A - B)}.\end{aligned}$$

Using the inverse Laplace lemma again, I obtain $p_0(t)$, $p_1(t)$ and $p_2(t)$:

$$\begin{aligned}p_0(t) &= \frac{(\lambda q)^2}{A^2 - B^2} + \frac{(\lambda q)^2}{2B(A + B)}e^{-(A+B)t} - \frac{(\lambda q)^2}{2B(A - B)}e^{-(A-B)t} \\ p_1(t) &= \frac{\lambda q \alpha}{A^2 - B^2} + \frac{\lambda q(\alpha - (A + B))}{2B(A + B)}e^{-(A+B)t} - \frac{\lambda q(\alpha - (A - B))}{2B(A - B)}e^{-(A-B)t} \\ p_2(t) &= \frac{\alpha(\alpha + \beta)}{A^2 - B^2} + \frac{\lambda q(\alpha + \beta)(\alpha - (A + B))}{2B(A + B)(\lambda q - (A + B))}e^{-(A+B)t} \\ &\quad - \frac{\lambda q(\alpha + \beta)(\alpha - (A - B))}{2B(A - B)(\lambda q - (A - B))}e^{-(A-B)t} \\ &\quad + \left[\frac{(\alpha + \beta)(\alpha - \lambda q)}{(\lambda q - A - B)(A - B - \lambda q)} + 1 \right] e^{-\lambda q t}.\end{aligned}$$

This leaves only one last case, namely $X(0) = 0$. As said before I will not obtain \hat{p}_0 first, but instead I will determine \hat{p}_2 . I have made this choice, because the inverse Laplace theorem is no longer applicable if I would first determine \hat{p}_0 . The theorem is no longer applicable, because there are no M and k such that $|g(p)| \lesssim M|p|^{-k}$. Similarly as in the above

$$\hat{p}_2 = \frac{\alpha(\alpha + \beta)}{s(s + A + B)(s + A - B)}. \quad (1.23)$$

and so

$$\hat{p}_1 = \frac{\alpha(s + \lambda q)}{s(s + A + B)(s + A - B)} \quad (1.24)$$

$$\hat{p}_0 = \frac{\lambda q(s + \lambda q)}{s(s + A + B)(s + A - B)(s + \alpha)} + \frac{1}{s + \alpha}. \quad (1.25)$$

Using the inverse Laplace transformation again leads to the following expression the $p_j(t)$ that I am interested in:

$$\begin{aligned}p_0(t) &= \frac{(\lambda q)^2}{A^2 - B^2} + \frac{\alpha \lambda q(\lambda q - (A + B))}{2B(A + B)(\alpha - (A + B))}e^{-(A+B)t} \\ &\quad - \frac{\alpha \lambda q(\lambda q - (A - B))}{2B(A - B)(\alpha - (A - B))}e^{-(A-B)t} + \left[\frac{\lambda q(\lambda q - \alpha)}{(\alpha - A - B)(A - B - \alpha)} + 1 \right] e^{-\alpha t} \\ p_1(t) &= \frac{\lambda q \alpha}{A^2 - B^2} + \frac{\alpha(\lambda q - (A + B))}{2B(A + B)}e^{-(A+B)t} - \frac{\alpha(\lambda q - (A - B))}{2B(A - B)}e^{-(A-B)t} \\ p_2(t) &= \frac{\alpha(\alpha + \beta)}{A^2 - B^2} + \frac{\alpha(\alpha + \beta)}{2B(A + B)}e^{-(A+B)t} - \frac{\alpha(\alpha + \beta)}{2B(A - B)}e^{-(A-B)t}.\end{aligned}$$

Obviously this method requires a lot of time for calculations. In the case of $N = 2$ it is still doable, but for N larger it becomes harder to find $p_j(t)$, $j = 0, 1, 2, \dots, N$. Next to that,

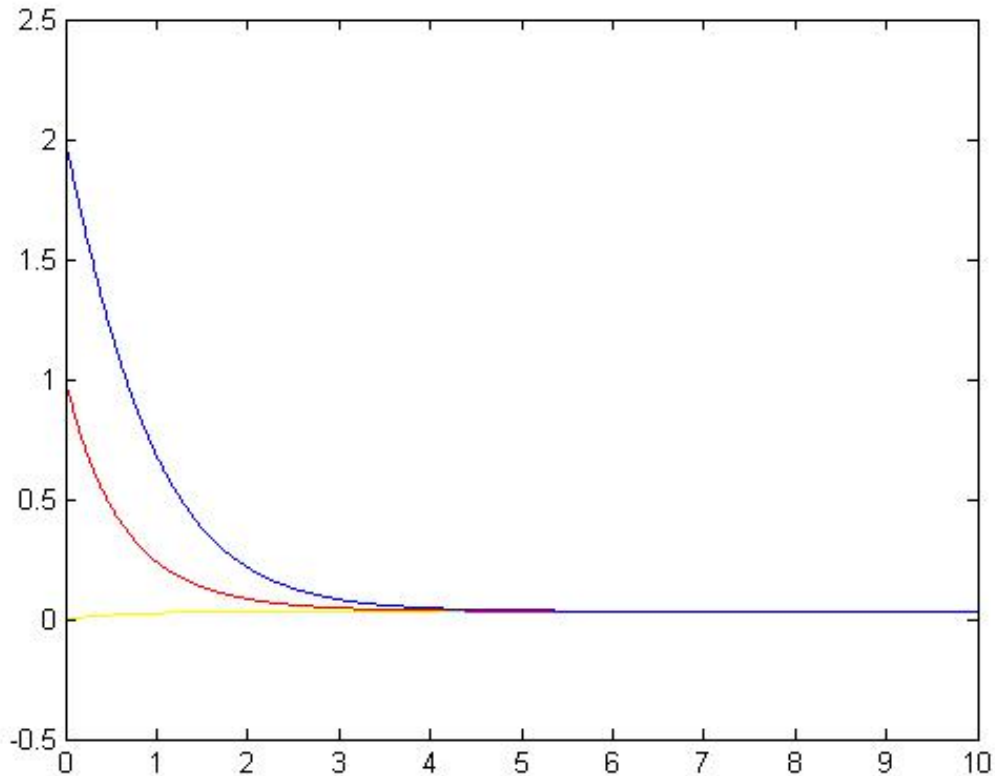


Figure 1.1: plots for $X(0) = 0, 1, 2$ and $T = 10$

every time N is increased, the number of starting states grows and the number of equations that needs to be solved grows even faster. Therefore this method is not the best to use for large N .

However, based on the p_j , $j = 0, 1, 2$, I have found in this example I made the following plots of the expected value of the number of infected trees. I have made plots for all three starting conditions with the following parameters: $\alpha = 0.05, \beta = 0.15, q = 0.9, \lambda = 2, T = 10$ days (These are the same parameters used by Gani and Stalls.).

Unfortunately, this example is too small to see any interesting effects I am looking for. Taking T larger gives a plot with all three starting case converging to the stationary distribution, which is to be expected. In the next chapter I will make plots for larger N , which will show interesting events. Since I am interested in the case of larger N , I need to find a way to resolve the problem of solving all these equations. Therefore I will try and use a numerical method in the next chapter. In this example it is also possible to use the computer, but still the transformations can't be done by computer, but have to be done by man. Matlab is used for plotting the expected value of infected trees against the time.

The method discussed in the next section still requires a lot of calculation, but with help of Matlab that method is better for numerical solutions. In the end I hope that method will give me more insight of what happens in case of larger values for N .

Chapter 2

The eigenvalue method

The work I've done in the previous sections were tedious, but easy to do for a plantation with a few trees. Of course I am more interested in the case of bigger plantations. To study bigger plantations, I need to come up with a method that is easier applicable for determining $p_j(t)$, $j = 0, 1, 2, \dots, N$. This method is the method of eigenvalues. The drawback of this method is, that it still requires a lot of calculations. However, Matlab can do a lot of these calculations in little time and give a better insight in what is really happening.

2.1 The method

In this section I will explain how the eigenvalue method works. The system (1.4) has the following solution

$$P(t) = e^{tQ}. \quad (2.1)$$

Here Q is a matrix which is easily found. This matrix consists of the birth- and deathrates of the birth-death-process described earlier. In the example this matrix is the following

$$Q = \begin{pmatrix} -\alpha & \alpha & 0 \\ \lambda q & -(\lambda q + \alpha + \beta) & \alpha + \beta \\ 0 & \lambda q & -\lambda q \end{pmatrix}. \quad (2.2)$$

In the case of more trees the Q -matrix looks similar, only bigger. Note that every row adds up to 0. Now finding $P(t)$ is still not easy to do, however Q can be diagonalised:

$$Q = UDU^{-1}. \quad (2.3)$$

Here U is the matrix of eigenvectors corresponding to the eigenvalues σ of Q . D is a diagonal matrix of eigenvalues σ of Q .

Writing Q in this form, makes it easier to find e^{tQ} . To see why it's easier, let's take a look at the definition of e^{tQ}

$$\begin{aligned} e^{tQ} &= \sum_{n=0}^{\infty} \frac{t^n Q^n}{n!} = \sum_{n=0}^{\infty} \frac{t^n (UDU^{-1})^n}{n!} \\ &= \sum_{n=0}^{\infty} \frac{t^n U D^n U^{-1}}{n!} = U \sum_{n=0}^{\infty} \frac{t^n D^n}{n!} U^{-1} = U e^{tD} U^{-1}. \end{aligned} \quad (2.4)$$

The product U and U^{-1} can be taken out of the summation, because when writing out $(UDU^{-1})^n$, U and U^{-1} cancel out, leaving UD^nU^{-1} . This only leaves e^{tD} . Because D is the diagonal matrix with the eigenvalues of Q , this power becomes:

$$e^{tD} = \begin{pmatrix} e^{\sigma_1 t} & 0 & \dots & 0 \\ 0 & e^{\sigma_2 t} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & e^{\sigma_N t} \end{pmatrix}.$$

Now it's easier to find $P(t)$. The drawback to this method is, that you have to find the eigenvalues, eigenvectors and the inverse of the matrix U . However I want to use this method to find solutions numerically, so this can be done by the computer.

2.2 Plantation with 2 trees revisited

Although I want to use the eigenvalue method for numerical purposes, I will be testing this method for the small plantation with two trees. I will do this to get a better understanding of the method and to see if this numerical solution works well enough.

The first thing to do now is finding the matrix Q . This Q is the same as in (2.2), since this served as an example for how a generator matrix Q should look like:

$$Q = \begin{pmatrix} -\alpha & \alpha & 0 \\ \lambda q & -(\lambda q + \alpha + \beta) & \alpha + \beta \\ 0 & \lambda q & -\lambda q \end{pmatrix}.$$

Now it's time to find the eigenvalues of Q . Of course, they are easily found by solving the following equation:

$$\det(\sigma I - Q) = 0. \quad (2.5)$$

In particular

$$\det(\sigma I - Q) = \det \begin{pmatrix} \sigma + \alpha & -\alpha & 0 \\ -\lambda q & \sigma + (\lambda q + \alpha + \beta) & -(\alpha + \beta) \\ 0 & -\lambda q & \sigma + \lambda q \end{pmatrix}.$$

Calculating this determinant leads to the following equation:

$$0 = \lambda q(-(\sigma + \alpha)(\alpha + \beta)) + (\sigma + \lambda q)((\sigma + \alpha)(\sigma + \lambda q + \alpha + \beta) - \alpha \lambda q)$$

Writing out the last brackets, this gives the next equation

$$0 = -\lambda q(\sigma + \alpha)(\alpha + \beta) + (\sigma + \lambda q)(\sigma + \alpha)(\sigma + \lambda q + \alpha + \beta) - \alpha \lambda q(\sigma + \lambda q).$$

This equation is precisely the same as the coefficient in front of \hat{p}_0 in equation (1.21). From the inverse Laplace transform I know the poles, which are precisely the zeroes that we are looking for. So the eigenvalues I find are the following:

$$\begin{aligned} \sigma_1 &= 0 \\ \sigma_2 &= -(A - B) \\ \sigma_3 &= -(A + B). \end{aligned} \quad (2.6)$$

Here A and B are the same as before: $A = \alpha + \frac{\beta}{2} + \lambda q$ and $B = \sqrt{(\alpha + \beta)\lambda q + \frac{\beta^2}{4}}$. Now it's time to find the eigenvectors belonging to σ_1 , σ_2 and σ_3 . These are found by solving the following systems:

$$Qx = \sigma_i x.$$

Here $x \in \mathbb{R}^3$ is a vector. Solving these systems is not hard and leads to the following eigenvectors:

$$\begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ \frac{\alpha - (A - B)}{\alpha(\lambda q - (A - B))} \\ \frac{\lambda q(\alpha - (A - B))}{\alpha(\lambda q - (A - B))} \end{pmatrix}, \begin{pmatrix} 1 \\ \frac{\alpha - (A + B)}{\alpha(\lambda q - (A + B))} \\ \frac{\lambda q(\alpha - (A + B))}{\alpha(\lambda q - (A + B))} \end{pmatrix}.$$

These eigenvectors correspond to σ_1, σ_2 and σ_3 respectively. Combining these three eigenvectors together, defines our matrix U :

$$U = \begin{pmatrix} 1 & 1 & 1 \\ 1 & \frac{\alpha - (A - B)}{\alpha(\lambda q - (A - B))} & \frac{\alpha - (A + B)}{\alpha(\lambda q - (A + B))} \\ 1 & \frac{\lambda q(\alpha - (A - B))}{\alpha(\lambda q - (A - B))} & \frac{\lambda q(\alpha - (A + B))}{\alpha(\lambda q - (A + B))} \end{pmatrix}.$$

The hardest part now is finding the inverse of U . In this case, finding the inverse consumes already a lot of time, so one could imagine how much time it will take for a larger matrix. Because of the fact that the inverse of U is hard to find, I will not do it here. So the method isn't good to use by hand. Since I am interested in the expected value of infected trees, I can do the following. Given $X(0) = a$ I will write the expectation as follows

$$\begin{aligned} \mathbb{E}(X(t)) &= 0 \cdot p_{a0}(t) + p_{a1}(t) + 2 \cdot p_{a2}(t) \\ &= c_1 + c_2 e^{\sigma_2 t} + c_3 e^{\sigma_3 t} = F(t). \end{aligned} \quad (2.7)$$

Here the $c_j, j = 1, 2, 3$, are linear combinations of the coefficients corresponding to the $p_j(t)$. From $F(t)$ I can now determine the c_j by calculating $F(0)$, $F'(0)$ and $F''(0)$. These are:

$$F(0) = p_{a1}(0) + 2p_{a2}(0) = c_1 + c_2 + c_3 \quad (2.8)$$

$$F'(0) = p'_{a1}(0) + 2p'_{a2}(0) = q_{a1} + q_{a2} = c_2 \sigma_2 + c_3 \sigma_3 \quad (2.9)$$

$$F''(0) = p_{a1}(0)'' + 2p_{a2}(0)'' = q_{a1}^2 + q_{a2}^2 = c_2 \sigma_2^2 + c_3 \sigma_3^2. \quad (2.10)$$

Now it's good to see that c_1 is easy to determine, since this is $\mathbb{E}(X(\infty))$, which is: $\pi_1 + 2\pi_2$. Now I have come to the point where it is still possible to find the expected value, without having to find U^{-1} .

2.3 Numerical solutions

In this section I am going to use Matlab to take a look at larger plantations. I will first find the numerical solutions of the expected value of the number of infected trees for $N = 2$. After that I will take a look at $N = 30$, since this is the number of trees used in the article of Gani and Stalls, for which I am studying the problem they have encountered.

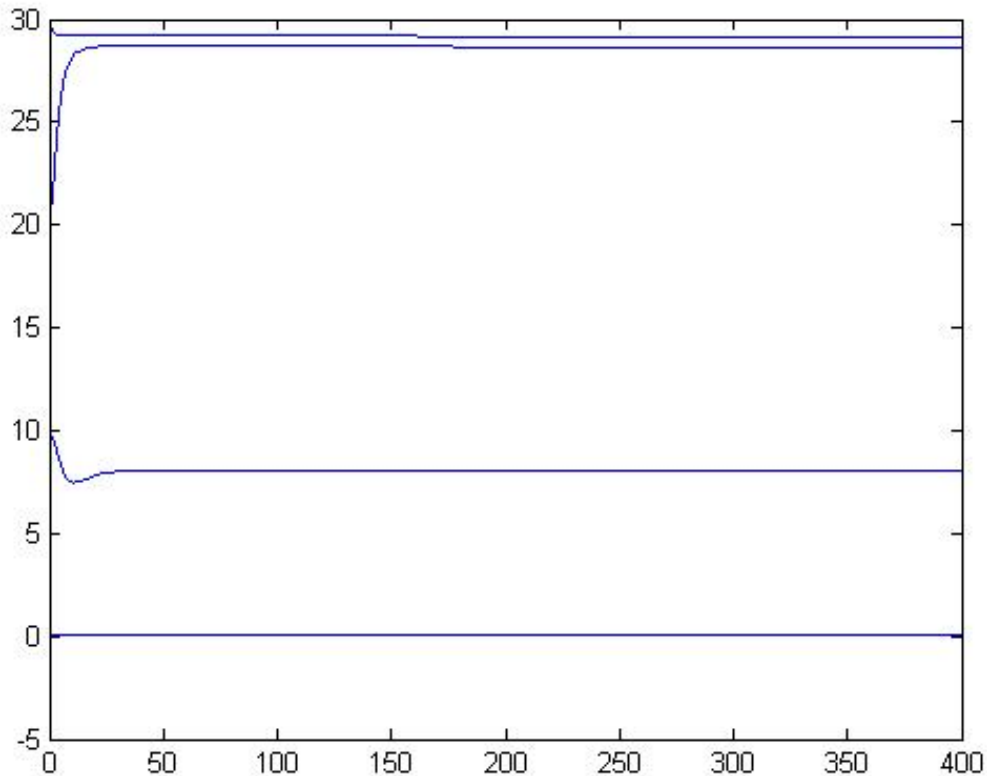


Figure 2.1: plots for $X(0) = 0, 10, 20, 30$ and $T = 400$

2.3.1 $N = 2$

In case $N = 2$, it is easy to find the expected values by hand, as shown before, but to verify that method I used Matlab too. However, the system with 2 trees is too small, so even in the short run the expected value of infected trees is reached in about 5 days (See figure 1.1). So for further study this case is no longer relevant.

2.3.2 $N = 30$

To study the problem Gani and Stalls encountered, I will use the numerical solutions for the expected values of the number of infected trees $N = 30$ trees. First I will make plots for several starting values with $T = 400$ days. I hope to see the same bump as encountered by Gani and Stalls. Next I will make T larger so I am able to see how long it takes to get to the stationary distribution. Having done this all I will move on to the next chapter in which I will try to explain the problem by the quasi-stationary distribution.

I will start by plotting the expected values for starting values: $X(0) = 0, 10, 20, 30$. The parameters used are $\alpha = 0.05, \beta = 0.15, q = 0.9, \lambda = 2, T = 400$ days. The m-file used for these plots is found in Appendix A.

In this figure it can be seen that in a year time the expected values of the number of infected trees seems to be constant and thus one could think to have found the stationary distribution.

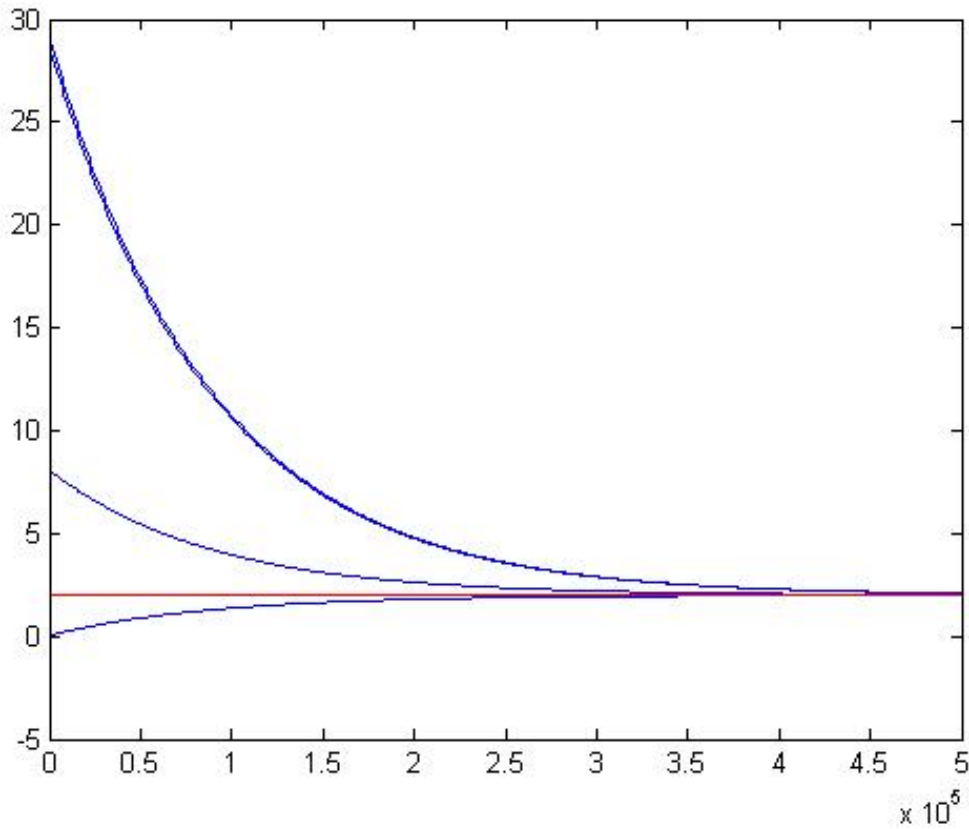


Figure 2.2: plots for $X(0) = 0, 10, 20, 30$ and $T = 50000$

Increasing the time T to 50000 gives a plot in which it is seen that the stationary distribution is not in the values found in the plots of $T = 400$.

I expect that this happens, because of the differences in the powers, which are found in the solutions of the $p_j(t)$. I expect that the first power, with the smallest eigenvalue, will slow the process down. In other words, the smaller eigenvalues will make the e -power converges to 0 in the limit much faster than the e -power in which the second biggest eigenvalue is found.

Chapter 3

Quasi-stationary distribution

In this chapter I am going try explaining the bumps arising in the plots by the quasi-stationary distribution. I will first introduce the idea of this quasi-stationary distribution and after that I am using the eigenvalue method again to see if it explains the bumps in the plots.

3.1 The idea

The idea of the quasi-stationary distribution is basically to leave out state 0. It will still be possible to start with zero infected trees, however it's only possible to leave this state, but it will never be entered again. For the Q -matrix this means that de first column will consist of zeroes only. The Q -matrix will now be denoted by ${}_0Q$, with the zero referring to the fact that state 0 can only be left never to come back there. For the example with 2 trees this matrix is

$${}_0Q = \begin{pmatrix} 0 & \alpha & 0 \\ 0 & -(\lambda q + \alpha + \beta) & \alpha + \beta \\ 0 & \lambda q & -\lambda q \end{pmatrix}. \quad (3.1)$$

Now I can use the eigenvalue method again on this matrix. The only extra thing I have to do now is normalizing the matrix I get when I've determined $e^{0Q t}$. This has to be done, since the $p_j(t)$ that are found this way no longer sum up to 1. However, I am not interested in the P -matrix, but I am interested in the number of infected trees at time t . So rather than normalizing the P -matrix, I will normalize the number of infected trees at time t .

Assuming $X(0) = a$ the expected value of the number of infected trees now becomes

$$\mathbb{E}_a(X(t)) = \frac{\sum_{j=0}^N j \cdot p_{aj}(t)}{\sum_{k=0}^N p_{ak}(t)}. \quad (3.2)$$

Here $p_{aj}(t)$ and $p_{ak}(t)$ are entries of the corresponding P -matrix.

Now I have defined this matrix and the expected value of the number of infected trees at time t , I will use Matlab again to find numerical solutions for the expected value of the number of infected trees. Again I will be using the eigenvalue method.

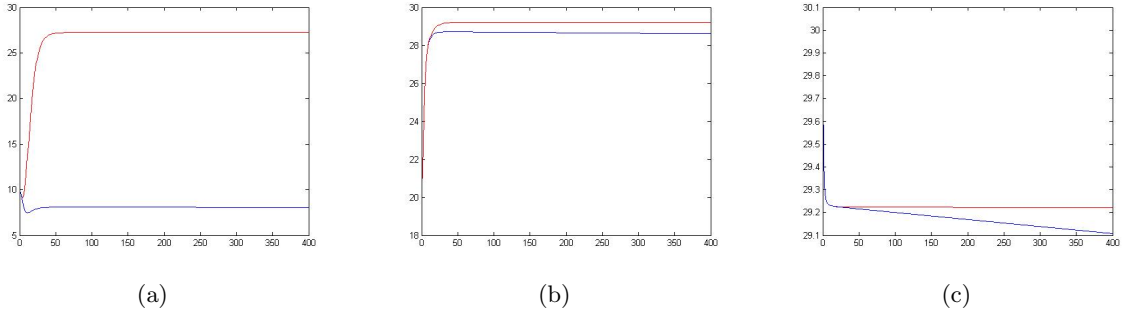


Figure 3.1: Plots for QSD with $X(0) = 10, 20$ and 30

3.2 Numerical solutions for the QSD

For finding numerical solutions using the matrix ${}_0Q$ (3.1), I am using almost the same m-file as I have used for finding the numerical solutions for the complete system. However, slight change I needed to make was the normalizing factor. The m-file used for plots is found in Appendix B.

For plotting the QSD I used 30 trees again and I have chosen the same parameters as before and made plots for different starting values of infected trees. These parameters were: $\alpha = 0.05, \beta = 0.15, q = 0.9, \lambda = 2$. Now I have chosen T to be 400 and next to the plot of the QSD, I have plotted the numerical solutions of the complete system too. That way I want to check if the QSD describes the complete system in the short run.

Above there are three figures with respectively $X(0) = 10, 20$ and 30 . The red plots are the QSD-plots. From these I can now conclude that the QSD is not a good explanation for the bumps that appear in the complete model. However, I have been trying to other ideas to try and explain the bumps.

The first idea I have tried is making a new restriction to the process. This time I didn't remove the possibility of entering the state of 0 infected trees, but instead of that I removed possibility of leaving the state 0. This means that in the long run 0 is an absorbing state. This means that the ${}_0Q$ -matrix changes into the following (again in the example of 2 trees)

$${}_0Q' = \begin{pmatrix} 0 & 0 & 0 \\ \lambda q & -(\lambda q + \alpha + \beta) & \alpha + \beta \\ 0 & \lambda q & -\lambda q \end{pmatrix}. \quad (3.3)$$

The downside to this idea is that, in the long run it will stay in the state 0 and thus it won't give a good description of the complete system. However, if this idea gives a good description in the short run it might need to be altered a little.

The plots I have made in Matlab for this model compare pretty well to the plots for the original model. The m-file used for the plots is found in Appendix C. The plots for the short run for both models come pretty much together (plots are found on the next page). Only for lower starting values there can be seen a little difference.

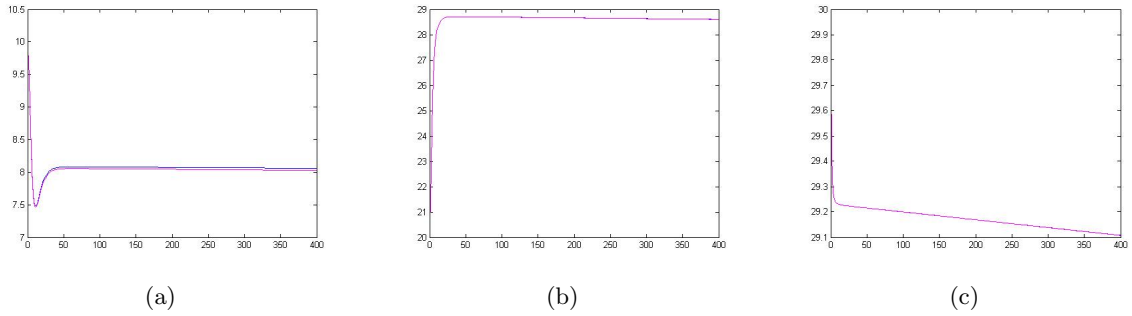


Figure 3.2: Plots for idea 1 with $X(0) = 10, 20$ and 30

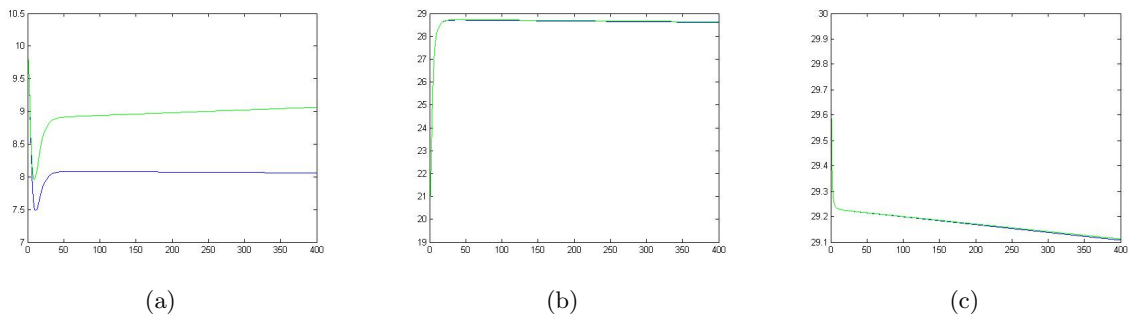


Figure 3.3: Plots for idea 2 with $X(0) = 10, 20$ and 30

Another idea I have been trying, is more based on the QSD. The possibility of entering case 0 is left out again and I tried plotting with and without normalization, but in this case it doesn't seem to matter if you normalize or not. Next to this change I made another little change in the ${}_0Q$ -matrix, making it the following

$${}_0Q'' = \begin{pmatrix} 0 & \alpha & 0 \\ 0 & -(\alpha + \beta) & \alpha + \beta \\ 0 & \lambda q & -\lambda q \end{pmatrix}. \quad (3.4)$$

The second change I have made is leaving out the λq in the entry ${}_0Q(1,1)$. The plots I get now are more accurate than with the λq . Again for this idea the stationary distribution is not good enough to give an explanation for the original idea, however in the short run it is. The short run plots above seem to be a good approximation of the complete model, however with lower starting values this model is a lot less accurate than that of the first alternative idea. The m-file used for these plots is found in Appendix D. Something worth of noting, is the fact that this last model has just a little adjustment to the original method, however it is strange to see that in the long run the expected value of infected trees is at about 21 for the chosen parameters. This is a huge difference with the original model, where the expected value of infected trees is about 2.

So concluding this chapter it seems that the quasi stationary distribution does not explain

the bumps that appear in the long run plots of the original model. However it might be that slight changes in the idea of the quasi stationary distribution might explain these bumps.

Chapter 4

Conclusion

After studying the model introduced by Gani and Stalls, I have tried to explain a bump seen in plots of the expected value of the number of infected trees by the theorem of the quasi stationary distribution. After this study I can conclude that this bump in the plots is not explained by the quasi-stationary distribution. Even worse the QSD doesn't come near an explanation. This is because the quasi-stationary distribution in the short run doesn't look like the behavior in the short of the original process

It is good to see however that there are a few points which might be studied more. One of these points is the stochastic process in which $Q(1,0) = 0$ and $Q(1,1) = -(\alpha + \beta)$. The stochastic process obtained this way does describe the original model well enough in the short run, however in the long run it is remarkable that this process has a very different stationary distribution. In example in the case of the parameters chosen by Gani and Stalls this is 21. So one could ask himself why this is so dependent of the starting value of infected trees, since the original model is only slightly different. Another remarkable thing to this process is the fact that normalizing this process doesn't seem to matter.

Another point worth of studying is another stochastic process. This process looks a lot like the process above, only here $Q(1,0)$ is not changed and $Q(0,1) = 0$, meaning that once state 0 is entered it will never be left. On the short run this describes the original model very well. Also this process works even better without the normalizing factor. Unfortunately state 0 is absorbing, so in the long run this process isn't very good as well.

The problem with these options is that in the long run, the stationary distribution is not the same as it seems in the short run. So in the end the problem remains the same as in the beginning. Maybe further study and adjustments of the processes as before might give better processes which give good way to find the distribution in the short run.

For real applications there are some suggestions however. One would like to control the process in the short run. This might be done by minimizing λ under the restriction that there is a probability of less then x percent that there are infected trees. Or minimizing λ under the restriction that the average number of infected trees is at most a value of y infected trees.

Downside to this is method is that λ may not depend on the states, since in principle one does not know the exact amount of infected trees at a time t . Next to that in the short run the system depends heavily on the initial number of infected trees, so it is necessary to make an estimate of the number of infected trees at time 0. Based on the last stochastic process it might also be good to try and minimize λ under the restriction that the total expected

number of infected trees at time t until absorption in 0 stays under a certain value. Again this is dependent on the number of infected trees at time 0.

Another interesting point that I have noticed while studying the model is the following: the more trees there are in the model, the eigenvalues seem to converge to certain values. In case of the parameters chosen here, they even have the same distance to each other. I think this is very unexpected and might even be good to be studied since it doesn't depend on the number of trees and for models like this one might use this fact to make an estimate of the eigenvalues.

Appendix A

```
function f = numeriek(lambda,q,alpha,beta,n,a,g,T);

Q=zeros(n);

Q(1,1)=-alpha;
Q(1,2)=alpha;
Q(n,n-1)=lambda*q;
Q(n,n)=-lambda*q;

for i=2:(n-1)
Q(i,i-1)=lambda*q;
Q(i,i+1)=alpha+beta*(i-1);
Q(i,i)=-(lambda*q+alpha+(i-1)*beta);
end

[U V]=eig(Q, 'nobalance');
v=sort(diag(V));

w=[];
for i=1:(n-1)
w=[w,v(i)];
vpa(w,10);
end

x=[];
for t=0:g:T
A=expm(t*V);
B=U*A*U(-1);
C=0;
for j=1:n
C=C+(j-1)*B(a+1,j);
end
x=[x,C];
end
t=[0:g:T];
plot(t,x)
```

Appendix B

```
function f = numeriekQSD(lambda,q,alpha,beta,n,a,g,T);

Q=zeros(n);

Q(1,2)=alpha;
Q(n,n-1)=lambda*q;
Q(n,n)=-lambda*q;
Q(2,3)=alpha+beta;
Q(2,2)=-(lambda*q+alpha+beta);

for i=3:(n-1)
    Q(i,i-1)=lambda*q;
    Q(i,i+1)=alpha+beta*(i-1);
    Q(i,i)=-(lambda*q+alpha+(i-1)*beta);
end

[U V]=eig(Q, 'nobalance');
v=sort(diag(V));

w=[];
for i=1:(n)
    w=[w,v(i)];
end
vpa(w,10);

x=[];

for t=0:g:T
    A=expm(t*V);
    B=U*A*U(-1);
    C=0;
    D=0;

    for i=1:n
        D=D+B(a+1,i);
    end
end
```

```
for j=1:n
C=C+(((j-1)*B(a+1,j))/D);
end
```

```
x=[x,C];
end
```

```
t=[0:g:T];
plot(t,x,'r')
```

Appendix C

```
function f = numeriekQ(0,1)(lambda,q,alpha,beta,n,a,g,T);

Q=zeros(n);

Q(2,1)=lambda*q;
Q(n,n-1)=lambda*q;
Q(n,n)=-lambda*q;
Q(2,3)=alpha+beta;
Q(2,2)=-(lambda*q+alpha+beta);

for i=3:(n-1)
    Q(i,i-1)=lambda*q;
    Q(i,i+1)=alpha+beta*(i-1);
    Q(i,i)=-(lambda*q+alpha+(i-1)*beta);
end

[U V]=eig(Q, 'nobalance');
v=sort(diag(V));

w=[];
for i=1:(n)
    w=[w,v(i)];
end
vpa(w,10);

x=[];
for t=0:g:T
    A=expm(t*V);
    B=U*A*U(-1);
    C=0;
    D=0;

    for i=1:n
        D=D+B(a+1,i);
    end
end
```

```
for j=1:n
C=C+(((j-1)*B(a+1,j)));
end
```

```
x=[x,C];
end
```

```
t=[0:g:T];
plot(t,x,'m')
```

Appendix D

```
function f = numeriekQ(1,0)(lambda,q,alpha,beta,n,a,g,T);
```

```
Q=zeros(n);
```

```
Q(1,2)=alpha;
```

```
Q(n,n-1)=lambda*q;
```

```
Q(n,n)=-lambda*q;
```

```
Q(2,3)=alpha+beta;
```

```
Q(2,2)=-(alpha+beta);
```

```
for i=3:(n-1)
```

```
Q(i,i-1)=lambda*q;
```

```
Q(i,i+1)=alpha+beta*(i-1);
```

```
Q(i,i)=-(lambda*q+alpha+(i-1)*beta);
```

```
end
```

```
[U V]=eig(Q, 'nobalance');
```

```
v=sort(diag(V));
```

```
w=[];
```

```
for i=1:(n)
```

```
w=[w,v(i)];
```

```
vpa(w,10);
```

```
end
```

```
x=[];
```

```
for t=0:g:T
```

```
A=expm(t*V);
```

```
B=U*A*U(-1);
```

```
C=0;
```

```
D=0;
```

```
for i=1:n
```

```
D=D+B(a+1,i);
```

```
end
```

```
for j=1:n
C=C+(((j-1)*B(a+1,j)));
end
```

```
x=[x,C];
end
```

```
t=[0:g:T];
plot(t,x,'g')
```


Bibliography

- [1] J. GANI & L. STALLS (2005), A continuous time Markov chain model for a plantation-nursery system. *Environments* **16**, 849–861.
- [2] H.A. PRIESTLEY (1997), *Introduction To Complex Analysis*. Cambridge University Press, Cambridge.