Computational Aspects of a Stochastic Metapopulation Model

Staffan Ulfberg

17th of December, 1993

Abstract

A stochastic, discrete time, single species metapopulation model with a finite number of patches, each of which can be either occupied or empty, can be described by a Markov chain with 2^n states, where n denotes the number of patches in the system. The analysis of such a Markov chain is computationally a very hard, or even impossible task if n is not very small. This final degree project provides a possible means for making approximations of the asymptotic incidence of occupancy and distribution of the number of occupied patches for systems where n is large enough to describe systems commonly found in nature.

Contents

Preface

This paper has been written as my Master's thesis in mathematics at the department of applied mathematics at the University of Luleå in Sweden. I hope that the results herein will be useful, or at least interesting, for people working in the field of mathematical biology and population dynamics.

I was introduced to the subject in spring 1993 by my instructor and examiner Professor Mats Gyllenberg, who has been working with models in biology for a long time. I would like to thank him for guiding me towards interesting fields of the subject.

I would like to thank Dmitrii Silvestrov at the department of applied mathematics who has generously shared his knowledge of Markov chain theory. I would also like to thank Professor Ilkka Hanski at the University of Helsinki for giving me access to his yet unpublished data for the butterfly *Melitaea Cinxia*.

Luleå Staffan Ulfberg
December, 1993

Chapter 1

Introduction

Models of local population dynamics typically consider how individuals in a homogeneous population interact with each other, and predict the behavior of a population's size. A collection of local populations that are connected by migration is called a metapopulation. The individuals in a metapopulation do not generally form one homogeneous population, since individuals may interact more with individuals in the same local population than with those in other local populations.

This paper deals with the stochastic model of metapopulations given in [5]. This model deals with metapopulations that have a finite number of local populations that each occupy one patch. The number of individuals in each local population is ignored; the model only considers if a patch is empty or occupied. Local populations may go extinct, and an empty patch may become colonized by colonizers from other occupied patches. Both these events are assumed to be stochastic and to occur in discrete time.

Local population dynamics are supposed to act on a faster time scale than the metapopulation dynamics, which leads to the assumption that the number of individuals on a colonized patch can be determined from the patch characteristics. The probability that a local population goes extinct often depends on the patch size or any other measure of how many individuals of the species in question the patch can carry. The probability of a patch becoming colonized at a specific time typically depends on how many surrounding patches are occupied, how far away these patches are and how many individuals they carry. It is not the purpose of this paper to provide methods for calculating these probabilities.

In nature, habitat areas can be islands in an archipelago, or flowers on which some species of insect live. Patchy environments also occur where large habitat areas have been fragmented. The model considered can handle metapopulations where local extinction and colonization probabilities are different for all patches. This is necessary to successfully apply metapopulation models to real metapopulations in which most often local population sizes are all different, and patch locations do not form any symmetric pattern.

In conservation biology, which is an area where metapopulation models are very useful, questions that arise may be "What will happen to a specific metapopulation if some patches are removed or disconnected from each other?", or "Will an existing metapopulation persist?".

Some important properties of a metapopulation are the stationary probabilities for the patches to be occupied (the incidence of occupancy), and the stationary probability distribution of the number of occupied patches. Similar properties are discussed in for example [4] and [1]. Expressions for these quantities are derived in [5], but these expressions are mainly of theoretical interest, since evaluation becomes practically impossible for metapopulations with a reasonable number of patches. (However, some interesting properties can be calculated using very small metapopulations.) It is the main purpose of this paper to provide a means of estimating the incidence and the stationary distribution of the number of occupied patches for metapopulations consisting of a quite large number of patches.

A review of some results in [5] can be found in Chapter 2, which also introduces some notation used throughout the paper. Chapter 2 also contains spectral analysis of Markov chains, that could be used to calculate how often local extinction and recolonization events occur. Chapter 3 presents two different methods for estimating the incidence of occupancy, of which one can also estimate the probability distribution of the number of occupied patches. Chapter 4 describes how to simulate the modelled metapopulation. Finally, Chapter 5 shows how the results can be used in practice. The metapopulation of *Melitaea cinxia* described in [7] is considered in this chapter.

Chapter 2

Mathematical Representation

To mathematically represent a metapopulation consisting of n patches we use a homogeneous Markov chain with state space

$$X = \left\{ \left[\begin{array}{cccc} x_1 & x_2 & \cdots & x_n \end{array} \right] \mid x_i \in \{0, 1\} \right\}.$$

The state of the metapopulation at time t is represented by the random vector

$$\overline{\eta}(t) = \left[\begin{array}{cccc} \eta_1(t) & \eta_2(t) & \cdots & \eta_n(t) \end{array} \right] \in X,$$

where $\eta_i(t) = 0$ indicates that patch i is empty and $\eta_i(t) = 1$ that patch i is occupied at time t. This makes a total of 2^n states that the metapopulation can assume.

An $(n \times n)$ *interaction matrix* $Q = (q_{ij})$ describes the local extinction probabilities for all local populations and colonization rates for all pairs of patches in the metapopulation. q_{ij} is the probability that colonizers originating from patch i will *not* colonize patch j in one time step given that patch i is inhabited. The local extinction probability of the local population at patch i is q_{ii}. The elements in the matrix Q can be determined by any arbitrary biological model, that may deal with patch sizes, locations and vegetation etc.

In this paper the following conditions are assumed to hold for the interaction matrix Q.

1. $q_{11} = 0$ and $q_{ii} > 0,\ i > 1$.
2. $q_{ii} < 1,\ i \in \{1, 2, \ldots, n\}$.
3. For each $(i, j) \in \{1, 2, \ldots, n\} \times \{1, 2, \ldots, n\}$, there exists an $m \in \mathbf{N}$ and a chain of indices $\{i_k\}_{k=0}^{m}$, $i = i_0$, $j = i_m$ such that $q_{i_k, i_{k+1}} < 1$, $k \in \{0, 1, \ldots, m-1\}$.

Condition 1 means that all patches except one have a positive extinction probability; one patch plays the role of a *mainland* and its local population cannot go extinct. If all extinction probabilities are greater than zero, the metapopulation will eventually become extinct, and the stationary probabilities $\pi(\overline{x}) = 0$, $\overline{x} \in X \setminus \overline{0}$ and $\pi(\overline{0}) = 1$. According to [5] however, a *quasi-stationary* behavior of the metapopulation can be observed before it goes extinct. To calculate the quasi-stationary probabilities it is possible to analyse a patch system that contains a mainland. The system that has no mainland is considered to be a perturbation of the mainland system and has quasi-stationary probabilities that are equal to the stationary probabilities of the analysed system.

Condition 2 means that no patch has a local extinction probability of 1, and finally, Condition 3 means that migrators from every patch can directly or indirectly, through other patches colonize every other patch. These conditions are equivalent to conditions A and B_1 in [5].

We suppose that the mainland is initially inhabited which implies that $\eta_1(t) = 1$ for all $t \geq 0$. Thus it is possible to consider only those states of X where $x_1 = 1$. This subset D of X has 2^{n-1} elements.

Given any state $\overline{x} \in D$, it is possible to reach all other states $\overline{y}$ in D. This is easily seen by considering the possibility that emigrants from the mainland first colonize all patches. After this all patches i for which $y_i = 0$ go extinct, and state $\overline{y}$ is reached. Thus the homogeneous Markov chain is also irreducible and can be described by an irreducible probability matrix P. Now, since this matrix has all row sums equal to 1, and the *Perron-Frobenius* eigenvalue lies between the smallest and largest row sum [9], the Perron-Frobenius eigenvalue is 1. The Perron-Frobenius theorem for irreducible matrices now implies that there is a unique stationary probability distribution $\overline{\pi}$, given by $\overline{\pi}P = \overline{\pi}$ [9]. It also holds that $\lim_{k\to\infty} P^k = \overline{1}\overline{\pi}$.

2.1 Stationary Probability Distribution

In this section we calculate the stationary probabilities for all states in D. From this we also calculate the probability distribution of the number of occupied patches, and the incidence of occupancy for each patch.

The probability that patch i is empty at time $t+1$ given that the metapopulation was in state $\overline{x}$ at time t is

$$q_i(\overline{x}) = \prod_{j=1}^{n} q_{ji}^{x_j}. \tag{2.1}$$

In this formula and throughout the paper the convention that $0^0 = 1$ is used.

The Markov chain transition probabilities $P(\overline{x}, \overline{y})$ that give the probabilities that the chain will switch from any state $\overline{x}$ to state $\overline{y}$ in one time step are

$$P(\overline{x}, \overline{y}) = \prod_{i=1}^{n} q_i(\overline{x})^{1-y_i}(1 - q_i(\overline{x}))^{y_i}, \quad \overline{x}, \overline{y} \in D. \tag{2.2}$$

The k step transition probabilities $P^k(\overline{x}, \overline{y})$ are defined by

$$P^k(\overline{x}, \overline{y}) = \begin{cases} 1, & k = 0,\ \overline{x} = \overline{y} \\ 0, & k = 0,\ \overline{x} \neq \overline{y} \\ \sum_{\overline{z} \in D} P(\overline{x}, \overline{z}) \cdot P^{k-1}(\overline{z}, \overline{y}), & k \geq 1 \end{cases}. \tag{2.3}$$

$P^k(\overline{x}, \overline{y})$ is the probability that state $\overline{x}$ leads to state $\overline{y}$ in k time steps.

The function $P(\overline{x}, \overline{y})$ defines a $(2^{n-1} \times 2^{n-1})$ Markov chain transition probability matrix P in a natural way by ordering the states with some function $\phi : \{1, 2, \ldots, 2^{n-1}\} \to D$. It follows that $P^k(\phi(i), \phi(j)) = p_{ij}^{(k)}$, where $P^k = \left\{p_{ij}^{(k)}\right\}$ denotes matrix powers.

We can now obtain the stationary probability distribution $\pi(\overline{y})$, $\overline{y} \in D$ by solving the following system of linear equations [5]:

$$\pi(\overline{y}) = \sum_{\overline{x} \in D} \pi(\overline{x}) P(\overline{x}, \overline{y}),\ \overline{y} \in D,\ \sum_{\overline{x} \in D} \pi(\overline{x}) = 1.$$

Alternatively, using matrix notation, we can obtain the vector $\overline{\pi}$, $\pi_i = \pi(\phi(i))$ by solving the eigenvector problem

$$\overline{\pi} = \overline{\pi} P, \ \sum_{i=1}^{2^{n-1}} \pi_i = 1. \tag{2.4}$$

Now let the function $\rho(\overline{x}) = \sum_{i=1}^{n} x_i$ denote the number of occupied patches when in state $\overline{x}$. The distribution of the number of occupied patches $\{p(k)\}_{k=1}^{n}$ is given by

$$p(k) = \sum_{\rho(\overline{x}) = k} \pi(\overline{x}), \tag{2.5}$$

or alternatively using matrix notation:

$$p(k) = \sum_{\rho(\phi(i)) = k} \pi_i. \tag{2.6}$$

The incidence of occupancy J_i of the ith patch is

$$J_i = \sum_{\overline{x} \in D | x_i = 1} \pi(\overline{x}), \tag{2.7}$$

or in matrix notation

$$J_i = \sum_{\phi(j)_i = 1} \pi_j. \tag{2.8}$$

Appendix A contains an implementation in Matlab for performing these computations.

2.2 Spectral Analysis

Let the function $\theta(\overline{\eta}(t))$ denote any arbitrary mapping from D to $\mathbf{R}$. A formula to calculate the spectral power density of the output $\theta(\overline{\eta}(t))$ from the Moore machine described by the Markov chain $\overline{\eta}(t)$ is now derived.

We have

$$P(\overline{\eta}(t+k) = \phi(j) \mid \overline{\eta}(t) = \phi(i)) = P_{ij}^{k} \tag{2.9}$$

and also that

$$P(\eta(t) = \phi(i)) = \pi_i. \tag{2.10}$$

The *auto-correlation function* is defined by

$$r(k) = E\left[\theta(t) \cdot \theta(t+k)\right]. \tag{2.11}$$

The auto-correlation function $r(k)$ describes how the output signal at time t is connected to the output at time $t+k$. With the help of (2.9), (2.10) and the definition of expectation we have

$$r(k) = \sum_{i=1}^{2^{n-1}} \sum_{j=1}^{2^{n-1}} \pi_i \theta(\phi(i))\theta(\phi(j)) P_{ij}^{|k|} = \overline{\theta}\Pi P^{|k|} \overline{\theta}^{T}, \tag{2.12}$$

where $\Pi = diag(\overline{\pi})$ and $\overline{\theta} = \left[\begin{array}{cccc} \theta(\phi(1)) & \theta(\phi(2)) & \cdots & \theta(\phi(2^{n-1})) \end{array}\right]$.

The spectral density function $R(f)$, that describes the output signal's power distribution in frequency, is defined by

$$R(f) = \sum_{k=-\infty}^{\infty} r(k) e^{-j2\pi f k}, \tag{2.13}$$

and becomes, with $r(k)$ as in (2.12):

$$R(f) = \sum_{k=-\infty}^{\infty} \overline{\theta}\Pi P^{|k|} \overline{\theta}^{T} e^{-j2\pi f k}. \tag{2.14}$$

To remove the infinite series expression and transfer (2.14) to a closed form expression we need the following properties.

Property 1 $P \cdot \overline{1}\overline{\pi} = \overline{1}\overline{\pi}$, *and* $P^k \cdot \overline{1}\overline{\pi} = \overline{1}\overline{\pi}$, $k \geq 0$.

Property 2 *For all* $k \geq 1$ *we have*

$$P^k - \overline{1}\overline{\pi} = (P - \overline{1}\overline{\pi})^k.$$

Proof. The equality is proved by induction over k. For $k = 1$ the property is trivially true. Now suppose that $P^k - \overline{1}\overline{\pi} = (P - \overline{1}\overline{\pi})^k$ is true for $k = n$. For $k = n + 1$ we have

$$(P - \overline{1}\overline{\pi})^{n+1} = (P - \overline{1}\overline{\pi})^n \cdot (P - \overline{1}\overline{\pi}) = (P^n - \overline{1}\overline{\pi}) \cdot (P - \overline{1}\overline{\pi}) = P^{n+1} - \overline{1}\overline{\pi},$$

where Property 1 is used.

Property 3 *If μ_i are the eigenvalues of P, then for any polynomial function g, $g(\mu_i)$ are the eigenvalues of $g(P)$. For a proof of this see [3], p. 84.*

It is now possible to rewrite the spectral density function as follows.

$$\begin{aligned} R(f) = \sum_{k=-\infty}^{\infty} \overline{\theta}\Pi(P^{|k|} - \overline{1}\overline{\pi})\overline{\theta}^T e^{-j2\pi fk} + \overline{\theta}\overline{\pi}^T\overline{\pi}\overline{\theta}^T \sum_{k=-\infty}^{\infty} e^{-j2\pi fk} = \\ = 2\,\mathrm{Re}\sum_{k=1}^{\infty} \overline{\theta}\Pi(P^k - \overline{1}\overline{\pi})\overline{\theta}^T e^{-j2\pi fk} + \\ +(\overline{\theta}\Pi\overline{\theta}^T - \overline{\theta}\overline{\pi}^T\overline{\pi}\overline{\theta}^T) + E\left[\theta(\overline{\eta}(t))\right]^2 \sum_{k=-\infty}^{\infty} e^{-j2\pi fk} = \\ = 2\overline{\theta}\Pi\,\mathrm{Re}\left[\sum_{k=1}^{\infty}\left((P - \overline{1}\overline{\pi})e^{-j2\pi f}\right)^k\right]\overline{\theta}^T + \\ +E\left[\theta(\overline{\eta}(t))^2\right] - E\left[\theta(\overline{\eta}(t))\right]^2 + E\left[\theta(\overline{\eta}(t))\right]^2 \sum_{k=-\infty}^{\infty} \delta(f-k), \end{aligned} \tag{2.15}$$

where the last step utilizes Property 2. The eigenvalues $\{\mu_i\}$ of the probability matrix P satisfy the condition $|\mu_i| \leq 1$, $i \in \{1, 2, \ldots, 2^{n-1}\}$ (follows from the Perron-Frobenius theorem, see [9]). Now define $g_k(x) = (x - x^k)e^{-j2\pi f}$ so that $(P - \overline{1}\overline{\pi})e^{-j2\pi f} = \lim_{k\to\infty} g_k(P)$. The eigenvalues $\{\lambda_i\}$ of $(P - \overline{1}\overline{\pi})e^{-j2\pi f}$ are, according to Property 3, given by $\lambda_i = \lim_{k\to\infty} g_k(\mu_i) = \lim_{k\to\infty}(\mu_i - \mu_i^k)e^{-j2\pi f}$, and $|\lambda_i| = \left|\lim_{k\to\infty}(\mu_i - \mu_i^k)\right| < 1, \quad i \in \{1, 2, \ldots, 2^{n-1}\}$, so the matrix power series in (2.15) converges (see [3], p. 112) and

$$\begin{aligned} R(f) = 2\overline{\theta}\Pi\,\mathrm{Re}\left[(P - \overline{1}\overline{\pi})e^{-i2\pi f}\left(I_{2^{n-1}} - (P - \overline{1}\overline{\pi})e^{-i2\pi f}\right)^{-1}\right]\overline{\theta}^T + \\ + \mathrm{Var}\left[\theta(\overline{\eta}(t))\right] + E\left[\theta(\overline{\eta}(t))\right]^2 \sum_{k=-\infty}^{\infty} \delta(f-k) = \\ = 2\overline{\theta}\Pi\,\mathrm{Re}\left[(P - \overline{1}\overline{\pi})\left(e^{i2\pi f} I_{2^{n-1}} - (P - \overline{1}\overline{\pi})\right)^{-1}\right]\overline{\theta}^T + \end{aligned}$$

$$+\,\mathrm{Var}\left[\theta(\overline{\eta}(t))\right] + E\left[\theta(\overline{\eta}(t))\right]^2 \sum_{k=-\infty}^{\infty} \delta(f-k). \tag{2.16}$$

Here δ denotes Dirac's delta function defined by $\int_{-\infty}^{\infty} \delta(x)\phi(x)dx = \phi(0)$ for any test function ϕ.

Chapter 3

Some Possible Reductions

The eigenvector problem (2.4) is computationally very tough even for quite moderate values of n. To use the stochastic metapopulation model considered for patch systems with reasonably many patches, we need to find alternative methods to calculate the incidence of occupancy J_i and the stationary distribution of the number of occupied patches $p(k)$. In this section two methods for approximating the incidence of occupancy are presented.

The first, that also approximates the distribution of the number of occupied patches, can be used if some patches are equal in a specific sense; in this case they can be lumped together. A Markov chain that keeps track of how many occupied patches there are in each lump, the *lumped Markov chain,* has considerably less states than the original one (where patches have not been lumped together). If the original Markov chain does not have any patches that can be lumped exactly, we could still obtain an approximation of J_i and $p(k)$ by slightly changing the colonization and extinction probabilities. The error introduced by such a change in the matrix Q is analysed at the end of the chapter.

The other method gives, for each patch i, an interval that contains the incidence of occupancy J_i. This is probably the better of the two methods to use in many cases, but unfortunately it doesn't give any information about the stationary distribution of the number of occupied patches. Since this method gives upper and lower limits for the incidence, it could also be used to reduce the number of states in the original Markov chain as follows. If some patches are inhabited with a very low probability, we could consider the possibility that their influence on the system is negligible, and analyse the system without them. Another possibility is that some patches are inhabited with a very high probability; in this case it is sometimes reasonable to assume that these patches are other mainlands. For each patch that is removed, the number of states in the system is reduced by a factor of two.

3.1 Lumping of Patches

We partition the patches $\{1, 2, \dots, n\}$ into g disjoint sets $S_1, S_2, \dots, S_g$ where $\bigcup_{i=1}^{g} S_i = \{1, 2, \dots, n\}$. Patches that should remain unlumped are put into separate sets with only one member. The lumping method described in this section requires the following conditions on the matrix Q, which also define the *single patch extinction probability,* t_k, and the *inter-lump colonization probability,* $1-r_k$, for each lump $k \in \{1, 2, \dots, g\}$.

1. $q_{ii} = q_{jj} = t_k, \quad i, j \in S_k.$
2. $q_{ij} = r_k, \quad i \neq j$ and $i, j \in S_k$.
3. $q_{li} = q_{lj}, \quad l \notin S_k$ and $i, j \in S_k$.
4. $q_{il} = q_{jl}, \quad l \notin S_k$ and $i, j \in S_k$.

Condition 1 ensures that all local populations within a lump of patches have the same local extinction probability t_k. Condition 2 means that individuals from all patches within a lump have the same probability, $1 - r_k$, to colonize other patches within the lump. Condition 3 means that the probability for individuals from each of the local populations within one lump to colonize all other patches must be equal. Likewise, Condition 4 means that the probability for each of the patches within a lump to be colonized by any other local population must be equal.

The metapopulation can now be represented by a Markov chain with state space

$$Y = \left\{ \begin{bmatrix} y_1 & y_2 & \cdots & y_g \end{bmatrix} \mid y_i \in \{0, 1, \dots, |S_i|\} \right\},$$

where $|S_i|$ denotes the number of elements in S_i. The state $\overline{\xi}(t)$ at time t records the number of occupied patches in each lump at that time. If each lump contains exactly one patch, the lumping conditions above are obviously satisfied and $Y = X$. If on the other hand some lumps contain more than one patch, we do not know which of the patches in a lump are inhabited at a specific moment. However, all other information about the metapopulation is preserved. It is straightforward to calculate the stationary distribution of the number of occupied patches given the stationary probabilities of the lumped Markov chain. The incidence of occupancy, however, requires some new formulae.

We first derive the state transition probabilities $P(\overline{x}, \overline{y})$ for the lumped Markov chain as follows. Let $q_i(\overline{x}, n)$ denote the probability that lump i will have n occupied patches at time $t + 1$, given that the Markov chain was in state $\overline{x}$ at time t. The probability $\sigma_k(\overline{x})$ that an inhabited patch $j \in S_k$ will not become extinct in one time step when in state $\overline{x}$ is

$$\sigma_k(\overline{x}) = 1 - t_k r_k^{x_k - 1} \prod_{i \notin S_k} q_{im}, \ m \in S_k, \tag{3.1}$$

and the probability $\nu_k(\overline{x})$ that a specific empty patch in S_k will be colonized during the next time unit is

$$\nu_k(\overline{x}) = 1 - r_k^{x_k} \prod_{i \notin S_k} q_{im}, \ m \in S_k. \tag{3.2}$$

Given that the metapopulation was in state $\overline{x}$ at time t, the probability that at time $t+1$, lump S_k will have a patches that have not become extinct since time t, and b patches that have been colonized since time t, can be determined with the help of (3.1) and (3.2). The probability is

$$f_k(\overline{x}, a, b) = \binom{x_k}{a} \sigma_k(\overline{x})^a (1 - \sigma_k(\overline{x}))^{x_k - a} \cdot$$

$$\cdot \binom{|S_k| - x_k}{b} \nu_k(\overline{x})^b (1 - \nu_k(\overline{x}))^{|S_k| - x_k - b} =$$

$$= \binom{x_k}{a} \left(1 - t_k r_k^{x_k - 1} \prod_{i \notin S_k} q_{i,m}\right)^a \left(t_k r_k^{x_k - 1} \prod_{i \notin S_k} q_{i,m}\right)^{x_k - a} \cdot$$

$$\cdot \binom{|S_k| - x_k}{b} \left(1 - r_k^{x_k} \prod_{i \notin S_k} q_{i,m}\right)^b \left(r_k^{x_k} \prod_{i \notin S_k} q_{i,m}\right)^{|S_k| - x_k - b},$$

where $m \in S_k$, $a, b \geq 0$, $a \leq x_k$ and $a + b \leq |S_k|$. So we have that

$$q_i(\overline{x}, n) = \sum_{j = \max\{0, n + x_i - |S_k|\}}^{\min\{x_i, n\}} f_i(\overline{x}, j, n - j),$$

from which the Markov chain transition probabilities $P(\overline{x}, \overline{y})$ are

$$P(\overline{x}, \overline{y}) = \prod_{i=1}^{g} q_i(\overline{x}, y_i). \tag{3.3}$$

To derive expressions for the distribution of the number of occupied patches, $p(k)$, and the incidence of occupancy, we calculate the stationary probabilities $\pi(\overline{x})$ by solving the linear equation system

$$\pi(\overline{y}) = \sum_{\overline{x} \in Y} \pi(\overline{x}) P(\overline{x}, \overline{y}), \ \overline{y} \in Y, \ \sum_{\overline{x} \in Y} \pi(\overline{x}) = 1.$$

The distribution of the number of occupied patches can now be evaluated analogously to equation (2.5) as

$$p(k) = \sum_{\sum_{j=1}^{g} x_j = k} \pi(\overline{x}).$$

The incidence of occupancy for the patches can be derived by observing that the probability that lump i has k occupied patches is

$$p_i(k) = \sum_{x_i = k} \pi(\overline{x}).$$

The incidence for a patch in lump i is thus

$$J_i = \sum_{k=1}^{|S_i|} \frac{k}{|S_i|} p_i(k).$$

Appendix B.1 contains Matlab `.m`-files for performing the computations described in this section. In some applications, we might have a metapopulation model that can be lumped as above in an exact manner. For example, we might have a model that considers five different patch sizes, where colonization probabilities depend only on the area of the patches. Even with restrictions on the matrix Q as above, one may construct metapopulation models that correspond to reality much better than models that assume an infinite number of patches.

It is also possible that only approximate geographical data or very uncertain local population dynamics models are available. Then some estimates of Q are not better than others and Q can be chosen quite freely.

3.2 Upper and Lower Bounds for the Incidence

When we consider metapopulations consisting of a quite large number of local populations, some local populations may have very small influence on some of the other local populations. When calculating the incidence of occupancy for a patch in such a metapopulation, we would like to exclude patches that do not effect the incidence of occupancy that we are trying to calculate. It would be hazardous to exclude patches from the calculation without knowing how the result is affected. This section presents an approach to reducing the complexity of the system by calculating the incidence of occupancy for the patches under certain conditions that make the number of states smaller. The results obtained are upper and lower bounds for the incidence of occupancy for all patches in the metapopulation. At best these bounds give a narrow interval which is known to contain the incidence. If the method does not work well, the difference between the upper and lower bounds may be quite large. I have found the method to produce quite good results in some realistic situations; see Section 5.2 for an example of this.

The method is based on the following observation. If we apply the condition that some patch or patches in a metapopulation are always empty, the incidence of occupancy cannot increase for any of the other patches. On the contrary, if we apply the condition that some patches are always occupied the incidence of the other patches will not decrease. A corollary to the following theorem proves this statement. First we introduce some notation used throughout this section.

We will use the $(n \times n)$ interaction matrices $Q = (q_{ij})$ and $\widetilde{Q} = (\widetilde{q}_{ij})$ to describe two metapopulations for which $q_{ij} \geq \widetilde{q}_{ij}$, $i, j \in \{1, 2, \ldots, n\}$. We write $q_i(\overline{x})$ and $\widetilde{q}_i(\overline{x})$ for the probabilities that patch i is empty one time step after the respective systems are in state $\overline{x}$. Analogously we write $P(\overline{x}, \overline{y})$ and $\widetilde{P}(\overline{x}, \overline{y})$ for the corresponding state transition probabilities, and J_i, $\widetilde{J}_i$ for the incidences of occupancy.

Let $\overline{e}_i$ denote the vector $\left[\begin{array}{ccc} e_1 & \cdots & e_n \end{array}\right]$, for which $e_i = 1$ and $e_j = 0$, $j \neq i$.

For any states $\overline{x}$ and $\overline{y}$ we write $\overline{x} \geq \overline{y}$ if $\forall i : x_i \geq y_i$, and similarly $\overline{x} > \overline{y}$ if $\overline{x} \geq \overline{y} \wedge \overline{x} \neq \overline{y}$.

We will now prove that the incidence is a monotonically increasing function of the elements in Q.

Lemma 1 *If $\overline{x} \geq \overline{y} \Rightarrow f(\overline{x}) \geq f(\overline{y})$ and $\widetilde{q}_i \leq q_i$ for $i \in \{1, 2, \ldots, n\}$, then*

$$\sum_{\overline{y} \in D} f(\overline{y}) \left(\prod_{i=1}^{n} \widetilde{q}_i^{1-y_i} (1 - \widetilde{q}_i)^{y_i} - \prod_{i=1}^{n} q_i^{1-y_i} (1 - q_i)^{y_i} \right) \geq 0.$$

Proof. First define ${}_k q_i$ by

$${}_k q_i = \begin{cases} \widetilde{q}_i, & i \leq k \\ q_i, & i > k \end{cases} .$$

Two special cases are ${}_n q_i = \widetilde{q}_i$ and ${}_0 q_i = q_i$. We now prove that

$$\Sigma_k = \sum_{\overline{y} \in D} f(\overline{y}) \left(\prod_{i=1}^{n} {}_k q_i^{1-y_i} (1 - {}_k q_i)^{y_i} - \prod_{i=1}^{n} q_i^{1-y_i} (1 - q_i)^{y_i} \right) \geq 0$$

for $k \in \{0, 1, \ldots, n\}$. We have that $\Sigma_0 = 0$, and prove that $\Sigma_p \geq \Sigma_{p-1}$, $p \leq n$, so that we have $\Sigma_k \geq 0$ for all k.

$$\Sigma_p - \Sigma_{p-1} =$$

$$= \sum_{\overline{y} \in D} f(\overline{y}) \left(\prod_{i=1}^{n} {}_p q_i^{1-y_i} (1 - {}_p q_i)^{y_i} - \prod_{i=1}^{n} {}_{p-1} q_i^{1-y_i} (1 - {}_{p-1} q_i)^{y_i} \right) =$$

$$= \sum_{\overline{y} \in D | y_p = 0} \left[f(\overline{y}) \left(\widetilde{q}_p \prod_{i \neq p} {}_p q_i^{1-y_i} (1 - {}_p q_i)^{y_i} - q_p \prod_{i \neq p} {}_p q_i^{1-y_i} (1 - {}_p q_i)^{y_i} \right) + \right.$$

$$\left. + f(\overline{y} + \overline{e}_p) \left((1 - \widetilde{q}_p) \prod_{i \neq p} {}_p q_i^{1-y_i} (1 - {}_p q_i)^{y_i} - (1 - q_p) \prod_{i \neq p} {}_p q_i^{1-y_i} (1 - {}_p q_i)^{y_i} \right) \right] =$$

$$= \sum_{\overline{y} \in D | y_p = 0} \left(f(\overline{y})(\widetilde{q}_p - q_p) + f(\overline{y} + \overline{e}_p)(q_p - \widetilde{q}_p) \right) \prod_{i \neq p} {}_p q_i^{1-y_i} (1 - {}_p q_i)^{y_i} \geq$$

$$\geq \sum_{\overline{y} \in D | y_p = 0} \left[f(\overline{y}) \left(\widetilde{q}_p - q_p + q_p - \widetilde{q}_p \right) \prod_{i \neq p} {}_p q_i^{1-y_i} (1 - {}_p q_i)^{y_i} \right] = 0.$$

The inequality uses that $f(\overline{y} + \overline{e}_p) \geq f(\overline{y})$ and that $q_p \geq \widetilde{q}_p$. This completes the proof. □

Theorem 1 *Suppose $Q = (q_{ij})$ and $\widetilde{Q} = (\widetilde{q}_{ij})$ are the $(n \times n)$ interaction matrices for two metapopulations such that $q_{ij} \geq \widetilde{q}_{ij}$, $i, j \in \{1, 2, \ldots, n\}$. Then*

$$\sum_{\overline{y} \geq \overline{e}_l} \left(\widetilde{P}^k(\overline{x}, \overline{y}) - P^k(\overline{x}, \overline{y}) \right) \geq 0$$

for all k, l and $\overline{x}$.

Proof. To prove this statement, we first prove that

$$\overline{x} \geq \overline{y} \Rightarrow \sum_{\overline{z} \geq \overline{e}_l} \left(P^k(\overline{x}, \overline{z}) - P^k(\overline{y}, \overline{z}) \right) \geq 0. \tag{3.4}$$

This is done by induction over k. When $k = 0$, the inequality is true by the definition of the state transition probabilities (equation (2.3)). Now suppose that it holds for $k = p$. We prove that it also holds for $k = p + 1$. For $\overline{x} \geq \overline{y}$,

$$\sum_{\overline{z} \geq \overline{e}_l} \left(P^{p+1}(\overline{x}, \overline{z}) - P^{p+1}(\overline{y}, \overline{z}) \right) =$$

$$= \sum_{\overline{z} \geq \overline{e}_l} \left(\sum_{\overline{s} \in D} P(\overline{x}, \overline{s}) P^p(\overline{s}, \overline{z}) - \sum_{\overline{s} \in D} P(\overline{y}, \overline{s}) P^p(\overline{s}, \overline{z}) \right) =$$

$$= \sum_{\overline{s} \in D} \left((P(\overline{x}, \overline{s}) - P(\overline{y}, \overline{s})) \sum_{\overline{z} \geq \overline{e}_l} P^p(\overline{s}, \overline{z}) \right) \geq 0,$$

where the inequality is given by putting $q_i = q_i(\overline{y})$, $\widetilde{q}_i = q_i(\overline{x})$ and $f(\overline{s}) = \sum_{\overline{z} \geq \overline{e}_l} P^p(\overline{s}, \overline{z})$. Since $\overline{x} \geq \overline{y}$ we have $q_i(\overline{x}) = q_i(\overline{y}) \cdot \prod_{x_j > y_j} q_{ij} \leq q_i(\overline{y})$. We also have $\overline{x} \geq \overline{y} \Rightarrow f(\overline{x}) \geq f(\overline{y})$ by the induction hypothesis. Lemma 1 then proves the inequality.

Now the theorem itself is proved by induction over k. For the case $k = 0$, it is true by the definition of the state transition probabilities. Suppose it holds for $k = p$. We will prove that it holds also for $k = p + 1$.

$$\sum_{\overline{y}\geq\overline{e}_l} \left(\widetilde{P}^{p+1}(\overline{x},\overline{y}) - P^{p+1}(\overline{x},\overline{y})\right) =$$

$$= \sum_{\overline{y}\geq\overline{e}_l} \left(\sum_{\overline{s}\in D} \widetilde{P}(\overline{x},\overline{s})\widetilde{P}^p(\overline{s},\overline{y}) - \sum_{\overline{s}\in D} P(\overline{x},\overline{s})P^p(\overline{s},\overline{y})\right) =$$

$$= \sum_{\overline{s}\in D} \left(\widetilde{P}(\overline{x},\overline{s}) \sum_{\overline{y}\geq\overline{e}_l} \widetilde{P}^p(\overline{s},\overline{y}) - P(\overline{x},\overline{s}) \sum_{\overline{y}\geq\overline{e}_l} P^p(\overline{s},\overline{y})\right) \geq$$

$$\geq \sum_{\overline{s}\in D} \left(\left(\widetilde{P}(\overline{x},\overline{s}) - P(\overline{x},\overline{s})\right) \sum_{\overline{y}\geq\overline{e}_l} P^p(\overline{s},\overline{y})\right) \geq 0,$$

where the first inequality is given by the induction hypothesis. The second inequality is given by using Lemma 1 again, this time with $q_i = q_i(\overline{x})$, $\widetilde{q}_i = \widetilde{q}_i(\overline{x})$ and $f(\overline{s}) = \sum_{\overline{y}\geq\overline{e}_l} P^p(\overline{s},\overline{y})$, which makes $\widetilde{q}_i \leq q_i$ and $\overline{x} \geq \overline{y} \Rightarrow f(\overline{x}) \geq f(\overline{y})$ by the inequality (3.4). This completes the proof. □

Corollary 1 *We have that $\widetilde{J}_i \geq J_i$, $i \in \{1, 2, \dots, n\}$, where J_i and $\widetilde{J}_i$ denote the incidences of occupancy for patches in the metapopulations described by the interaction matrices Q and $\widetilde{Q}$, respectively.*

Proof. From equation (2.7) we have

$$\widetilde{J}_i - J_i = \sum_{\overline{x}\in D|x_i=1} (\widetilde{\pi}(\overline{x}) - \pi(\overline{x})) = \sum_{\overline{x}\geq\overline{e}_i} (\widetilde{\pi}(\overline{x}) - \pi(\overline{x})) =$$

$$= \lim_{k\to\infty} \sum_{\overline{y}\geq\overline{e}_i} \left(\widetilde{P}^k(\overline{x},\overline{y}) - P^k(\overline{x},\overline{y})\right).$$

Theorem 1 completes the proof. □

To consider a metapopulation with interaction matrix Q under the assumption that the local populations on one or more patches cannot go extinct, we can construct a new interaction matrix $\widetilde{Q}$, where $\widetilde{q}_{ij} = q_{ij}$, except that $\widetilde{q}_{ii} = 0$ for some is. Thus $\widetilde{q}_{ij} \leq q_{ij}$ and the incidences of occupancy for all patches in the new system is larger than or equal to those in the original system.

If, on the other hand, we want to examine a metapopulation under the assumption that one or more patches cannot become colonized (or, equivalently, are always empty), we can construct a new interaction matrix $\widetilde{Q}$, where $\widetilde{q}_{ij} = q_{ij}$, except that $\widetilde{q}_{ij} = 1$ for some js. Here $q_{ij} \leq \widetilde{q}_{ij}$ and the incidences of occupancy for all patches in the new system will be lower than or equal to those in the original system.

We write $J_k^-(I)$ to denote the probability that patch k is occupied in a metapopulation in which only patches in the set I can become colonized. This is according to the argument above a lower bound for J_k. For $0 \leq m \leq n-2$ we can now calculate successively better lower bounds for the incidence of occupancy as follows.

1. For $i = 0$, define $I_0^- = \{1, k\}$.

2. For $i < m$, define I_{i+1}^- from I_i^- by

$$I_{i+1}^- = I_i^- \cup \left\{ \underset{j \in \{1,2,\ldots,n\} \setminus I_i}{\text{argmax}} \; J_k^- \left(I_i^- \cup \{j\} \right) \right\}.$$

Here $\text{argmax}_{x \in A} f(x)$ denotes any member x in A such that $f(y) \leq f(x)$, $y \neq x$. We first require that all patches except patch k and the mainland are always empty. The condition is then released, patch by patch, a total of m times. Of course the algorithm for choosing the patch where to enable colonization is greedy, but works well and guarantees that $J_k^-(I_m^-)$ is a lower bound for J_k.

Analogously, I_m^+ is chosen so that $J_k^+(I_m^+)$ is an upper bound for J_k. The only difference is that we write $J_k^+(I)$ for the probability that patch k is occupied in a metapopulation where all patches not in the set I are always occupied.

1. For $i = 0$, define $I_0^+ = \{1, k\}$.

2. For $i < m$, define I_{i+1}^+ from I_i^+ by

$$I_{i+1}^+ = I_i^+ \cup \left\{ \underset{j \in \{1,2,\ldots,n\} \setminus I_i}{\text{argmin}} \; J_k^+ \left(I_i^+ \cup \{j\} \right) \right\}.$$

Note that $J_k^+(I)$ and $J_k^-(I)$ can be calculated by analysing a Markov chain with only $2^{|I|-1}$ states. Appendix B.2 contains Matlab programs to perform the iterative procedures described above.

If used on a metapopulation where dispersal between local populations is not too large, this method can provide very close upper and lower bounds for the incidence. If, on the other hand, all patches are closely connected to each other, the stationary probability distribution may be drastically changed when conditions as those above are applied. In the latter case the method may not work very well, but can, however, often give some hints on how the metapopulation will behave.

3.3 Error Analysis

Whenever we consider a real metapopulation, where patch sizes and their spatial locations are all known, and use some model to calculate the interaction matrix $Q = \begin{bmatrix} \overline{q}_1 & \overline{q}_2 & \cdots & \overline{q}_n \end{bmatrix}$, usually Q does not satisfy the lumping conditions of section 3.1. Since the system (2.4) is practically impossible to solve in many cases, a possible approach is to construct a matrix $Q^* = \begin{bmatrix} \overline{q}_1^* & \overline{q}_2^* & \cdots & \overline{q}_n^* \end{bmatrix}$ that satisfies the lumping conditions, but still is close to the original Q. If we remove patches from the system to reduce the number of states in the Markov chain, this is equivalent to analysing a matrix Q^* which differs from Q.

It is the purpose of this section to investigate how the difference between Q and Q^* affects the stationary distribution of the number of occupied patches and the incidence of occupancy for the patches.

We require that $0 \leq q_{ji}^* \leq 1$ for all $i, j \in \{1, 2, \ldots, n\}$ and define the $n \times n$ matrix $E = (e_{ij})$, where $e_{ij} = q_{ij}^*/q_{ij}$, and also the indicators

$$B(x) = \begin{cases} 0, & x < 1 \\ 1, & x \geq 1 \end{cases},$$

$$y_u(i) = \begin{cases} 0, & \prod_{j=1}^n e_{ji}^{B(e_{ji})} \geq \frac{1-\prod_{j=1}^n q_{ji}^*}{1-\max_j q_{ji}} \\ 1, & \text{otherwise} \end{cases},$$

and

$$y_l(i) = \begin{cases} 0, & \prod_{j=1}^n e_{ji}^{1-B(e_{ji})} \leq \frac{1-\max_j q_{ji}}{1-\prod_{j=1}^n q_{ji}^*} \\ 1, & \text{otherwise} \end{cases}.$$

To determine a relative upper bound for the error in the Markov chain transition probabilities $P(\overline{x}, \overline{y})$ we first notice the inequalities

$$\frac{P^*(\overline{x}, \overline{y})}{P(\overline{x}, \overline{y})} = \prod_{i=1}^n \frac{q_i^*(\overline{x})^{1-y_i}(1-q_i^*(\overline{x}))^{y_i}}{q_i(\overline{x})^{1-y_i}(1-q_i(\overline{x}))^{y_i}} =$$

$$= \prod_{i=1}^n \left(\prod_{j=1}^n e_{ji}^{x_j} \right)^{1-y_i} \cdot \prod_{i=1}^n \left(\frac{1-\prod_{j=1}^n {q_{ji}^*}^{x_j}}{1-\prod_{j=1}^n {q_{ji}}^{x_j}} \right)^{y_i} \leq$$

$$\leq \prod_{i=1}^n \left(\prod_{j=1}^n e_{ji}^{B(e_{ji})} \right)^{1-y_i} \cdot \prod_{i=1}^n \left(\frac{1-\prod_{j=1}^n q_{ji}^*}{1-\max_j q_{ji}} \right)^{y_i} \leq$$

$$\leq \prod_{i=1}^n \left(\prod_{j=1}^n e_{ji}^{B(e_{ji})} \right)^{1-y_u(i)} \cdot \prod_{i=1}^n \left(\frac{1-\prod_{j=1}^n q_{ji}^*}{1-\max_j q_{ji}} \right)^{y_u(i)} = \delta_u, \tag{3.5}$$

and

$$\frac{P^*(\overline{x},\overline{y})}{P(\overline{x},\overline{y})} = \prod_{i=1}^{n} \frac{q_i^*(\overline{x})^{1-y_i}(1-q_i^*(\overline{x}))^{y_i}}{q_i(\overline{x})^{1-y_i}(1-q_i(\overline{x}))^{y_i}} =$$

$$= \prod_{i=1}^{n}\left(\prod_{j=1}^{n} e_{ji}^{x_j}\right)^{1-y_i} \cdot \prod_{i=1}^{n}\left(\frac{1-\prod_{j=1}^{n} q_{ji}^{*\,x_j}}{1-\prod_{j=1}^{n} q_{ji}^{x_j}}\right)^{y_i} \geq$$

$$\geq \prod_{i=1}^{n}\left(\prod_{j=1}^{n} e_{ji}^{1-B(e_{ji})}\right)^{1-y_i} \cdot \prod_{i=1}^{n}\left(\frac{1-\max_j q_{ji}}{1-\prod_{j=1}^{n} q_{ji}^{*}}\right)^{y_i} \geq$$

$$\geq \prod_{i=1}^{n}\left(\prod_{j=1}^{n} e_{ji}^{1-B(e_{ji})}\right)^{1-y_l(i)} \cdot \prod_{i=1}^{n}\left(\frac{1-\max_j q_{ji}}{1-\prod_{j=1}^{n} q_{ji}^{*}}\right)^{y_l(i)} = \delta_l. \tag{3.6}$$

From (3.5) and (3.6) it is now clear that

$$\delta_l - 1 \leq \frac{P^*(\overline{x},\overline{y}) - P(\overline{x},\overline{y})}{P(\overline{x},\overline{y})} \leq \delta_u - 1 \Rightarrow$$

$$\Rightarrow \left|\frac{P^*(\overline{x},\overline{y}) - P(\overline{x},\overline{y})}{P(\overline{x},\overline{y})}\right| \leq \max\left\{1-\delta_l, \delta_u - 1\right\},$$

and that we can pick δ so that $|P^*(\overline{x},\overline{y}) - P(\overline{x},\overline{y})| < \delta P(\overline{x},\overline{y})$ holds for all $\overline{x},\overline{y} \in X$. If we define $\Delta P = P^* - P$, where P is the matrix form of the transition probabilities, we have that

$$\left\|\Delta P^T\right\|_1 < \delta \left\|P^T\right\|_1 = \delta, \tag{3.7}$$

where the last equality uses that P is a probability matrix and has all row sums equal to 1.

We will now consider an alternative way to compute the stationary distribution given by (2.4). First rewrite the equation in the form

$$\overline{0} = \overline{\pi}(P - I_{2^{n-1}}) \Leftrightarrow (P^T - I_{2^{n-1}})\overline{\pi}^T = \widetilde{P}\overline{\pi} = \overline{0}, \quad \widetilde{P} = (\widetilde{p}_{ij}). \tag{3.8}$$

Since P has all row sums equal to 1, $\widetilde{P}$ has all column sums equal to 0. This implies that row 2^{n-1} of $\widetilde{P}$ can be written as the negative sum of all other rows; row 2^{n-1} is thus redundant. Define the $(2^{n-1} \times 2^{n-1})$ matrix A to have rows $1, 2, \ldots, 2^{n-1} - 1$ equal to the corresponding rows in $\widetilde{P}$, but to have all ones in row 2^{n-1}. The equation

$$A\overline{\pi}^T = \begin{bmatrix} \widetilde{p}_{1,1} & \cdots & \widetilde{p}_{1,2^{n-1}} \\ \vdots & & \vdots \\ \widetilde{p}_{2^{n-1}-1,1} & \cdots & \widetilde{p}_{2^{n-1}-1,2^{n-1}} \\ 1 & \cdots & 1 \end{bmatrix} \overline{\pi}^T = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} = \overline{b}$$

has the same solution as (3.8) with the additional condition $\sum_{i=1}^{2^{n-1}} \pi_i = 1$. Define A^* from P^* in the same way and also $\Delta A = A^* - A$. Now we have

$$\|\Delta A\|_1 \leq \left\|\Delta P^T\right\|_1,$$

and from (3.7)

$$\left\|\Delta P^T\right\|_1 < \delta,$$

which together lead to

$$\|\Delta A\|_1 < \delta.$$

We can now make an estimate of $\left\|(\overline{\pi}^* - \overline{\pi})^T\right\|_1$, where $\overline{\pi}^*$ is the solution to $A^*(\overline{\pi}^*)^T = \overline{b}$ as follows. Since

$$\overline{\pi}^* - \overline{\pi} = \overline{\pi}^* - \overline{b}^T(A^{-1})^T = \overline{\pi}^* - \overline{\pi}^*(A^{-1}A^*)^T =$$

$$= \overline{\pi}^* - \overline{\pi}^*(A^{-1}(A + \Delta A))^T = -\overline{\pi}^*(A^{-1}\Delta A)^T,$$

and $\left\|(\overline{\pi}^*)^T\right\|_1 = 1$, we have that

$$\left\|(\overline{\pi}^* - \overline{\pi})^T\right\|_1 \leq \|\Delta A\|_1 \left\|A^{-1}\right\|_1 \left\|(\overline{\pi}^*)^T\right\|_1 < \delta \left\|A^{-1}\right\|_1. \tag{3.9}$$

To determine the error in the approximation of the number of occupied patches $p^*(k) = \sum_{\rho(\phi(i))=k} \pi_i^*$ (cf. eq (2.6)), we consider

$$\sum_{k=1}^{n} |p^*(k) - p(k)| = \sum_{k=1}^{n} \left| \sum_{\rho(\phi(i))=k} \pi_i^* - \pi_i \right| \leq \sum_{k=1}^{n} \sum_{\rho(\phi(i))=k} |\pi_i^* - \pi_i| =$$

$$= \sum_{i=1}^{2^{n-1}} |\pi_i^* - \pi_i| = \left\|(\overline{\pi}^* - \overline{\pi})^T\right\|_1 < \delta \left\|A^{-1}\right\|_1,$$

and in a similar way an error bound for the approximated incidence $J_i^* = \sum_{\phi(j)_i=1} \pi_j^*$ (cf. eq (2.8)) is

$$\sum_{i=1}^{n} |J_i^* - J_i| = \sum_{i=1}^{n} \left| \sum_{\phi(j)_i=1} \pi_j^* - \pi_j \right| \leq \sum_{i=1}^{n} \sum_{\phi(j)_i=1} \left|\pi_j^* - \pi_j\right| =$$

$$= \sum_{j=1}^{2^{n-1}} \sum_{\phi(j)_i=1} \left|\pi_j^* - \pi_j\right| \leq n \left\|(\overline{\pi}^* - \overline{\pi})^T\right\|_1 < n\delta \left\|A^{-1}\right\|_1.$$

Unfortunately $\|A^{-1}\|_1$ is practically impossible to calculate when n is so large that we have to make approximations in the first place. Also, even if $\|A^{-1}\|_1$

could be successfully evaluated, the resulting error limits are very rough. The lumping of patches works acceptably well in many situations, but the error bounds calculated can be several hundred percents in these cases. Anyway, the analysis above shows that the distribution of the number of occupied patches and the incidence of occupancy are continuous functions with respect to the elements in Q.

Chapter 4

Simulation

Because of the difficulty in obtaining analytical results that describe the metapopulation properties, an attracting method to get reasonably accurate estimates of the distribution of the number of occupied patches and the incidence is to make some sort of simulation of the system. In this chapter a method for this is presented together with analysis of the rate of convergence.

4.1 Simulation Method

The simulation method described herein generates a trajectory $\{\overline{\eta}(t)\}_{t=1}^{N}$ from an initial state $\overline{\eta}(0)$ that can be chosen arbitrarily so that $\overline{\eta}(0) \in D$. As N approaches infinity, the distribution of the number of times each state is visited approaches the stationary distribution of state probabilities. In Section 4.2 the rate of convergence is analyzed.

The fundamental step in the simulation method is a function that, given $\overline{\eta}(t)$ by random determines $\overline{\eta}(t+1)$. Since the total number of possible states is very large, it is not possible to record the number of times that each state has been visited. However, this is not necessary to get estimates of the distribution of the number of occupied patches and the incidences of occupancy; it suffices to count the number of times that each of the patches has been inhabited (for calculating the incidences), and the number of times that the metapopulation has had each number of patches inhabited during the simulation process. The counts are divided by N to get estimates of J_i ($\widehat{J}_i$) and $p(i)$ ($\widehat{p}(i)$). Below is a step-by-step description of the simulation method.

1. Set current time $t = 0$, set the current state to the initial state ($\overline{x} = \overline{\eta}(0)$), and also set $p^c(i) = J_i^c = 0,\ i \in \{1, 2, \ldots, n\}$.

2. From the current state $\overline{x}$, calculate the next state $\overline{y}$ in the trajectory:

(a) Set all patches uninhabited ($\overline{y} = \overline{0}$).

(b) For each $(i, j) \in \{1, 2, \ldots, n\} \times \{1, 2, \ldots, n\}$, determine by random if patch i colonizes patch j. This is done by drawing a pseudo random number r from $U(0, 1)$; if it is larger than q_{ij} then we assign $\overline{y}_j \leftarrow 1$. (We have that the probability that patch i colonizes patch j in one time step is $1 - q_{ij} = 1 - P(r \leq q_{ij}) = P(r > q_{ij})$.) For an extensive investigation of pseudo random number generation, see [8].

3. Increase current time by one, and assign the new state to the current ($\overline{x} \leftarrow \overline{y},\ t \leftarrow t + 1$).

4. Increase the counts of the number of times a specific number of patches have been occupied and of the number of times the patches have been inhabited ($p^c(\sum_{i=1}^{n} x_i) \leftarrow p^c(\sum_{i=1}^{n} x_i) + 1$ and $J_i^c = J_i^c + x_i,\ i \in \{1, 2, \ldots, n\}$).

5. If $t < N$, continue at step 2.

6. Calculate the estimates $\widehat{J}_i = J_i^c/N$ and $\widehat{p}(i) = p^c(i)/N$, $i \in \{1, 2, \ldots, n\}$.

Appendix C contains a simulation program in the C programming language to simulate the behavior of a metapopulation by the means of the method described above.

4.2 Error Analysis

Since the Markov chain is ergodic, the probabilities $P(\eta_i(t) = 1) = E[\eta_i(t)]$, $i \in \{1, 2, \ldots, n\}$ converges towards the stationary incidence of occupancy J_i given any initial state $\eta(0)$, and the rate of convergence is geometric. In fact, if α is the second largest eigenvalue for the state transition matrix P, then

$$|E[\eta_i(t)] - J_i| \leq C\alpha^t,$$

where C is a constant that can be determined from P [2]. We write $\widehat{J}_i\ (k)$ for the estimate given by simulating k steps. Formally $\widehat{J}_i(k)$ is

$$\widehat{J}_i(k) = \frac{\sum_{j=1}^{k} \eta_i(j)}{k},$$

with expectation

$$E\left[\widehat{J}_i(k)\right] = E\left[\frac{\sum_{j=1}^{k} \eta_i(j)}{k}\right] = \frac{\sum_{j=1}^{k} E[\eta_i(j)]}{k}.$$

The estimate is evidently biased, and we consider

$$\left|E\left[\widehat{J}_i(k)\right] - J_i\right| = \left|\frac{\sum_{j=1}^{k}\left(E\left[\eta_i(j)\right] - J_i\right)}{k}\right| \leq$$

$$\leq \frac{\sum_{j=1}^{k}\left|E\left[\eta_i(j)\right] - J_i\right|}{k} \leq \frac{\sum_{j=1}^{k} C\alpha^j}{k} \leq \frac{C}{k(1-\alpha)}, \tag{4.1}$$

which tends to 0 as $k \to \infty$.

To determine the probability that the estimate differs from the real incidence more than ϵ after k simulation steps, Chebysjev's inequality gives

$$P\left(\left|\widehat{J}_i(k) - J_i\right| > \epsilon\right) \leq \frac{E\left[\left(\widehat{J}_i(k) - J_i\right)^2\right]}{\epsilon^2}. \tag{4.2}$$

To bound the right-hand value of this inequality we consider

$$\frac{E\left[\left(\widehat{J}_i(k) - J_i\right)^2\right]}{\epsilon^2} = \frac{\mathrm{Var}\left[\widehat{J}_i(k)\right] + \left(E\left[\widehat{J}_i(k)\right] - J_i\right)^2}{\epsilon^2},$$

which can be bounded with the help of (4.1) and the following evaluation of $\mathrm{Var}\left[\widehat{J}_i(k)\right]$:

$$\mathrm{Var}\left[\widehat{J}_i(k)\right] = \mathrm{Var}\left[\frac{1}{k}\sum_{j=1}^{k}\eta_i(j)\right] =$$

$$\frac{1}{k}\sum_{j=1}^{k}\frac{1}{k}\mathrm{Var}\left[\eta_i(j)\right] + \frac{1}{k}\sum_{j\neq l}\frac{1}{k}\mathrm{Cov}\left[\eta_i(j), \eta_l(l)\right],$$

where we have

$$\mathrm{Var}\left[\eta_i(j)\right] = E\left[\eta_i(j)^2\right] - E\left[\eta_i(j)\right]^2 = E\left[\eta_i(j)\right]\left(1 - E\left[\eta_i(j)\right]\right) \leq \frac{1}{4}$$

since $0 \leq E\left[\eta_i(j)\right] \leq 1$, and from [2]:

$$\mathrm{Cov}\left[\eta_i(j), \eta_l(l)\right] \leq C\alpha^{|j-l|}$$

Thus we have

$$\mathrm{Var}\left[\widehat{J}_i(k)\right] \leq \frac{1}{k}\sum_{j=1}^{k}\frac{1}{4k} + \frac{1}{k}\sum_{j\neq l}\frac{1}{k}C\alpha^{|j-l|} =$$

$$\frac{1}{4k} + \frac{2}{k}\sum_{j=1}^{k-1}\frac{k-j}{k}C\alpha^j \leq \frac{1}{4k} + \frac{2C}{k(1-\alpha)}.$$

Together with equation (4.2) we can now calculate the probability of the simulation error being greater than ϵ:

$$P\left(\left|\widehat{J}_i(k) - J_i\right| > \epsilon\right) \leq \frac{1}{4k\epsilon^2} + \frac{2C}{k(1-\alpha)\epsilon^2} + \frac{C^2}{k^2(1-\alpha)^2\epsilon^2}. \tag{4.3}$$

Methods for finding the constants C and α given the state transition probability matrix can be found in [2]. In our case, however, the size of this matrix is too large to perform computations on it. It is a subject for further research to develop methods for finding upper bounds for these constants.

Chapter 5

Applications

In the metapopulations analysed in this chapter the interaction matrix Q is calculated using a model that is based on models used in [5] and [6]. Let d_{ij} be the distance from patch i to patch j ($d_{ii} = 0$), and A_i the area of patch i. The distance between two patches is measured from the center of one patch to the center of the other, but any other choices for measuring the distance would be possible. It is not necessary that $d_{ij} = d_{ji}$; wind or other factors could be incorporated in the distance measure. The local extinction probabilities q_{ii}, $i \in \{1, 2, \ldots, n\}$ are

$$q_{ii} = e^{-cA_i},$$

and the probability that migrators from patch i does not colonize patch j in one time step, q_{ij}, is

$$q_{ij} = e^{-bA_i e^{-ad_{ij}}},$$

where a, b and c are model parameters. The parameter a is a measure of how bad the species is at migrating long distances. The parameter b describes how densely individuals inhabit the patches, and finally c is a measure of how fast the local extinction probabilities decrease with increasing patch size. Since all local extinction probabilities are non-zero, there is no mainland in a system that uses this model. In the analysis in this chapter, we will consider the stationary probabilities in a somewhat modified metapopulation model. The lowest local extinction probability will be changed to zero, which is equivalent to analysing the system under the condition that one of the patches is always occupied. Results in [5] show that we can expect a quasi-stationary behavior of the original system that could be analysed in this way.

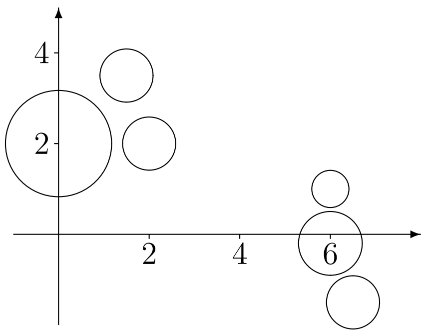

Figure 5.1: Spatial locations of patches in the hypothetical metapopulation of the example.

Patch no. (i)	**Area**	**Position**	J_i
1	4	(0,2)	1.0000
2	1	(2,2)	0.9454
3	1.5	(6,-0.2)	0.6142
4	1	(1.5,3.5)	0.9357
5	1	(6.5,-1.5)	0.5853
6	0.5	(6,1)	0.4429

Table 5.1: Geographical data and predicted incidence of occupancy for the example metapopulation.

5.1 A Small Patch System

In this example, we will consider the system of patches that is depicted in Figure 5.1. The model parameters are $a = b = 2.5$ and $c = 5$. We now use equations (2.1) and (2.2), define the transition probability matrix P and solve equation (2.4). Since this system contains only six patches, $|D| = 2^{6-1} = 32$, and P is a (32×32) matrix. Now equations (2.6) and (2.8) are used to obtain the distribution of the number of occupied patches and the incidences of occupancy for all patches. Table 5.1 shows the exact areas and locations of the patches and the incidence J_i as well. We consider the system under the assumption that patch number 1 is always occupied (patch number one is a mainland), since this patch has the lowest local extinction probability. Figure 5.2 shows the distribution of the number of occupied patches, which in this case is a bimodal function.

As we will now see, the bimodal distribution of the number of occupied patches does not always imply a core-satellite distribution in the original system where the mainland is actually a quasi-mainland and is not supposed to be inhabited forever. It merely says that it is very probable that the metapopulation will have either quite a lot, or just a few occupied patches. The "core-satellite switches" may be very rare, as is the case in this example. If the metapopulation is observed in nature, we will most probably not be able to observe any switches at all under

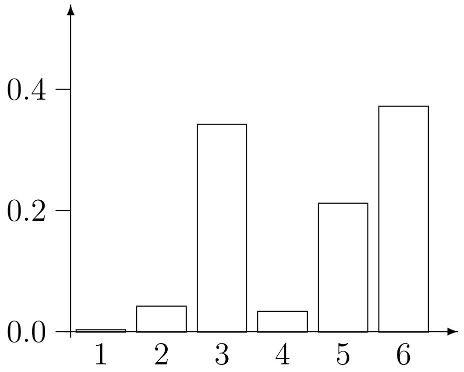

Figure 5.2: Predicted stationary distribution of the number of occupied patches in the hypothetical metapopulation from Section 5.1.

a reasonably short time of observation. In some systems the switches may be so rare that the metapopulation has most probably become extinct before any switches actually occur and we can think of the system as having two different quasi-stationary probability distributions.

The patches are clearly lumped geographically in two clusters. Sometimes all patches in the cluster not containing the mainland are empty, and sometimes one or more of them are occupied. Introduce the indicator

$$\theta(\overline{\eta}(t)) = \theta(t) = \begin{cases} 0, & \eta_3 = \eta_5 = \eta_6 = 0 \\ 1, & \text{otherwise} \end{cases}$$

to distinguish between these cases. Now apply equation (2.16) to calculate the power spectrum for the Moore machine that has $\theta(t)$ as output. The total energy is

$$\int_{-\frac{1}{2}}^{\frac{1}{2}} R(f)df \approx 0.2313$$

and the energy for signal components that have a period larger than 1000 is

$$\int_{-0.001}^{0.001} R(f)df \approx 0.2215,$$

which is approximately 0.96 times the total energy.

If we try to simulate the metapopulation instead of using algebraic methods to investigate it, then one possible result is the one that has been depicted in Figure 5.3. The estimate of the incidence of patch number 3, $\widehat{J}_3$, is plotted against the number of simulation steps. We see that after about 5 million steps, the simulation does not seem to converge anymore, or at least converges very slowly.

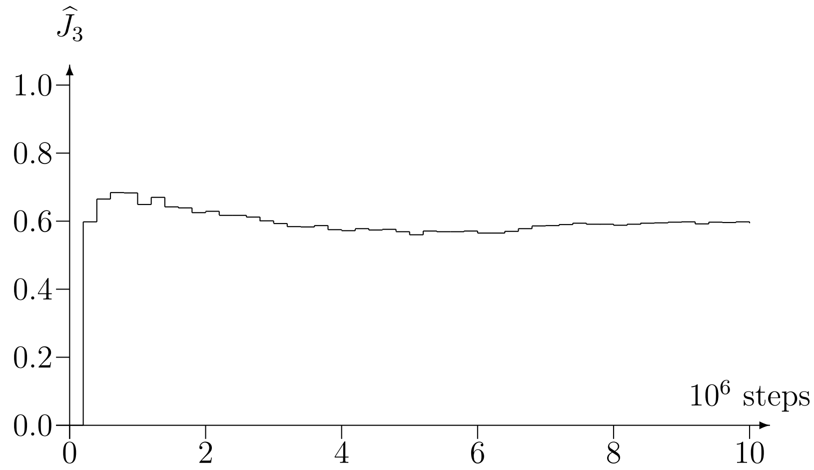

Figure 5.3: Result when simulating the hypothetical metapopulation from Section 5.1. The estimate of the incidence of patch number 3, $\widehat{J}_3$, is plotted against the number of simulation steps.

It seems to be very common that systems that switch from having only a few inhabited patches to having many inhabited patches with a very low frequency are difficult to simulate. The problem is to determine when to stop the simulation process.

5.2 A Population of *Melitaea Cinxia*

This section considers the metapopulation that is investigated in [7]. In an area of 15.5 km^2 in the southwest of Finland, 50 habitat patches were surveyed for the presence or absence of the butterfly *Melitaea cinxia.* Figure 5.4 is a map of the study region, where the habitat patches have been approximated to circular discs. Geographical data for the area can be found in Table 5.2.

To make any analysis, we have to choose some values for the model parameters. I have found that $a = c = 0.001$ and $b = 0.00001$ is a reasonable choice, that makes the metapopulation behave in a realistic way. With this choice, the local extinction probability for a population on patch number 9 is of the order 10^{-20}. We choose this patch as a mainland, and thus neglect the possibility that this patch may become extinct. The Markov chain that represents the metapopulation has 2^{50-1} states, which makes all of the formulas presented in Chapter 2 useless for practical purposes. Instead, the method to obtain lower and upper bounds for the incidence of occupancy for the patches from Section 3.2 is used.

Table 5.2 shows the results obtained by using the method with $m = 4$, which

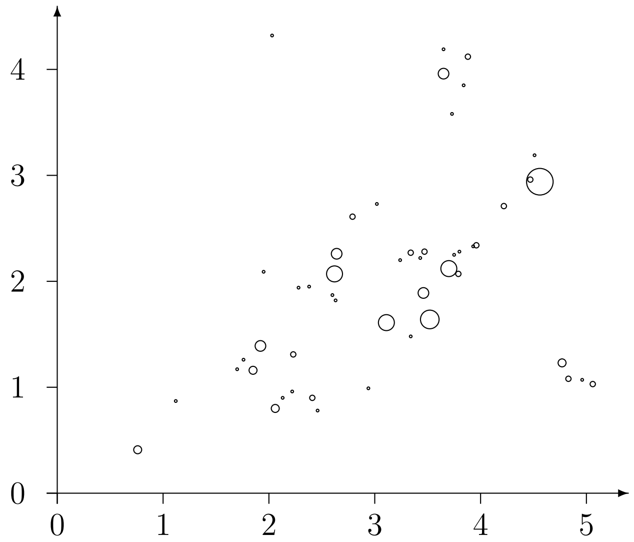

Figure 5.4: Spatial locations of patches in the metapopulation of *Melitaea cinxia* from [7]. Numerical data can be found in Table 5.2.

means systems of up to 6 patches have been analysed. The total time for all calculations in this table was less than one hour using `.m`-files in Matlab$^{\rm TM}$ on a Sun Sparc 2 computer. I have found that the results get only slightly better when m is changed from say 3 or 4 up to 6 or 7. The former choice of course allows for much faster computations (with $m = 7$, the same computer took 104 hours to complete the calculations, with only marginal improvements).

Figure 5.5 shows the relation between patch size and incidence of occupancy. The value for the incidence has been obtained by taking the mean value of the limits from Table 5.2. It can be seen that our model predicts an s-shaped incidence-area relation; this relation can be predicted by other models of immigration-extinction equilibria, as pointed out in [1].

Patch no. (i)	**Area** (m^2)	**Position** (km)	$J_i^-(I_4^-)$	$J_i^+(I_4^+)$	**Status**[a]
1	26620	(3.52,1.64)	1.000	1.000	P
2	660	(2.03,4.32)	0.139	0.218	P
3	430	(3.65,4.19)	0.276	0.359	P
4	8300	(3.65,3.96)	0.999	0.999	P
5	2400	(3.88,4.12)	0.749	0.805	P
6	600	(3.84,3.85)	0.379	0.473	A
7	480	(3.73,3.58)	0.379	0.483	P
8	200	(4.51,3.19)	0.436	0.517	P
9	46000	(4.56,2.94)	1.000	1.000	P
10	2500	(4.47,2.96)	0.904	0.927	P
11	3000	(3.96,2.34)	0.935	0.957	P
12	1000	(3.93,2.33)	0.662	0.760	P
13	1300	(3.79,2.07)	0.750	0.826	P
14	18000	(3.70,2.12)	1.000	1.000	P
15	9000	(3.46,1.89)	1.000	1.000	P
16	150	(3.02,2.73)	0.343	0.489	P
17	1900	(2.79,2.61)	0.748	0.841	P
18	9000	(2.64,2.26)	1.000	1.000	P
19	15200	(2.62,2.07)	1.000	1.000	P
20	300	(1.95,2.09)	0.284	0.421	A
21	12	(2.28,1.94)	0.292	0.432	P
22	40	(2.38,1.95)	0.322	0.461	P
23	100	(2.63,1.82)	0.376	0.519	P
24	20	(2.60,1.87)	0.357	0.497	P
25	800	(3.24,2.20)	0.604	0.731	P
26	200	(3.43,2.22)	0.472	0.612	P
27	1600	(3.47,2.28)	0.780	0.859	P
28	2500	(3.34,2.27)	0.894	0.935	P
29	15600	(3.11,1.61)	1.000	1.000	P
30	100	(3.34,1.48)	0.444	0.565	A
31	1375	(5.06,1.03)	0.468	0.575	P
32	100	(4.96,1.07)	0.214	0.302	A
33	5000	(4.77,1.23)	0.977	0.984	P
34	3000	(4.83,1.08)	0.845	0.889	P
35	300	(2.94,0.99)	0.342	0.476	P
36	1500	(2.23,1.31)	0.599	0.737	A
37	1200	(2.41,0.90)	0.479	0.638	P
38	4000	(2.06,0.80)	0.921	0.956	P
39	750	(2.22,0.96)	0.361	0.527	A
40	225	(2.13,0.90)	0.239	0.383	A
41	3250	(1.85,1.16)	0.864	0.921	P
42	12	(1.70,1.17)	0.183	0.309	A
43	450	(1.76,1.26)	0.277	0.426	P
44	8000	(1.92,1.39)	0.999	0.999	P
45	1050	(1.12,0.87)	0.261	0.404	P
46	4800	(0.76,0.41)	0.889	0.936	P
47	100	(3.75,2.25)	0.461	0.587	P
48	1800	(4.22,2.71)	0.806	0.861	P
49	400	(3.80,2.28)	0.530	0.650	P
50	1000	(2.46,0.78)	0.411	0.573	P

Table 5.2: The metapopulation of *Melitaea cinxia* investigated in lower and upper bounds for the incidence calculated by the method described in Section 3.2.

[a]Status of the patch when investigated (Presence/Absence).

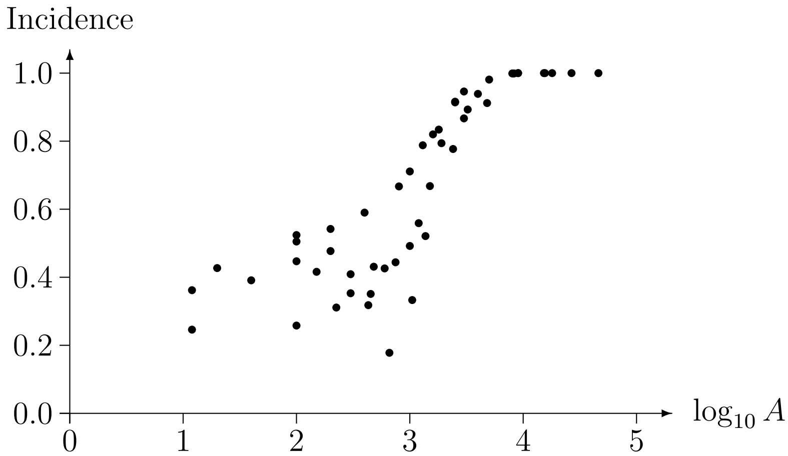

Figure 5.5: Plot of the predicted incidence of occupancy versus patch area when the metapopulation model is used on the data from [7].

Appendix A

Program that Calculates J_i and $p(k)$

This appendix contains Matlab `.m`-files to calculate the stationary probabilities discussed in section 2.1. The first file provided is one that calculates the interaction matrix Q from the model in Chapter 5 given the model parameters a, b and c, the patch areas and also the coordinates of the patches in a metapopulation.

```
function Q=qbymodel(a,b,c,A,pos)
%The function qbymodel.m calculates the probabilities that
%patches do not colonize each other according to the model
%described in Chapter 5.
%
%Function call: Q=qbymodel(a,b,c,A,pos)
%
%Inputs: a    Measure of how bad individuals are at migrating
%             long distances.
%        b    Measure of the carrying capacity per area unit.
%        c    Measure of how good individuals are at surviving.
%        A    Vector containing the area of the patches.
%        pos  A (2 x n) matrix with each column describing the
%             position of a patch.
%
%Output: Q    The calculated interaction matrix for the system.

n=length(A);

%-------------------------------------------------------------
% Calculate distance between patches
```

```
%-------------------------------------------------------------
tmp1=ones(n,1)*pos(1,:);
tmp2=pos(1,:)'*ones(1,n);
tmp3=ones(n,1)*pos(2,:);
tmp4=pos(2,:)'*ones(1,n);
d=sqrt((tmp1-tmp2).^2+(tmp3-tmp4).^2);

%-------------------------------------------------------------
% Now calculate the matrix Q from areas and distances
%-------------------------------------------------------------
Q=exp(-b*exp(-a*d).*(ones(n,1)*A)');
Q=Q-diag(diag(Q,0));
Q=Q+diag(exp(-c*A));

%-------------------------------------------------------------
% Patch no 1 is mainland
%-------------------------------------------------------------
Q(1,1)=0;
```

The following .m-file calculates the stationary probabilities given an interaction matrix Q, from the program above.

```
function [p,P,w,J]=metapop(Q)
%The function metapop.m calculates the stationary probabilities
%of the metapopulation model used in this text.
%
%Function call: [p,P,w,J]=metapop(Q)
%
%Input:  Q    The interaction matrix where Q(i,j) is the
%             probability that patch i will not colonize
%             patch j in one time step.
%
%Outputs:p    Row vector describing the stationary distribution
%             of the number of occupied patches.
%        P    Transition matrix for the Markov process.
%        w    Stationary probabilities for all states.
%        J    Incidence for patches in the metapopulation.

n=length(Q);

%-------------------------------------------------------------
```

```
% Construct the matrix bits, which have 2^(n-1) columns
% each describing one state.
%-------------------------------------------------------------
oneline=ones(1,2^(n-1));
D1=[2^(n-1):2^n-1];
bits=ones(n,1)*D1;
weights=2 .^[n-1:-1:0]'*oneline;
bits=1-(floor(bits./weights)/2 == floor(bits./weights/2));

%-------------------------------------------------------------
% Calculate the probabilities from equation (2.1).
%-------------------------------------------------------------
q=ones(n,2^(n-1));
for i=1:n,
    q(i,:)=prod((Q(:,i)*oneline).^bits);
end;

%-------------------------------------------------------------
% Now calculate the state transition matrix P.
%-------------------------------------------------------------
P=ones(2^(n-1),2^(n-1));
for x=1:2^(n-1),
    P(x,:)=
    prod((q(:,x)*oneline).^(1-bits).*((1-q(:,x))*oneline).^bits);
end;

%-------------------------------------------------------------
% Find the eigenvector w that has eigenvalue 1.
%-------------------------------------------------------------
[V,D]=eig(P');
I=find(sum(D)>0.999999);
if length(I)>1,
    V,sum(D),P,Q
    I=input('Which one is 1?');
end;
w=V(:,I)/sum(V(:,I));

%-------------------------------------------------------------
% Now calculate the incidence and stationary distribution
% of the number of occupied patches.
%-------------------------------------------------------------
```

```
p=sum(w*ones(1,n).*((ones(n,1)*sum(bits))'==oneline'*[1:n]));
w=w';
J=[1:n];
for i=1:n,
    J(i)=sum(w(find(bits(i,:)==1)));
end;
```

Appendix B

Programs that Estimate J_i and $p(k)$

In this appendix we provide Matlab `.m`-files to calculate estimates as described in Chapter 3.

B.1 Calculation with Lumped Metapopulation

The method described in Section 3.1 is implemented in the following program. The program does not require the lumping conditions to hold, but instead calculates the geometric mean of all values that are required to be equal by the method.

```
function [p,P]=lumppop(Q,G)
%The function lumppop.m calculates an approximation of the
%result given by metapop.m, given that patches are lumped as
%described in Section 3.1.
%
%Function call: [p,P]=lumppop(Q,G)
%
%Inputs: Q    The interaction matrix where Q(i,j) is the
%             probability that patch i will not colonize
%             patch j in one time step.
%       G    Partitioning of patches.
%             Each row in this matrix describes one lump. The
%             mainland (patch no 1) must not occur in any lump!
%
%Outputs:p    Row vector describing the stationary distribution
```

```
%              of the number of occupied patches.
%         P    Transition matrix for the Markov process.
%
% This function does not check for Q to satisfy the conditions
% reqiured to be able to lump patches.  Instead it calculates
% the geometric mean of all probabilities that are assumed to
% be equal.

n=length(Q);

%-------------------------------------------------------------
% First make sure that G contains all patches.  That is,
% add lumps with only one member if some patches are not
% present in G.
%-------------------------------------------------------------
[ng,l]=size(G);
if l==0, l=1; end;
for i=[n:-1:1],
    if length(find(G==i))==0,
        G=[i zeros(1, l-1);G];
    end;
end;

%-------------------------------------------------------------
% Calculate the number of lumps, and the number of patches
% in each.
%-------------------------------------------------------------
[ng,l]=size(G);
np=[sum([G zeros(ng,1)]' ~= 0) 1];

%-------------------------------------------------------------
% Produce all states for the lumped Markov chain.
%-------------------------------------------------------------
state=[1 zeros(1, ng)];
while state(ng+1)==0,
    s=[s;state([1:ng])];
    i=2;
    while state(i)==np(i),
        state(i)=0;
        i=i+1;
    end;
```

```
    state(i)=state(i)+1;
end;
[ns,tmp]=size(s);

%-------------------------------------------------------------
% Calculate the geometric mean for all entries that should
% be equal in Q.
%-------------------------------------------------------------
for i=1:ng,
    g=G(i,find(G(i,:)~=0));

    t=diag(Q); t=prod(t(g))^(1/np(i));
    Q(g(1),g(1))=t;

    if np(i) > 1,
        r=Q(g,g); r=r-diag(diag(r))+diag(ones(1,np(i)));
        r=prod(prod(r))^(1/(np(i)^2-np(i)));
        Q(g(1),g(2))=r;
    end;

    for j=[1:i-1 i+1:ng],
        Q(G(j,1),g(1))=
        prod(prod(Q(G(j,[1:np(j)]),g)))^(1/np(i)/np(j));
    end;
end;

%-------------------------------------------------------------
% Calculate the state transition probabilities.
%-------------------------------------------------------------
P=ones(ns,ns);
for x=1:ns,
    q=ones(ng,max(np)+1);
    for i=1:ng,
        g=G(i,find(G(i,:)~=0));
        gl=length(g);
        for yi=0:np(i),
            t=Q(g(1),g(1));
            if gl > 1,
                r=Q(g(1),g(2));
            else
                r=1;
```

```
            end;

            o=prod(Q(G([1:i-1 i+1:ng],1),g(1))'.^
              s(x,[1:i-1 i+1:ng]));

            j=[max([0 s(x,i)+yi-gl]):min([s(x,i) yi])];

            over1=over(s(x,i),j);
            over2=over(gl-s(x,i),yi-j);

            prob=over1.*(1-t*r^(s(x,i)-1)*o).^j.*
              (t*r^(s(x,i)-1)*o).^(s(x,i)-j).*
              over2.*(1-r^s(x,i)*o).^(yi-j).*
              (r^s(x,i)*o).^(gl-yi-s(x,i)+j);
              q(i,yi+1)=sum(prob);
        end;
    end;
    for y=1:ns,
        P(x,y)=prod(q(s(y,:)*ng+[1:ng]));
    end;
end;

%-------------------------------------------------------------
% At last, calculate the Perron-Frobenius eigenvector and
% the distribution of the number of occupied patches.
%-------------------------------------------------------------
[V,D]=eig(P');
[M,I]=max(sum(D));
out=V(:,I)/sum(V(:,I));

p=[1:n];
for i=1:n,
    p(i)=sum(out(find(sum(s')==i)));
end;
```

The program above requires requires the following piece of code to work.

```
function n=over(a,b)
%Function call: n=over(a,b)
%
%This function takes an integer and an integer vector and produces
```

```
%an integer vector where elements are:
%
%                  a!
%           n(i)=---------------
%                b(i)! (a-b(i))!
n=b;
for i=1:length(b),
    n(i)=prod([1 2:a])/prod([1 2:b(i)])/prod([1 2:a-b(i)]);
end;
```

B.2 Upper and Lower Bounds for the Incidence

The method for obtaining upper and lower bounds for the incidence of occupancy described in Section 3.2 is implemented in the following two .m-files. The first calculates a lower bound for the incidence of occupancy for a specific patch, and the second calculates an upper bound and works much in the same way.

```
function mn=minjc(Q,m,c)
%The function minjc.m calculates a lower limit for the
%incidence of occupancy for patch number c given the
%interaction matrix Q by the method described in Section 3.2.
%
%Function call: function mn=minjc(Q,m,c)
%
%Inputs: Q    The interaction matrix where Q(i,j) is the
%             probability that patch i will not colonize
%             patch j in one time step.
%       m    The number of patches for which the condition that
%             they can not become occupied are removed.
%       c    The patch for which the calculations should be done.
%
%Output: mn   Lower limit for the incidence for patch number c.

[n, tmp]=size(Q);

%------------------------------------------------------------
% The sets(vectors) incl and excl keep track of which
% patches are considered.  Patch number c will always be
% number 2 in the new constructed metapopulation.  We keep
% the best limit so far in mn.
```

```
%-------------------------------------------------------------
incl=[1 c];
excl=[2:c-1 c+1:n];
[p,P,w,J] = metapop(Q(incl,incl));
mn=J(2);

%-------------------------------------------------------------
% Add one more patch at a time to the system.
%-------------------------------------------------------------
for i=1:m,
    %---------------------------------------------------
    % Try which of the not yet included patches gives
    % the best result.
    %---------------------------------------------------
    bsf_ix = 1;
    for j=1:length(excl),
        pop=[incl excl(j)];
        [p,P,w,J] = metapop(Q(pop,pop));
        if J(2)>mn,
            mn=J(2);
            bsf_ix = j;
        end;
    end;
    incl=[incl excl(bsf_ix)];
    excl=excl([1:bsf_ix-1 bsf_ix+1:length(excl)]);
end;

function mx=maxjc(Q,m,c)
%The function maxjc.m calculates an upper limit for the incidence
%of occupancy for patch number c given the interaction matrix Q by
%the method described in Section 3.2.
%
%Function call: function mx=maxjc(Q,m,c)
%
%Inputs: Q    The interaction matrix where Q(i,j) is the
%             probability that patch i will not colonize
%             patch j in one time step.
%        m    The number of patches for which the condition that
%             they can not become empty are removed.
%        c    The patch for which the calculations should be done.
%
```

```
%Output: mx   Upper limit for the incidence for patch number c.

[n, tmp]=size(Q);

%--------------------------------------------------------------
% The sets(vectors) incl and excl keep track of which
% patches are considered.  Patch number c will always be
% number 2 in the new constructed metapopulation.  We keep
% the best limit so far in mx.
% To remove the patches that are always inhabited from the
% system, we recalculate the probability that migrators from
% the mainland occupy the other patches.
%--------------------------------------------------------------
incl=[1 c];
excl=[2:c-1 c+1:n];

Q1=Q(incl,incl);
Q1(1,:)=prod(Q([1 excl],incl));

[p,P,w,J] = metapop(Q1);
mx=J(2);

%--------------------------------------------------------------
% Add one more patch at a time to the system.
%--------------------------------------------------------------
for i=1:m,
    %------------------------------------------------------
    % Try which of the not yet included patches gives
    % the best result.
    %------------------------------------------------------
    bsf_ix = 1;
    for j=1:length(excl),
        pop=[incl excl(j)];

        %----------------------------------------------
        % Recalculate Q(mainland,:) for each system.
        %----------------------------------------------
        Q1=Q(pop,pop);
        Q1(1,:)=prod(Q([1 excl([1:j-1 j+1:length(excl)])],pop));

        [p,P,w,J] = metapop(Q1);
```

```
        if J(2)<mx,
            mx=J(2);
            bsf_ix = j;
        end;
    end;
    incl=[incl excl(bsf_ix)];
    excl=excl([1:bsf_ix-1 bsf_ix+1:length(excl)]);
end;
```

Appendix C

Simulation Program

This is a simple implementation of the simulation method described in Section 4.1. In contrast to the programs in the preceding appendices, this program is written in C to allow for much faster computations. The program simply reads input data from the standard input and writes it's result back to the standard output. When used for a specific purpose, as for generating Figure 5.3, the output part of the program could easily be modified to produce output of whatever form is needed.

```
#include <stdlib.h>
#include <math.h>

#define MAXDIM 200 /* Maximum number of patches */

int n;
double q[MAXDIM][MAXDIM];        /* The interaction matrix Q */
int p1[MAXDIM] = {1};            /* Initial state */
long dist[MAXDIM], inc[MAXDIM];  /* Cumulative number of occupied
                                 /* patches and number of times
                                 /* each patch was visited */

/* This is an implementation of a fibbonacci random number
   generator with lags 33 and 97. */

double u[100];
void initcrand(void) {
    int i;

    for (i = 0; i < 100; i++)
```

```
        u[i] = (double)(rand()%1000) / 1000;
}

double crand(void) {
    static i = 97, j = 33;
    double res;

    if ((res = u[i]-u[j]) < 0)
        res++;
    u[i]=res;
    if (--i == 0) i = 97;
    if (--j == 0) j = 97;

    return res;
}

/* This function takes care of one time step of disperal
   between patches. It changes the current state (p1). */

void makemove(void) {
    int i, j;
    int p2[MAXDIM];

    for (j = 0; j < n; j++) {
        p2[j]=0;
        for (i = 0; !p2[j] && i < n; i++)
            if (p1[i] && crand() >= q[i][j])
                p2[j] = 1;
    }
    for (j = 0; j < n; j++)
        p1[j]=p2[j];
}

/* This function simulates length time steps and incremants dist
   and inc accordingly. */

void simu(long length) {
    long i, j, k;

    for (i=0; i<length; i++) {
        makemove();
```

```
        for (k = j = 0; j < n; k += p1[j++])
            inc[j] += p1[j];
        dist[k-1]++;
    }
}

/* Main funtion of the program.  Reads the number of patches
   (dimension of the matrix Q) and the matrix Q itself, row
   by row, from stdin.
   Prints periodical output of estimates on stdout. */

void main(void) {
    long i, j, k, l = 0;
    double nn;

    initcrand(); /* Initialize random number generator */

    /* Read the input data */
    scanf("%lf",&nn); n = nn;
    for (i = 0; i < n; i++)
        for (j = 0; j < n; j++)
            scanf("%lf", &q[i][j]);

    /* Make 5*200000 simulation steps, and produce output every
       200000 steps.  Subject to modification! */

    for (k = 0; k <  5; k++) {
        simu(200000l); l += 200000l;

        /* Print current estimates */
        printf("Number of steps: %ld\n", l);
        printf("i\tJ(i)\tp(i)\n");
        printf("---------------------\n");
        for (i = 0; i < n; i++)
            printf("%d\t%5.3lf\t%5.3lf\n",
                i+1, (double)inc[i]/l, (double)dist[i]/l);
        printf("\n");
    }
}
```

Bibliography

[1] Diamond, J. M. (1978), *Critical Areas for Maintaining Viable Populations of Species,* chapter from book, "The Breakdown and Restoration of Ecosystems," eds. Holdgate, M. W. and Woodman, M. J. Plenum Press, New York.

[2] Feller, W. (1968), *An Introduction to Probability Theory and its Applications, vol. I,* John Wiley & Sons Inc., New York-London-Sydney.

[3] Gantmacher, F. R. (1960), *The Theory of Matrices, vol. I,* Chelsea, ISBN 0-8284-0131-4.

[4] Gilpin, M. E. and Diamond, J. M. (1981), *Immigration and extinction probabilities for individual species: Relation to incidence functions and species colonization curves,* Ecology, vol. 78, no. 1, pp. 392-396.

[5] Gyllenberg, M. and Silvestrov, D. S. (1995), *Quasi-stationary distributions of a stochastic metapopulation model,* in press Journal of Mathematical Biology.

[6] Hanski, I. (1994), *A practical model of metapopulation dynamics,* in press Journal of Animal Ecology.

[7] Hanski, I., Kuussaari, M. and Nieminen, M. (1994), *Metapopulation structure and migration in the butterfly Melitaea cinxia,* in press Ecology.

[8] Knuth, D. E. (1969), *The Art of Computer Programming, vol. 2: Seminumerical Algorithms,* Addison-Wesley, ISBN 0-201-03822-6.

[9] Seneta, E. (1973), *Non-negative Matrices and Markov Chains,* Springer-Verlag New York, ISBN 0-387-90598-7, 3-540-90598-7.