Bayesian parameter inference for Individual-Based Models using Particle Markov chain Monte Carlo (PMCMC)

Mira Kattwinkel*[1,2], Peter Reichert[1]

[1] Eawag: Swiss Federal Institute of Aquatic Science and Technology, Department of Systems Analysis, Integrated Assessment and Modelling, Überlandstrasse 133, 8600 Dübendorf, Switzerland

[2] Current address: University Koblenz-Landau, Quantitative Landscape Ecology, Fortstraße 7, 76829 Landau, Germany

* corresponding author: mira.kattwinkel@gmx.net

This document is the accepted manuscript version of the following article:
Kattwinkel, M., & Reichert, P. (2017). Bayesian parameter inference for individual-based models using a Particle Markov Chain Monte Carlo method. Environmental Modelling and Software, 87, 110-119. https://doi.org/10.1016/j.envsoft.2016.11.001

Highlights

- A procedure for Bayesian inference for agent-/individual-based models (AMBs/IBMs)
- Implementation based on Particle Markov Chain Monte Carlo (PMCMC)
- Good performance and insights into model behaviour for simple IBM with virtual data
- Bayesian inference for AMBs/IBMs promotes advances in ecological research

Abstract

Parameter estimation for agent-based and individual-based models (ABMs/IBMs) is often performed by manual tuning and model uncertainty assessment is often ignored. Bayesian inference can jointly address these issues. However, due to high computational requirements of these models and technical difficulties in applying Bayesian inference to stochastic models, the exploration of its application to ABMs/IBMs has just started. We demonstrate the feasibility of Bayesian inference for ABMs/IBMs with a Particle Markov Chain Monte Carlo (PMCMC) algorithm developed for state-space models. The algorithm profits from the model's hidden Markov structure by jointly estimating the system state and the marginal likelihood of the parameters using time-series observations. The PMCMC algorithm performed well when tested on a simple predator-prey IBM using artificial observation data. Hence, it offers the possibility for Bayesian inference for ABMs/IBMs. This yields additional insights into model behaviour and uncertainty and extends the usefulness of ABMs/IBMs in ecological and environmental research.

Keywords

parameter estimation; calibration; agent-based model (ABM); individual-based model (IBM); Particle Markov Chain Monte Carlo (PMCMC); approximate Bayesian computation (ABC)

Software availability

The IBM and the particle filtering were coded in Java using jdk 1.8, the MCMC algorithm was implemented in R 3.1. The Java Development Kit is freely available for various operating systems at http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html; R is freely available for various operating systems at https://cran.r-project.org/.

All programming was performed by the authors if not stated otherwise and the code can be found in the SI.

1. Introduction

Agent-based or individual-based models (ABMs/IBMs; used synonymously here) were introduced to consider the finite number of and variability among the entities modelled (e.g. animals, plants, farmers, households), their individual behaviour, and the local nature of their interactions in ecological, environmental, and socio-economic systems (Huston et al., 1988; Grimm, 1999; Brown, D.G. et al., 2004). IBMs describe all individuals in a population as individual entities and consider the stochastic nature of processes such as birth and death for which we can only estimate a probability. Additionally, in contrast to other stochastic models (e.g. stochastic population models), IBMs explicitly incorporate individual characteristics (e.g. mass, life stage, phenotype) that differ among the individuals, they enable consideration of the spatio-temporal arrangement of organisms, and interactions among individuals (DeAngelis and Mooij, 2005). Hence, they contribute to the understanding of how system dynamics emerge from the variability and behaviour of individual organisms (Grimm and Railsback, 2005).

Typically, IBMs are more complex than other, in particular analytically tractable model types (Grimm and Railsback, 2005). They usually require Monte Carlo simulation even for forward modelling and common issues are longer runtimes and computer memory demands of IBMs compared to other model types. Hence, both model description and analysis may be impeded. In particular, when modelling is used for prediction, in research as well as in real world decision support, the assessment of model uncertainty is a very important task (Refsgaard et al., 2007; Reichert et al., 2015). Recently, much progress has been made in how to systematically describe and document IBMs, which facilitates their reproducibility and testing (Grimm et al., 2006; Grimm et al., 2010; Grimm et al., 2014). Despite the complex structure of these models, propagation of parameter uncertainty and of stochasticity of the modelled processes to model results is conceptually simple and numerically easy to implement by means of Monte Carlo simulation. This is, however, not the case for parameter inference (i.e. model calibration). Thiele et al. (2014) give an extensive overview of the possible approaches to analyse and calibrate IBMs, including practical examples. Additionally, pattern oriented modelling (POM, Grimm et al., 2005) has been suggested for constructing and calibrating IBMs (Topping et al., 2012) and other model types. Nevertheless, parameter estimation is often still performed in an ad-hoc manner, regularly relying on manual tuning, and assessment of model output uncertainty is frequently ignored.

Bayesian inference offers a consistent framework for updating the current state of scientific knowledge about a system (model parameters and potentially also model structure) based on observed data (Ellison 2004; Gelman et al., 2014). As, in a Bayesian context, knowledge is described by probability distributions, this procedure intrinsically contains an assessment of uncertainty. Formally, Bayesian inference is based on the formulation of the model as a probability distribution of model outcomes for

given parameter values, the so-called likelihood function of the model, and another probability distribution describing prior information about parameter values. The posterior distribution of model parameters combines this prior knowledge on model structure and parameters with information from observed data corresponding to model outcomes. It is calculated as the normalized product of the prior distribution of the parameters and the likelihood function of the model with actual data substituted for the model outcomes.

Except for very simple cases, Bayesian inference cannot be done analytically. For this reason, numerical schemes have been developed to sample from the posterior distribution of model parameters. Properties of the posterior distribution, such as its mean, variance-covariance structure, and marginal distributions of individual parameters, can then be derived from this sample. Commonly applied techniques to sample from a posterior for Bayesian inference are Metropolis or Metropolis-Hastings Markov chain Monte Carlo (MCMC) sampling schemes (Gamerman and Lopes, 2006; Gelman et al., 2014; Hastings, 1970; Metropolis et al., 1953). These schemes are often easy to apply as they directly use the product of the prior times the likelihood function and do not require normalization of this product. However, due to the complexity of IBMs that typically contain various stochastic processes, it is practically unfeasible to evaluate their complete likelihood functions. For instance, because repeated IBM runs for a given parameter set will yield different results to a greater or lesser extent one cannot just run the IBM once and calculate the likelihood based on these results.

There are two general approaches to deal with analytically intractable likelihoods: Approximate Bayesian Computation (ABC) techniques sample from the posterior by using approximate conditioning of simulations on the data (Albert et al., 2014; Beaumont, 2010; Csillery et al., 2010; Hartig et al., 2011; Marjoram et al., 2003). In addition, the dimensionality of the data is usually reduced by using summary statistics, ideally constructed with a minimum loss of information (Blum et al., 2013; Fearnhead and Prangle, 2012). Alternatively, the likelihood function can be approximated numerically, e.g. by fitting a parametric distribution to an output sample simulated from the likelihood, and then be used in a standard MCMC scheme (Hartig et al., 2014). While these general approaches are applicable to stochastic models of any structure, another option for approximating the likelihood may be beneficial that benefits from a specific model structure. Most IBMs have a “hidden Markov structure”. A model has a Markov structure if the future model states do not depend on the past other than through the current state. If the states are observed incompletely, the observations do not follow a Markov structure, and the underlying Markov structure of the states is hidden (or latent). This is typically the case in IBMs with observations consisting of time-series data (e.g. of abundances). Figure 1 illustrates this hidden Markov structure of a typical IBM consisting of individuals of a predator species and a prey

species at different life stages and with different masses for which observations are only available for the abundances of individuals exceeding a certain size (proportional to the weight).

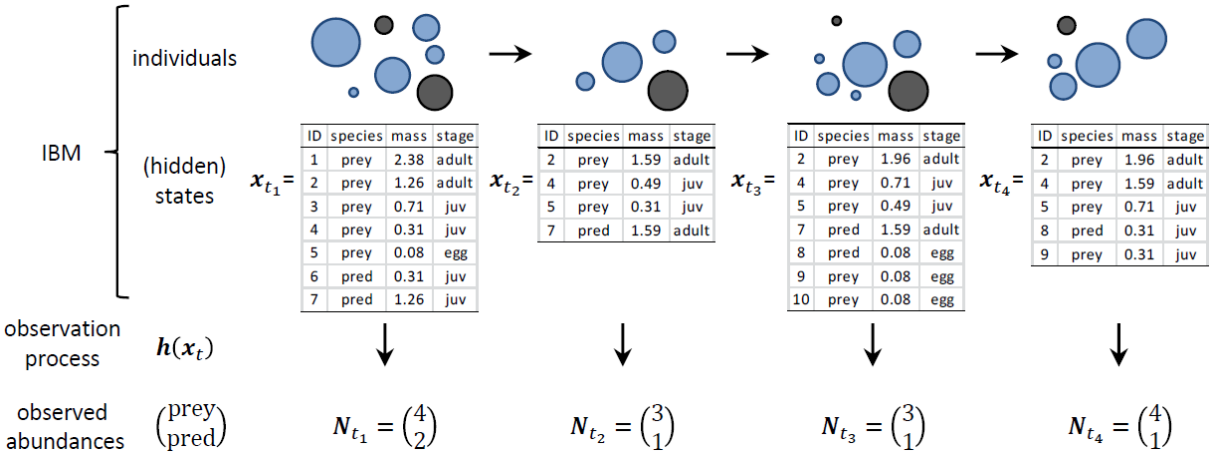


Figure 1: Illustration of a two-species IBM with individuals of either prey or predator species at different life stages and with different masses as a state-space or hidden Markov model. Circles depict individuals in the individual-based simulation (preys are represented in blue, predators in grey), the states $\boldsymbol{x}_t$ are represented by all properties of the individuals necessary for the evolution to the next time step (only examples of the properties are given here), the observation process $\boldsymbol{h}(\boldsymbol{x}_t)$ filters for individuals large enough to be detected, and the vector $\boldsymbol{N}(t)$ consists of the observed abundances of the two species in the model. Note that the observation process is incomplete by not identifying eggs, not distinguishing juveniles (juv) and adults, and not resolving masses of the individuals.

Particle Markov Chain Monte Carlo (PMCMC) methods (an MCMC technique including an internal particle filtering or Sequential Monte Carlo (SMC) step for the state space) have been developed to estimate states and parameters of state-space or hidden Markov models. Such algorithms jointly estimate the system state and the marginal likelihood of the parameters by particle filtering based on the observations. Subsequently, they use this marginal likelihood in a Markov chain with a Metropolis acceptance/rejection step to estimate the model parameters. Particle filtering consists of propagating a sample of states (particles) through the state-space model (i.e. replicate model runs of the stochastic model) and re-sampling them iteratively at each time step based on importance weights to condition them on the observations. This class of algorithms was developed by Beaumont (2003), Andrieu and Roberts (2009), and Andrieu et al. (2010). Kantas et al. (2015) provide a comprehensive review on particle methods for parameter estimation in state-space models. These techniques have been applied successfully to state-space models in many application areas (e.g. Flury and Shephard, 2011; Golightly and Henderson, 2014), but have not yet reached attention in the ABM/IBM ecological modelling community.

Compared to ABC, we expect a gain in efficiency for PMCMC due to the re-sampling of the particles at each time step and thus profiting from the known model structure by guiding the solutions for the states (for given parameter values) by the observations. Owen et al. (2015) demonstrated that this may lead to an efficiency gain in particular for long time series. On the other hand, the number of particles has to be chosen large enough to avoid a strong bias and sufficiently small to guarantee efficiency.

So far there has been insufficient experience with Bayesian inference for IBMs to provide recommendations on efficient numerical algorithms, and the scientific community is currently at a stage of evaluating the feasibility of selected numerical schemes (Hartig et al., 2014; Lagarrigues et al., 2015; van der Vaart et al., 2015). In this study, we present a PMCMC algorithm as described above and its application to an ecological IBM. We aim at contributing to the evaluation of different numerical schemes for parameter estimation of individual-based stochastic models by testing the feasibility of this Bayesian inference algorithm using a two species IBM and artificial observation data.

2 Material and Methods

First, we describe the combination of the IBM with the observation model as a hidden Markov model (Figure 1) resulting in the likelihood function of the model. Then, we briefly introduce the concept of Bayesian inference and give a step by step description of the numerical implementation scheme of the PMCMC algorithm. Finally, we describe the application used to test the algorithm.

2.1 Model Description

The model consists of an individual-based, stochastic model of the dynamics of a simple predator-prey system and an observation model that describes the random observation errors of the observable abundances of the two species in the community. The observation model is needed to describe which features of the states are observable (here: abundances of sufficiently large individuals) and to consider the observation error of the abundances (or other state variables; note that observations of abundances in ecological systems are often highly uncertain depending on factors including the size of the animals, patchiness of the habitat or opportunities for hiding). Combining both models leads to the overall likelihood function.

2.1.1 Individual-Based Community Model

We use a model of a simple aquatic predator-prey system to test and exemplify the application of the algorithm for Bayesian inference with an IBM. The model is inspired by the differential equation based dynamic food web model Streambugs (Schuwirth and Reichert, 2013; Schuwirth et al., 2015) and by its

recent application to a mesocosm experiment with aquatic macroinvertebrates (Kattwinkel et al., 2016). In the IBM, the prey feeds on an inexhaustible food source while the predator exclusively feeds on the prey. The model is based on the metabolic theory of ecology (Brown, J.H. et al., 2004; Sibly et al., 2013); the basal metabolic rate ($r_{\text{basal }i}$) of all individuals, i, scales with body mass and temperature. Additional individual variability in this rate is described by the parameter $sd_{\text{f basal}}$. The vital rates (i.e. respiration, mortality and ingestion) are calculated from $r_{\text{basal }i}$ by multiplication with the factors f_{resp} and $f_{\text{resp egg}}$ (respiration; for hatched individuals and eggs, respectively), f_{mort} and $f_{\text{mort egg}}$ (mortality), and $f_{\text{ingest prey}}$ and $f_{\text{ingest pred}}$ (ingestion; species specific for the prey and predator species, respectively). Additionally, the ingestion rate is modified by the available food density and by a density-dependent self-inhibition factor, which depends on the parameters $K_{\text{dens prey}}$ and $K_{\text{dens pred}}$, respectively. Ingested food is used for maintenance (respiration), somatic growth and, for adults, accumulating reproductive mass. Egg laying takes place if sufficient reproductive mass has been accumulated, if it is the reproductive season of the species, and if sufficient time since the last reproduction event has passed. A detailed model description based on the ODD (Overview, Design concepts, and Details) protocol (Grimm et al., 2006; Grimm et al., 2010) is given in the SI.

The current state of the system at time t is described by a state vector, $\boldsymbol{x}_t$, which contains all relevant information to perform the next time step (Figure 1). In our example $\boldsymbol{x}_t$ consists of the species types, life stages, somatic and reproductive masses, elapsed times since the last egg laying event, the maximum masses attained so far, and individual specific parameters of all individuals in the community. Based on this information, the dynamics of the population is Markovian, i.e. the future of the system does not depend on information from the past other than that contained in $\boldsymbol{x}_t$. This means that we can formulate the time evolution of the community over discrete time intervals from t_{j-1} to t_j by an initial distribution and a stochastic, iterative time-evolution:

$$\begin{aligned} \boldsymbol{X}_{t_0} &\sim f_0\left(\boldsymbol{x}_{t_0} \mid \boldsymbol{\theta}\right) \\ \boldsymbol{X}_{t_j} &\sim f_j\left(\boldsymbol{x}_{t_j} \middle| \boldsymbol{x}_{t_{j-1}}, \boldsymbol{\theta}\right) \quad , \quad \text{for } j \geq 1 \quad . \end{aligned} \tag{1}$$

In these equations, $\boldsymbol{X}_t$ is the vector of random variables representing the probability distribution of the state, $\boldsymbol{x}_t$, $\boldsymbol{\theta}$ is the vector of model parameters (combining the parameters of the time evolution and the observation model (see below)), f_0 describes the initial distribution of the state vector, and f_j, $j \geq 1$ describes the evolution step from time t_{j-1} to time t_j according to the IBM. Hence, according to equation (1), the dynamics of the model is defined by the probability density of the initial state, f_0, and the conditional probability densities of succeeding states, given the previous state, f_j. Note that the conditional probability densities f_j extend over state vectors $\boldsymbol{x}_{t_j}$ of different lengths because individuals

may die and eggs may be laid (Figure 1). When formulated for a constant time step, all f_j for $j \geq 1$ would be the same. However, to allow for irregularly spaced time steps, we make f_j dependent on j.

Given the model equations (1), the joint probability density of all states is given as

$$f(\boldsymbol{x}_{t_0}, \ldots, \boldsymbol{x}_{t_n} | \boldsymbol{\theta}) = f_0(\boldsymbol{x}_{t_0} \mid \boldsymbol{\theta}) \prod_{j=1}^{n} f_j\left(\boldsymbol{x}_{t_j} \middle| \boldsymbol{x}_{t_{j-1}}, \boldsymbol{\theta}\right) \quad . \tag{2}$$

The formulation of the time stepping function, f_j ($j \geq 1$), in equation (1) is based on growth and respiration that affect the individual mass, on the survival probability given the death rate, on predation and on egg laying (keeping environmental influence factors constant over the time step). Several parameters in this relationship are formulated probabilistically. Due to the large and varying numbers of individuals, this leads to a probability density of the states given in equation (2), which is not easily tractable analytically but it is easy to draw samples from it, i.e. run simulations with the IBM.

2.1.2 Observation Model

For each observation time point, t, the observation model first extracts the vector of observable abundances of the modelled species, $\boldsymbol{N}_t$, from the comprehensive state vector, $\boldsymbol{x}_t$:

$$\boldsymbol{N}_t = \boldsymbol{h}(\boldsymbol{x}_t) \quad . \tag{3}$$

In our example, the function $\boldsymbol{h}$ accounts for the fact that no eggs and only individuals greater than a certain minimum mass are sampled in the observation process (e.g. with a mesh of a certain mesh size). Note that in contrast to the random variable $\boldsymbol{X}_t$, the length of which depends on the number of individuals in the community, the length of the vector $\boldsymbol{N}_t$ always equals the number of different species present in the model (in our case two) (Figure 1).

Propagation of the probability distribution of the time series of states $\boldsymbol{X}_{t_0}$, …, $\boldsymbol{X}_{t_n}$ (2) induces a probability distribution of the observable abundances at n time points:

$$P_{\mathrm{IBM}}(\boldsymbol{N}_{t_1}, \ldots, \boldsymbol{N}_{t_n} | \boldsymbol{\theta}) = \int\limits_{h(\boldsymbol{x}_{t_1})=\boldsymbol{N}_{t_1}, \ldots, h(\boldsymbol{x}_{t_n})=\boldsymbol{N}_{t_n}} f(\boldsymbol{x}_{t_0}, \ldots, \boldsymbol{x}_{t_n} | \boldsymbol{\theta})\, d\boldsymbol{x}_{t_0} \cdots d\boldsymbol{x}_{t_n} \quad . \tag{4}$$

This distribution represents our knowledge of the observable abundances as estimated by the model. Note that we allow for an initialization time of the model and thus start predicting observations at time t_1 rather than at t_0.

As the observation process induces additional uncertainty due to errors related to sampling, identifying and counting organisms, we need a probabilistic model that describes the observation uncertainty for true abundances given. Following Kattwinkel et al. (2016) and references cited therein, we use a negative binomial distribution to describe this process for aquatic invertebrates. Due to patchiness of

individuals observations of abundances are often overdispersed (i.e. the variance, σ^2, is larger than the mean, μ) compared to a Poisson distribution (with $\sigma^2 = \mu$) that would be adequate to describe counts of uniformly distributed individuals. We parameterize this negative binomial distribution using the parameter k, according to Elliot (1971):

$$k = \frac{\mu^2}{\sigma^2 - \mu} \quad . \tag{5}$$

The observation process is described by:

$$P_{\text{obs}}(\boldsymbol{N}_t^{\text{obs}} | \boldsymbol{N}_t, \boldsymbol{\theta}) = \prod_{i=1}^{m} \frac{\Gamma(N_{t,i}^{\text{obs}} + k)}{\Gamma(k) \cdot N_{t,i}^{\text{obs}}!} \cdot \left(\frac{k}{k + N_{t,i}}\right)^k \cdot \left(\frac{N_{t,i}}{N_{t,i} + k}\right)^{N_{t,i}^{\text{obs}}} \quad , \tag{6}$$

where μ is replaced by the modelled abundance of species i, $N_{t,i}$, m is the number of species, and k is an element of the overall parameter vector $\boldsymbol{\theta}$.

2.1.3 Marginal Likelihood Function

To predict observations, we have to combine the individual-based model (1) or (2) with the observation model given by the equations (3) and (4). This leads to the marginal likelihood function (i.e. integrated over the not directly observed states at the observation time points) for the observed abundances:

$$\begin{aligned} &P_{\text{marglikeli}}\left(\boldsymbol{N}_{t_1}^{\text{obs}}, \dots, \boldsymbol{N}_{t_n}^{\text{obs}} | \theta\right) \\ &= \int f_0(\boldsymbol{x}_{t_0} | \theta) \prod_{j=1}^{n} f_j\left(\boldsymbol{x}_{t_j} \middle| \boldsymbol{x}_{t_{j-1}}, \boldsymbol{\theta}\right) \prod_{j=1}^{n} P_{\text{obs}}\left(\boldsymbol{N}_{t_j}^{\text{obs}} \middle| \boldsymbol{h}(\boldsymbol{x}_{t_j}), \boldsymbol{\theta}\right) \mathrm{d}\boldsymbol{x}_{t_0} \dots \mathrm{d}\boldsymbol{x}_{t_n} \quad . \end{aligned} \tag{7}$$

As mentioned above, the joint density of the states is not easily tractable analytically. This is even more difficult for the marginal integral (7), which requires approximate treatment.

2.2 Bayesian Inference

2.2.1 Concept

The aim of Bayesian inference is to update prior knowledge about model parameters using observations (measurements) of some model output. To this end, we formulate the conditional probability distribution of the parameters given observations using the model likelihood function (section 2.1.3) and the prior probability distribution of model parameters.

Applied to the present context, the posterior probability density of model parameters, f_{post}, after learning from the observed abundances of species, $\boldsymbol{N}_{t_1}^{\text{obs}}, \dots, \boldsymbol{N}_{t_n}^{\text{obs}}$, is proportional to the product of

the marginal likelihood, $P_{\text{marglikeli}}$, with the actual observations substituted for $\boldsymbol{N}_{t_1}^{\text{obs}}, \dots, \boldsymbol{N}_{t_n}^{\text{obs}}$, and the prior probability density of the parameters, f_{pri}:

$$f_{\text{post}}\left(\boldsymbol{\theta}\middle|\boldsymbol{N}_{t_1}^{\text{obs}}, \dots, \boldsymbol{N}_{t_n}^{\text{obs}}\right) \propto P_{\text{marglikeli}}\left(\boldsymbol{N}_{t_1}^{\text{obs}}, \dots, \boldsymbol{N}_{t_n}^{\text{obs}}|\boldsymbol{\theta}\right) f_{\text{pri}}(\boldsymbol{\theta}) \quad . \tag{8}$$

This equation is very difficult to evaluate due to the complicated structure of $P_{\text{marglikeli}}$ (see equation (7)). For this reason, we have to rely on numerical techniques to derive properties of f_{post} from a sample drawn approximately from this distribution.

2.2.2 Numerical Implementation

The observable abundances, $\boldsymbol{N}_t$, are functions of the states, $\boldsymbol{N}_t = \boldsymbol{h}(\boldsymbol{x}_t)$, which are observed with an observation error (equations (3) and (6)**Error! Reference source not found.**). As these abundances are not a complete description of the states, the model does not have a Markov structure for $\boldsymbol{N}_t$. To profit from the hidden Markov structure of the model, we chose a Particle Markov Chain Monte Carlo algorithm (PMCMC) for estimating the states and approximating the marginal likelihood by particle filtering, and estimating the parameters with a Metropolis sampler based on the approximated marginal likelihood. This algorithm was developed by Beaumont (2003), Andrieu and Roberts (2009), and Andrieu et al. (2010) for state-space models (with constant lengths of the states, $\boldsymbol{x}_{t_j}$).

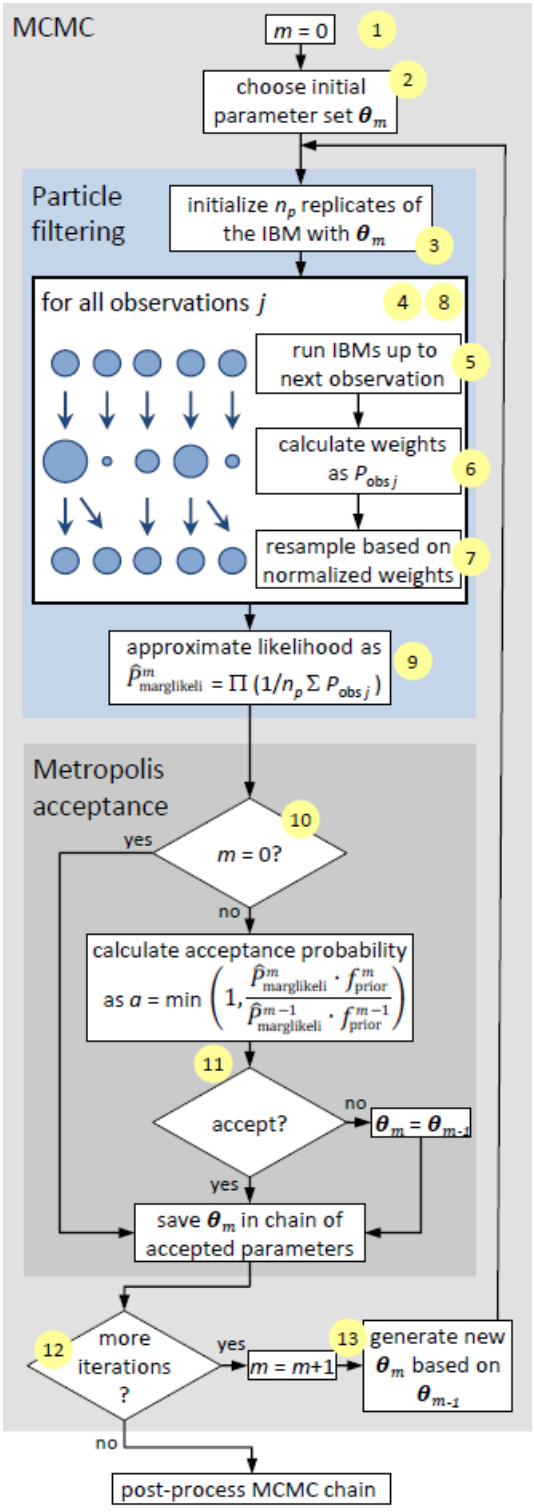


Figure 2: Flowchart of the PMCMC algorithm combining MCMC for model parameters with particle filtering for states. Numbers refer to the steps in the scheme given in the text. Blue dots visualize the particle filtering.

The PMCMC algorithm is based on the steps (Figure 2):

1. Set $m = 0$ (index for Markov chain sample of parameters).
2. Choose initial parameter values, $\boldsymbol{\theta}_0$.

3. Sample n_p replicates (particles), $\boldsymbol{x}_{t_0}^1, \ldots, \boldsymbol{x}_{t_0}^{n_p}$ from $f_0(\boldsymbol{x}_{t_0}|\boldsymbol{\theta}_m)$.
4. Set $j = 1$ (index for time points).
5. For all replicates, l=1,…,n_p, sample $\boldsymbol{x}_{t_j}^l$ from $f_j\left(\boldsymbol{x}_{t_j}\middle|\boldsymbol{X}_{t_{j-1}} = \boldsymbol{x}_{t_{j-1}}^l, \boldsymbol{\theta}_m\right)$.
6. For all replicates, l=1,…,n_p, calculate $w_j^l = P_{\text{obs}}\left(\boldsymbol{N}_{t_j}^{\text{obs}}\middle|\boldsymbol{h}(\boldsymbol{x}_{t_j}^l), \boldsymbol{\theta}_m\right)$ using the actually observed data for $\boldsymbol{N}_{t_j}^{\text{obs}}$.
7. Resample $\boldsymbol{x}_{t_j}^1, \ldots, \boldsymbol{x}_{t_j}^{n_p}$ using normalized importance weights proportional to w_j^l to get an equally weighted sample $\boldsymbol{x}_{t_j}^1, \ldots, \boldsymbol{x}_{t_j}^{n_p}$ of $\boldsymbol{X}_{t_j}$.

   Note that this is an unweighted sample of $f\left(\boldsymbol{x}_{t_j}\middle|\boldsymbol{x}_{t_0}, \ldots, \boldsymbol{x}_{t_{j-1}}, \boldsymbol{N}_{t_1}^{\text{obs}}, \ldots, \boldsymbol{N}_{t_j}^{\text{obs}}, \boldsymbol{\theta}_m\right) \propto P_{\text{obs}}\left(\boldsymbol{N}_{t_j}^{\text{obs}}\middle|\boldsymbol{h}(\boldsymbol{x}_{t_j}), \boldsymbol{\theta}_m\right) f_j\left(\boldsymbol{x}_{t_j}\middle|\boldsymbol{x}_{t_{j-1}}, \boldsymbol{\theta}_m\right)$ according to equation (9)**Error! Reference source not found.** since we draw from $f_j\left(\boldsymbol{x}_{t_j}\middle|\boldsymbol{x}_{t_{j-1}}, \boldsymbol{\theta}_m\right)$ and resample using weights proportional to $P_{\text{obs}}\left(\boldsymbol{N}_{t_j}^{\text{obs}}\middle|\boldsymbol{h}(\boldsymbol{x}_{t_j}), \boldsymbol{\theta}_m\right)$.
8. If $j < n$ increase j to j+1 and continue with step 5, otherwise continue with step 9.

   Note that after reaching $j = n$ we have a sample of size n_p of the states conditional on the observed abundances and the parameters $(\boldsymbol{X}_{t_0}, \ldots, \boldsymbol{X}_{t_n}|\boldsymbol{N}_{t_1}^{\text{obs}}, \ldots, \boldsymbol{N}_{t_n}^{\text{obs}}, \boldsymbol{\theta}_m)$.
9. Calculate the approximate marginal likelihood as $\hat{P}_{\text{marglikeli}}\left(\boldsymbol{N}_{t_1}^{\text{obs}}, \ldots, \boldsymbol{N}_{t_n}^{\text{obs}}|\boldsymbol{\theta}_m\right) = \prod_{j=1}^{n} \overline{w_j}$ with $\overline{w_j} = \frac{1}{n_p}\sum_{l=1}^{n_p} w_j^l$.

   See below for a derivation of this step.
10. If $m = 0$, continue with step 13, otherwise with step 11.
11. Accept $\boldsymbol{\theta}_m$ with probability $\min\left(1, \frac{\hat{P}_{\text{marglikeli}}\left(\boldsymbol{N}_{t_1}^{\text{obs}}, \ldots, \boldsymbol{N}_{t_n}^{\text{obs}}|\boldsymbol{\theta}_m\right) f_{\text{pri}}(\boldsymbol{\theta}_m)}{\hat{P}_{\text{marglikeli}}\left(\boldsymbol{N}_{t_1}^{\text{obs}}, \ldots, \boldsymbol{N}_{t_n}^{\text{obs}}|\boldsymbol{\theta}_{m-1}\right) f_{\text{pri}}(\boldsymbol{\theta}_{m-1})}\right)$, otherwise set $\boldsymbol{\theta}_m$ to $\boldsymbol{\theta}_{m-1}$.

    Note that this is a standard Metropolis acceptance/rejection step except that the marginal likelihood was replaced by its numerical estimate calculated with the particle filter.
12. Stop if Markov chain of parameters is sufficiently long (check convergence).
13. Set m to m+1, sample a new parameter vector $\boldsymbol{\theta}_m$ from a normal distribution around $\boldsymbol{\theta}_{m-1}$ and continue with step 3.

    Note: Another symmetrical proposal distribution would also work. A conventional adaptive scheme, e.g. by adapting the step length based on the current acceptance rate and manually updating the covariance matrix based on the samples accepted so far, can be used initially to find a proposal distribution that leads to an efficient Markov chain

sampler. A Metropolis-Hastings scheme could also be used; the non-symmetrical proposal distribution would then have to be considered in the acceptance probability in step 11.

The approximation of the likelihood in step 9 is based on the following: We need to derive conditional probability distributions to iteratively condition the states, $\boldsymbol{x}_{t_j}$, on the data, $\boldsymbol{N}_{t_1}^{\mathrm{obs}}, \dots, \boldsymbol{N}_{t_j}^{\mathrm{obs}}$, and to factorize the marginal likelihood. First, the state at time t_j can be conditioned on the previous state at time t_{j-1} and the observation at time t_j as follows (the interpretations of functions f and P without indices are defined by their arguments):

$$\begin{aligned} f\left(\boldsymbol{x}_{t_j} \middle| \boldsymbol{x}_{t_0}, \dots, \boldsymbol{x}_{t_{j-1}}, \boldsymbol{N}_{t_1}^{\mathrm{obs}}, \dots, \boldsymbol{N}_{t_j}^{\mathrm{obs}}, \boldsymbol{\theta}\right) &= f\left(\boldsymbol{x}_{t_j} \middle| \boldsymbol{x}_{t_{j-1}}, \boldsymbol{N}_{t_j}^{\mathrm{obs}}, \boldsymbol{\theta}\right) \\ &= \frac{P\left(\boldsymbol{N}_{t_j}^{\mathrm{obs}} \middle| \boldsymbol{x}_{t_{j-1}}, \boldsymbol{x}_{t_j}, \boldsymbol{\theta}\right) f_j\left(\boldsymbol{x}_{t_j} \middle| \boldsymbol{x}_{t_{j-1}}, \boldsymbol{\theta}\right)}{P\left(\boldsymbol{N}_{t_j}^{\mathrm{obs}} \middle| \boldsymbol{x}_{t_{j-1}}, \boldsymbol{\theta}\right)} \\ &\propto P_{\mathrm{obs}}\left(\boldsymbol{N}_{t_j}^{\mathrm{obs}} \middle| h(\boldsymbol{x}_{t_j}), \boldsymbol{\theta}\right) f_j\left(\boldsymbol{x}_{t_j} \middle| \boldsymbol{x}_{t_{j-1}}, \boldsymbol{\theta}\right) \quad . \end{aligned} \tag{9}$$

This equation allows us to sequentially sample the states $\boldsymbol{x}_{t_j}$ (from one observation time point to the next). Second, the marginal likelihood can be factorized as follows:

$$P_{\mathrm{marglikeli}}\left(\boldsymbol{N}_{t_1}^{\mathrm{obs}}, \dots, \boldsymbol{N}_{t_n}^{\mathrm{obs}} | \boldsymbol{\theta}\right) = P_1\left(\boldsymbol{N}_{t_1}^{\mathrm{obs}} | \boldsymbol{\theta}\right) \prod_{j=2}^{n} P_j\left(\boldsymbol{N}_{t_j}^{\mathrm{obs}} \middle| \boldsymbol{N}_{t_1}^{\mathrm{obs}}, \dots, \boldsymbol{N}_{t_{j-1}}^{\mathrm{obs}}, \boldsymbol{\theta}\right) \quad , \tag{10}$$

with

$$\begin{aligned} & P_1\left(\boldsymbol{N}_{t_1}^{\mathrm{obs}} | \boldsymbol{\theta}\right) = \int f_0(\boldsymbol{x}_{t_0} | \boldsymbol{\theta}) f_1\left(\boldsymbol{x}_{t_1} \middle| \boldsymbol{x}_{t_0}, \boldsymbol{\theta}\right) P_{\mathrm{obs}}\left(\boldsymbol{N}_{t_1}^{\mathrm{obs}} \middle| \boldsymbol{h}\left(\boldsymbol{x}_{t_1}\right), \boldsymbol{\theta}\right) \mathrm{d}\boldsymbol{x}_{t_0} \mathrm{d}\boldsymbol{x}_{t_1} \\ & P_j\left(\boldsymbol{N}_{t_j}^{\mathrm{obs}} \middle| \boldsymbol{N}_{t_1}^{\mathrm{obs}}, \dots, \boldsymbol{N}_{t_{j-1}}^{\mathrm{obs}}, \boldsymbol{\theta}\right) \\ & = \int f_j\left(\boldsymbol{x}_{t_j} \middle| \boldsymbol{x}_{\boldsymbol{t_{j-1}}}, \boldsymbol{\theta}\right) f\left(\boldsymbol{x}_{t_{j-1}} \middle| \boldsymbol{N}_{t_1}^{\mathrm{obs}}, \dots, \boldsymbol{N}_{t_{j-1}}^{\mathrm{obs}}, \boldsymbol{\theta}\right) P_{\mathrm{obs}}\left(\boldsymbol{N}_{t_j}^{\mathrm{obs}} \middle| \boldsymbol{h}\left(\boldsymbol{x}_{t_j}\right), \theta\right) \mathrm{d}\boldsymbol{x}_{t_{j-1}} \mathrm{d}\boldsymbol{x}_{t_j} \quad , \\ & j > 1 \quad . \end{aligned} \tag{11}$$

This equation allows us to calculate a numerical approximation to the marginal likelihood as it is needed for parameter inference (see equation (8)). Hence step 9 is an approximation to the equations (10)**Error! Reference source not found.** and (11)**Error! Reference source not found.** in which the integrals have been replaced by averages over all replicates, the terms $f_j\left(\boldsymbol{x}_{t_j} \middle| \boldsymbol{x}_{t_{j-1}}, \boldsymbol{\theta}_m\right)$ and $f\left(\boldsymbol{x}_{t_{j-1}} \middle| \boldsymbol{N}_{t_1}^{\mathrm{obs}}, \dots, \boldsymbol{N}_{t_{j-1}}^{\mathrm{obs}}, \boldsymbol{\theta}_m\right)$ are considered by using the sample of $\boldsymbol{X}_{t_{j-1}}$ after resampling and that of $\boldsymbol{X}_{t_j}$ before resampling, and $P_{\mathrm{obs}}\left(\boldsymbol{N}_{t_j}^{\mathrm{obs}} \middle| \boldsymbol{h}\left(\boldsymbol{x}_{t_j}\right), \boldsymbol{\theta}_m\right)$ is equal to the weights $\boldsymbol{w}_j^l$.

2.3 Application

To test the proposed algorithm, we ran the IBM to generate a sample of artificial data. To this end, parameter settings were chosen resulting in stable population dynamics (SI, Table S1). For environmental input we used water temperature measurements of an outdoor stream mesocosm experiment, interpolated to daily values. We chose on observation period of two years and sampled every 37 days, resulting in 20 data points. In this example, we assume the observation parameter *k* (=20) to be known. For real world applications, it could be estimated from independent observations or be part of the model parameters inferred.

To test the dependence of the algorithm's performance regarding the number of inferred parameters and their identifiability, we chose nine parameters that modify the individual vital rates and are easy to interpret: five parameters describing general aspects (f_{resp}, f_{mort}, $f_{\text{resp egg}}$, $f_{\text{mort egg}}$, $sd_{\text{f basal}}$) and four species specific ones ($f_{\text{ingest prey}}$, $f_{\text{ingest predator}}$, $K_{\text{dens prey}}$, $K_{\text{dens predator}}$). We inferred the parameters individually, in pairs of two, and jointly. We used log-normally distributed priors with the mean given by the true value used for data generation and a relative standard deviation of one. We approximated the marginal likelihood in the particle filtering step based on 100 particles (i.e. replicate model simulations). Model initialization is described in the SI. For joint estimation of all nine parameters, we applied 20 replicate chains of the PMCMC algorithm, and for estimation of one or two parameters at a time we used three replicates. These replicates were used to test for convergence (based on Gelman and Rubin's potential scale reduction factor using the function gelman.diag in the R package coda (Plummer et al., 2006)) and to improve the numerical accuracy of the result.

We assessed the impact of the number of particles on the marginal likelihood approximation by comparing the mean and variance of the marginal likelihood estimate for 10, 20, 50, 100, 500 and 1000 particles. To this end, we repeated the marginal likelihood estimation (i.e. the particle filtering steps 3-9) 100 times for 22 different parameter sets of nine parameters (true value, maximum posterior and 20 random samples from the MCMC chains).

The IBM and the particle filtering steps were coded in Java, and the MCMC algorithm was implemented in R (R Core Team, 2015). Parallelization was used to run several MCMC chains in parallel but not for the IBM or the particle filtering step.

3 Results

3.1 Algorithm performance

The suggested PMCMC algorithm performed well in our example, hence convergence was reached (Gelman and Rubin's potential scale reduction factor ≤ 1.02, upper confidence limit ≤ 1.03 for all

parameters). To achieve convergence in the MCMC chains with nine inferred parameters, more than 15000 iterations with 20 chains in parallel were necessary after a burn-in period of 1000 iterations (the burn-in period was short due to optimised starting values and covariance matrices from preliminary runs). The whole process required more than three weeks. One iteration within the algorithm consisting of 100 replicates, 20 observation time points and 2604 simulation time steps took approximately 90 s on 2.5 GHz CPUs.

To keep the computational burden within acceptable limits, we ran the algorithm with the minimum number of particles that led to an acceptable bias in the marginal likelihood. In our case, we used 100 particles to reduce the bias in the estimates (Figure 3; see SI Figure S1 for similar plots for all 22 parameter sets). Obviously, the use of more particles would increase the accuracy of the calculation.

In stochastic models, model simulations for a particular set of parameter values might by chance result in an exceptionally good fit, even if the number of particles is high. To prevent the MCMC algorithm from getting stuck when the marginal likelihood by chance was very high, the approximate marginal likelihood was recalculated if no new parameter set was accepted for 20 iterations, i.e. the particle filtering step was repeated for the last accepted parameter set. Note that this step should be avoided if possible as the convergence proof relies on not resampling the marginal likelihood. However, in the current application, this recalculation was necessary in only 0.38 % of all iterations (approx. 1100 times in more than 300000 iteration steps). Hence, we accept this small bias in order to speed up the parameter inference process. Increasing the number of particles would decrease the frequency of such an event, while it would further increase the computation time of each MCMC step (Figure 3).

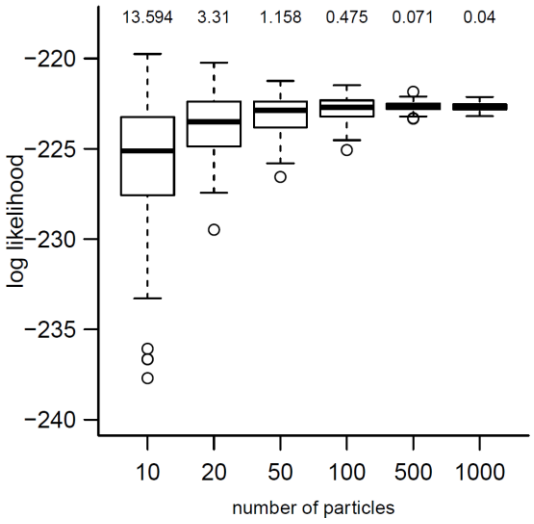


Figure 3: Boxplots of samples of the approximate marginal log likelihood for the maximum posterior data set as a function of the number of particles. The numbers on top give the variance based on 100 repetitions for each number of particles.

3.2 Comparison of prior and posterior distributions

When comparing marginal posterior with marginal prior distributions, a narrowing in the posterior indicates a gain in information resulting from new data. Additionally, the maximum of the posterior may be shifted compared to the prior. A small difference between a marginal prior and posterior indicates insufficient information in the data to learn about this parameter value, possibly due to correlations between parameters or due to insensitivity of the model results to the parameter.

When parameters were inferred individually (one at a time), the identifiability of most parameters was good. This is reflected in a narrow posterior around the true value, e.g. for the parameter $K_{\text{dens prey}}$, which describes the half-saturation constant for self-inhibition of prey (similar to Figure 3 A). When two or more parameters were inferred jointly, the identifiability depended on the parameter combinations. For instance, $K_{\text{dens prey}}$ was easy to infer if estimated together with $K_{\text{dens predator}}$ (Figure 4 left). However, $K_{\text{dens predator}}$ was not identifiable (very similar prior and posterior marginals) because the model is insensitive to this parameter at the chosen values for the other parameters. There was only a weak correlation present between these two parameters. $K_{\text{dens prey}}$ was no longer identifiable when inferred jointly with the parameter $f_{\text{ingest prey}}$, the multiplication factor of the basal metabolic rate for ingestion, due to the strong correlation of these two parameters (Figure 4 right).

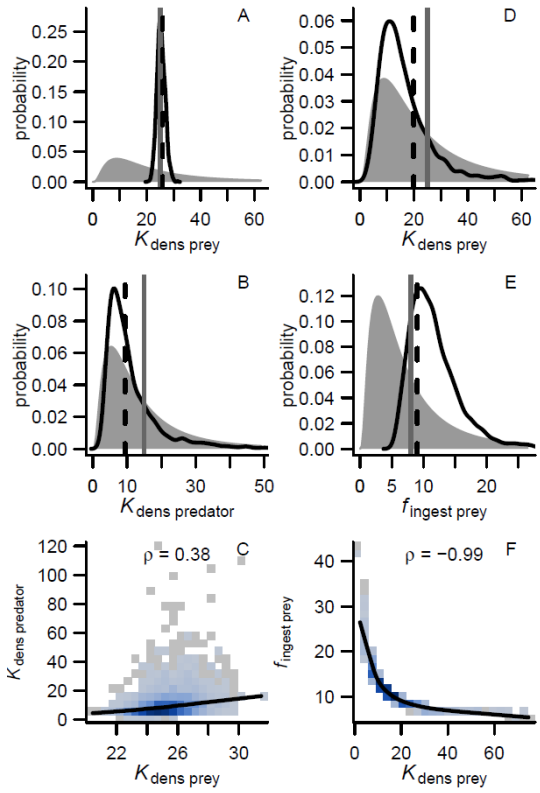
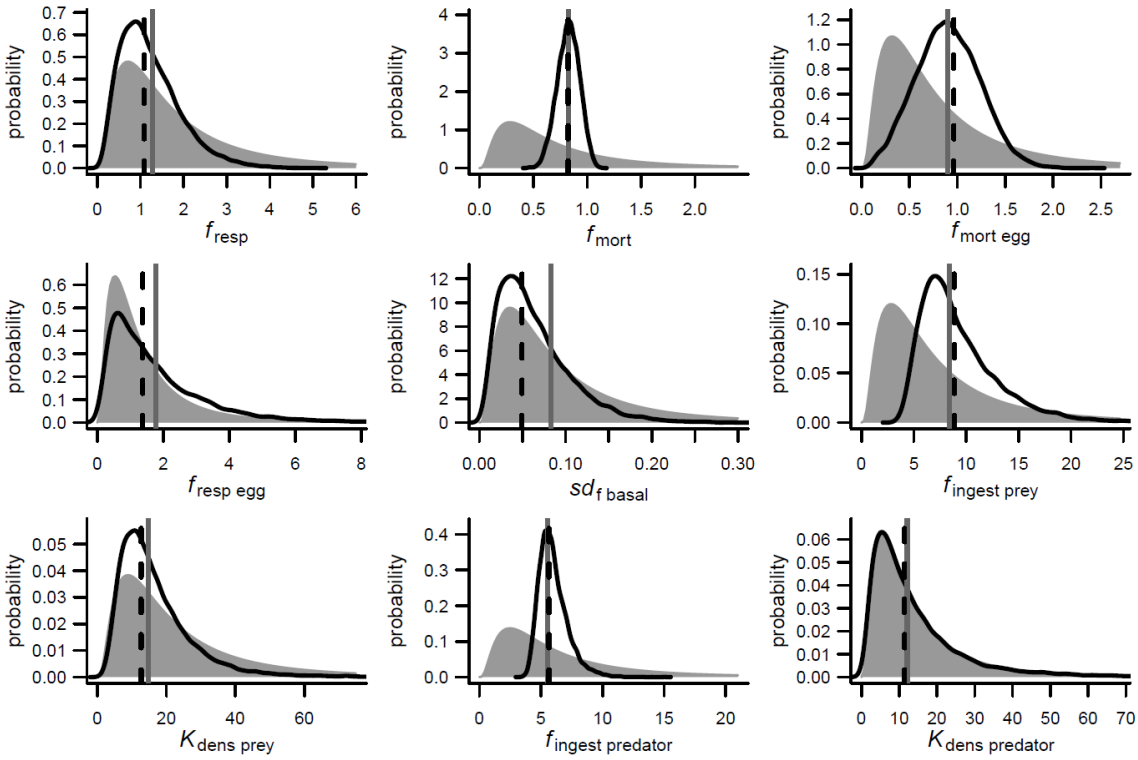


Figure 4: Comparison of marginal prior and posterior probability densities (A, B, D, E) and visualization of joint two dimensional posterior probability densities and the corresponding correlation coefficients (Spearman's ρ; C, F). Left: joint inference of parameters $K_{dens\ prey}$ and $K_{dens\ predator}$; right: joint inference of parameters $K_{dens\ prey}$ and $f_{ingest\ prey}$; all other parameters were fixed at the values used for data generation. Grey area: marginal prior probability density; black line: marginal posterior probability density; grey line: true value used for data generation, black dashed line: parameter value at the maximum posterior density (A, B, D, E). Grey to blue colour gradient: posterior density at sample points; black line: local smoother (C, F).

When we inferred nine parameters jointly, some were clearly identifable and the parameter values at the overall maximum posterior density was in accordance with the true values used for data generation (f_{mort}, $f_{mort\ egg}$, $f_{ingest\ prey}$, $f_{ingest\ predator}$; Figure 5). For $f_{ingest\ predator}$, the maximum posterior estimate was slightly shifted toward smaller values. Other parameters were not identifiable. Again, some strong parameter correlations were present (f_{resp} – $f_{ingest\ predator}$ Spearman's ρ = 0.79, f_{mort} – $f_{mort\ egg}$ ρ = -0.85, $f_{ingest\ prey}$ – $K_{dens\ prey}$ ρ = -0.78; SI, Figure S2).



409

410 Figure 5: Marginal prior and posterior probability densities for the joint inference of nine parameters; all other
411 parameters were fixed at the values used for data generation. Grey area: marginal prior probability density;
412 black line: marginal posterior probability density; grey line: true value used for data generation; black dashed
413 line: parameter value at the maximum posterior density.

414

415 ## 3.3 Comparison of observations and simulations

416 The simulation results of the particle filtering run with the parameter values at the maximum posterior
417 density corresponded well with the observations (Figure 6 dark grey area; joint estimation of nine
418 parameters). By conditioning the simulations on the observations, the particle filtering algorithm
419 "dragged" the simulations towards the observations (particularly visible in the predator simulation).
420 The range of the simulation outputs of the run with the maximum posterior density indicated the
421 stochasticity of the model. The simulations using a sample from the posterior were slightly wider than
422 those at the maximum posterior, as would be expected (Figure 6 medium grey area). Moreover,
423 simulations without particle filtering (posterior check) using a sample from the posterior were still
424 wider but also captured the pattern in the dynamics and the observations lay within the simulation
425 results (Figure 6, light grey area).

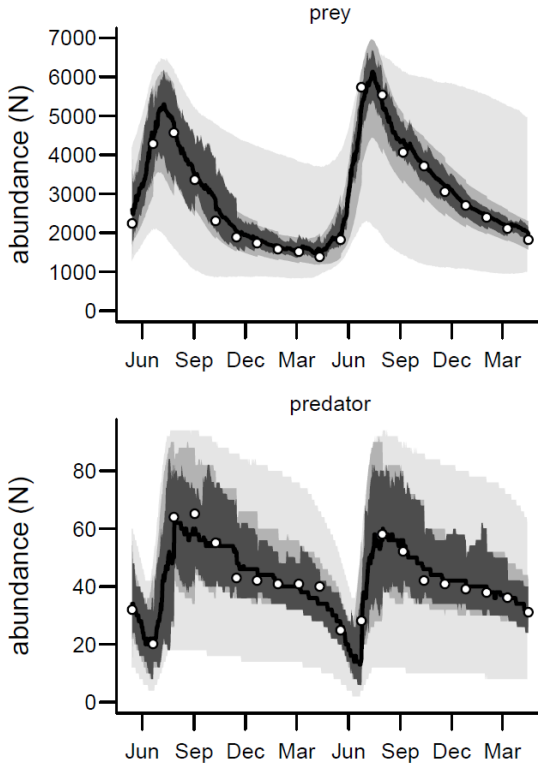


Figure 6: Population dynamics over time displaying the posterior knowledge of the true predator and prey abundances. Observed abundance used for parameter inference (white dots); distribution of predicted abundances conditioned on the observations at the maximum of the posterior of the parameters, $\boldsymbol{h}\left(\boldsymbol{X_t}\middle|\boldsymbol{N}_{\boldsymbol{t_1}}^{\mathbf{obs}},\ldots,\boldsymbol{N}_{\boldsymbol{t_n}}^{\mathbf{obs}},\boldsymbol{\theta}_{\mathbf{max\,post}}\right)$, visualized by the 5 – 95% quantile range of 100 particles used (dark grey area) and the median (black line); analogous distribution considering the full posterior, $\boldsymbol{h}\left(\boldsymbol{X_t}\middle|\boldsymbol{N}_{\boldsymbol{t_1}}^{\mathbf{obs}},\ldots,\mathbf{N}_{\mathbf{t_n}}^{\mathbf{obs}},\boldsymbol{\Theta}_{\mathbf{post}}\right)$ (medium grey area), using 5000 samples from the posterior of the parameters and one sample simulation from 100 particles from each of these; posterior check depicted by the 5 – 95% quantile range of one run each without particle filtering using 5000 samples from the posterior of the parameters (light grey area).

4 Discussion

In this study, we successfully demonstrate the applicability of a PMCMC algorithm for parameter inference for IBMs with time-series data. The MCMC chains converged and yielded interpretable posteriors distributions. Additionally, predictions from the posterior covered well the observations.

Due to the lack of studies using alternative techniques for Bayesian inference of IBMs in ecological research, we cannot quantitatively validate our results. However, due to the good performance of the algorithm we are confident that the PMCMC approach chosen is a feasible way of doing Bayesian inference for IBMs. One drawback of the approach is the relatively long computation time of one MCMC step (~90 s, depending on the number of particles) and thus the long runtime necessary to reach convergence in the MCMC chains. Hence, the approach is not yet applicable to more complex IBMs with longer runtimes. Long runtimes and high resource demand are general problems of IBMs,

increasingly attenuated, however, by ever-growing computing capacity. Parallelization of both the IBM as well as the algorithm (i.e. running the replicate simulations in parallel) would increase the computation speed and thereby enhance the range of applicability of the approach. This should be kept in mind when implementing the IBM. Another technical difficulty might be the particle sampling at each observation time step. To this end, the complete state of the IBM, including all individuals and their current properties, needs to be stored and replicated. This can require considerable storage capacity.

A sufficiently large number of particles used for inference is crucial to avoid bias of the likelihood approximation. On the other hand, the number has to be limited to keep the run time in feasible limits. A compromise can be found by testing the stability of the likelihood approximation based on a sample of parameter sets (e.g. from the prior) with various numbers of particles as demonstrated in Figure 3. The optimal number of particles needs to take into account both the efficiency of the algorithm (including acceptance rate; i.e. not getting "stuck") and the computing time and it depends on the bias and variance in the approximated likelihood for one particular parameter set. The variance found with 100 particles tested for several parameter sets (Figure 3 and Figure S1) was in general below the value of 3.28 recommend for this variance under the assumption of unbiased estimates of the likelihood (Sherlock et al., 2015). Due to the typically longer run times of IBMs compared to other stochastic models, the overall run time of the inference process might still be large even for an optimised number of particles.

As this was one of the first proofs of a concept concerning a numerical implementation of Bayesian inference of model parameters for IBMs, there are several research directions that should be explored. One direction is to test the potential of alternative approaches and compare them directly for the same IBM. One option are rejection methods that sample from the prior to propose new parameter values, including rejection ABC (Marchand et al., 2015; van der Vaart et al., 2015). These techniques have the disadvantage that they can be expected to be much less efficient than the algorithm presented, particularly if there is significant information gain. This disadvantage may be partially offset by the ease with which extreme parallelization can be utilized (Marjoram et al., 2003). ABC can also be applied within an MCMC approach (Foley et al., 2015). Both may profit from the fact that only one run per set of parameter values is needed in contrast to several ones in the particle filtering approach. Additional options are other ABC techniques, e.g. the simulated annealing approach to ABC (Albert et al., 2014). All these ABC approaches do not explicitly make use of the hidden Markov model structure and therefore may have a disadvantage in terms of efficiency but might over-compensate for that by other, more efficient aspects. An additional advantage of the PMCMC presented is that it does not rely on summary statistics, which can lead to a substantial loss of information (Robert et al., 2011). Owen et

al. (2015) concluded that for stochastic kinetic models, PMCMC was mostly more efficient than ABC in particular for longer time-series data with fine temporal resolution. This advantage may decrease with increasing stochasticity of the model behaviour between consecutive observation time steps that decreases the efficiency of the particle filtering process. In such cases, in addition to a finer temporal resolution of the observations, ideas to increase the efficiency of the particle filtering process, such as the use of a fast approximation (Golightly et al., 2015) may become important. Another research direction is to try to improve the efficiency of the approach presented and compare the PMCMC approach chosen with an SMC-SMC approach that also applies particle filtering for parameter inference (Chopin et al., 2013). Eventually, performance comparisons of different approaches for different IBMs may lead to guidelines regarding which method to choose for which inference problem.

In conclusion, the algorithm presented was capable of estimating the parameters of a two-species IBM from artificial data. Hence, the PMCMC algorithm offers a new possibility for parameter estimation and uncertainty assessment for IBMs. It demonstrates the feasibility of Bayesian inference for IBMs, at least currently for IBMs with relatively short runtimes. We hope that our study will stimulate the development of even more efficient numerical implementations of Bayesian inference for IBMs and their application. This may significantly advance environmental and ecological research with IBMs due to additional insights into model behaviour and uncertainty in particular when compared to forward simulations to which ecological studies with ABMs/IBMs are currently often limited.

Acknowledgments

We thank Nele Schuwirth, Carlo Albert and Oliver Jakoby for discussions and constructive comments on the manuscript. We also thank three anonymous reviewers for helpful comments on an earlier version of this manuscript.

Refernces

Albert, C., Künsch, H.R., Scheidegger, A. (2014) A simulated annealing approach to approximate Bayes computations. Stat. Comput., 1-16.

Andrieu, C., Doucet, A., Holenstein, R. (2010) Particle Markov chain Monte Carlo methods. J. Roy. Stat. Soc. B, 72, 269-342.

Andrieu, C., Roberts, G.O. (2009) The pseudo-marginal approach for efficient Monte Carlo computations. The Annals of Statistics, 37, 697-725.

Beaumont, M.A. (2003) Estimation of population growth or decline in genetically monitored populations. Genetics, 164, 1139-1160.

Beaumont, M.A. (2010) Approximate Bayesian Computation in Evolution and Ecology. Annual Review of Ecology, Evolution, and Systematics, Vol 41 (eds D.J. Futuyma, H.B. Shafer, D. Simberloff), pp. 379-406.

Brown, D.G., Page, S.E., Riolo, R., Rand, W. (2004) Agent-based and analytical modeling to evaluate the effectiveness of greenbelts. Environ. Modell. Softw., 19, 1097-1109.

Brown, J.H., Gillooly, J.F., Allen, A.P., Savage, V.M., West, G.B. (2004) Toward a metabolic theory of ecology. Ecology, 85, 1771-1789.

Blum, M.G.B., Nunes, M.A., Prangle, D., Sisson, S.A. (2013) A Comparative Review of Dimension Reduction Methods in Approximate Bayesian Computation. Stat. Sci., 28, 189-208.

Chopin, N., Jacob, P.E., Papaspiliopoulos, O. (2013) SMC^2: an efficient algorithm for sequential analysis of state space models. J. Roy. Stat. Soc. B, 75, 397-426.

Csillery, K., Blum, M.G.B., Gaggiotti, O.E., Francois, O. (2010) Approximate Bayesian Computation (ABC) in practice. Trends Ecol. Evol., 25, 410-418.

DeAngelis, D.L., Mooij, W.M. (2005) Individual-based modeling of ecological and evolutionary processes. Annu. Rev. Ecol. Evol. S., pp. 147-168.

Elliot, J.M. (1971) Some methods for the statistical analysis of samples of benthic invertebrates. Freshwater Biological Association, Kendal.

Ellison, A.M. (2004) Bayesian inference in ecology. Ecol. Lett., 7, 509-520.

Fearnhead, P., Prangle, D. (2012) Constructing summary statistics for approximate Bayesian computation: semi-automatic approximate Bayesian computation. J. Roy. Stat. Soc. B, 74, 419-474.

Flury, T., Shephard, N. (2011) Bayesian inference based only on simulated likelihood: particle filter analysis of dynamic economic models. Economet. Theor. 27, 933-956.

Foley, B., Saltz, J., Nuzhdin, S., Marjoram, P. (2015) A Bayesian Approach to Social Structure Uncovers Cryptic Regulation of Group Dynamics in Drosophila melanogaster. Am. Nat., 185, 797-808.

Gamerman, D., Lopes, H.F. (2006) Markov Chain Monte Carlo: Stochastic Simulation for Bayesian Inference. Taylor & Francis Inc.

Gelman, A., Carlin, J., Stern, H., Dunson, D., Vehtari, A., Rubin, D. (2014) Bayesian Data Analysis, Third edn. Chapman & Hall.

Golightly, A., Henderson, D.A., Sherlock, C. (2015) Delayed acceptance particle MCMC for exact inference in stochastic kinetic models. Stat. Comput., 25, 1039-1055.

Grimm, V. (1999) Ten years of individual-based modelling in ecology: what have we learned and what could we learn in the future? Ecol. Model., 115, 129-148.

Grimm, V., Augusiak, J., Focks, A., Frank, B.M., Gabsi, F., Johnston, A.S.A., Liu, C., Martin, B.T., Meli, M., Radchuk, V., Thorbek, P., Railsback, S.F. (2014) Towards better modelling and decision support: Documenting model development, testing, and analysis using TRACE. Ecol. Model., 280, 129-139.

Grimm, V., Berger, U., Bastiansen, F., Eliassen, S., Ginot, V., Giske, J., Goss-Custard, J., Grand, T., Heinz, S.K., Huse, G., Huth, A., Jepsen, J.U., Jorgensen, C., Mooij, W.M., Muller, B., Pe'er, G., Piou, C., Railsback, S.F., Robbins, A.M., Robbins, M.M., Rossmanith, E., Ruger, N., Strand, E., Souissi, S., Stillman, R.A., Vabo, R., Visser, U., DeAngelis, D.L. (2006) A standard protocol for describing individual-based and agent-based models. Ecol. Model., 198, 115-126.

Grimm, V., Berger, U., DeAngelis, D.L., Polhill, J.G., Giske, J., Railsback, S.F. (2010) The ODD protocol A review and first update. Ecol. Model., 221, 2760-2768.

Grimm, V., Railsback, S.F. (2005) Individual-based Modeling and Ecology. Princeton University Press.

Grimm, V., Revilla, E., Berger, U., Jeltsch, F., Mooij, W.M., Railsback, S.F., Thulke, H.H., Weiner, J., Wiegand, T., DeAngelis, D.L. (2005) Pattern-oriented modeling of agent-based complex systems: Lessons from ecology. Science, 310, 987-991.

Hartig, F., Calabrese, J.M., Reineking, B., Wiegand, T., Huth, A. (2011) Statistical inference for stochastic simulation models - theory and application. Ecol. Lett., 14, 816-827.

Hartig, F., Dislich, C., Wiegand, T., Huth, A. (2014) Technical Note: Approximate Bayesian parameterization of a process-based tropical forest model. Biogeosciences, 11, 1261-1272.

Hastings, W.K. (1970) Monte Carlo sampling methods using Markov chains and their applications. Biometrika, 57, 97–109.

Huston, M., DeAngelis, D., Post, W. (1988) New computer-models unify ecological theory - Computer-simulations show that many ecological patterns can be explained by interactions among individual organisms. Bioscience, 38, 682-691.

572 Kantas, N., Doucet, A., Singh, S.S., Maciejowski, J., Chopin, N. (2015) On particle methods for
573 parameter estimation in state-space models. Stat. Sci., 30, 328-351
574 Kattwinkel, M., Reichert, P., Rüegg, J., Liess, M., Schuwirth, N. (2016) Modelling macroinvertebrate
575 community dynamics in stream mesocosms contaminated with pesticide. Environ. Sci.
576 Technol., 50, 3165-3173.
577 Lagarrigues, G., Jabot, F., Lafond, V., Courbaud, B. (2015) Approximate Bayesian computation to
578 recalibrate individual-based models with population data: Illustration with a forest simulation
579 model. Ecol. Model., 306, 278-286.
580 Marchand, P., Harmon-Threatt A.-N., Chapela, I. (2015) Testing models of bee foraging behavior
581 through the analysis of pollen loads and floral density data. Ecol. Model., 313, 41-49.
582 Marjoram, P., Molitor, J., Plagnol, V., Tavare, S. (2003) Markov chain Monte Carlo without likelihoods.
583 P. Nat. Acad. Sci. USA, 100, 15324-15328.
584 Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A. and Teller, E. (1953) Equation of state
585 calculations by fast computing machines. J. Chem. Phys., 21, 1087–1092.
586 Owen, J., Wilkinson, D.J. and Gillespie, C.S. (2015) Likelihood free inference for Markov processes: a
587 comparison. Stat. Appl. Genet. Mo. B., 14, 189-209.
588 Plummer, M., Best, N., Cowles, K., Vines, K. (2006) CODA: Convergence Diagnosis and Output Analysis
589 for MCMC, R News, 6, 7-11. R Core Team (2015) R: A Language and Environment for
590 Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria.
591 R Core Team (2015). R: A language and environment for statistical computing. R Foundation for
592 Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.
593 Refsgaard, J.C., van der Sluijs, J.P., Højberg, A.L., Vanrolleghem, P.A. (2007) Uncertainty in the
594 environmental modelling process - A framework and guidance. Environ. Modell. Softw., 22,
595 1543-1556.
596 Reichert, P., Langhans, S., Lienert, J. and Schuwirth, N. (2015) The Conceptual Foundation of
597 Environmental Decision Support. J. Environ. Manage. 154, 316-332.
598 Robert, C.P., Cornuet, J.-M., Marin, J.-M., Pillai, N.S. (2011) Lack of confidence in approximate
599 Bayesian computation model choice. P. Nat. Acad. Sci. USA, 108, 15112-15117.
600 Schuwirth, N., Dietzel, A., Reichert, P. (2015) The importance of biotic interactions for the prediction
601 of macroinvertebrate communities under multiple stressors. Funct. Ecol., 30, 974-984.
602 Schuwirth, N., Reichert, P. (2013) Bridging the gap between theoretical ecology and real ecosystems:
603 modeling invertebrate community composition in streams. Ecology, 94, 368-379.
604 Sherlock, C., Thiery, A.H., Roberts, G.O., Rosenthal, J.S. (2015) On the efficiency of pseudo-marginal
605 random walk Metropolis algorithms. The Annales of Statistics, 43, 238–275

606 Sibly, R.M., Grimm, V., Martin, B.T., Johnston, A.S.A., Kulakowska, K., Topping, C.J., Calow, P., Nabe-
607 Nielsen, J., Thorbek, P., DeAngelis, D.L. (2013) Representing the acquisition and use of energy
608 by individuals in agent-based models of animal populations. Methods Ecol. Evol., 4, 151-161.
609 Thiele, J.C., Kurth, W., Grimm, V. (2014) Facilitating Parameter Estimation and Sensitivity Analysis of
610 Agent-Based Models: A Cookbook Using NetLogo and 'R'. JASSS, 17, 11.
611 Topping, C.J., Dalkvist, T., Grimm, V. (2012) Post-Hoc Pattern-Oriented Testing and Tuning of an
612 Existing Large Model: Lessons from the Field Vole. PLoS ONE, 7.
613 van der Vaart, E., Beaumont, M.A., Johnston, A.S.A., Sibly, R.M. (2015) Calibration and evaluation of
614 individual-based models using Approximate Bayesian Computation. Ecol. Model., 312, 182-
615 190.