

UDC: 519.245:577.27

Markov chain Monte Carlo parameter estimation of the ODE compartmental cell growth model

Luzyanina T. ¹, Bocharov G. ²

¹ *Institute of Mathematical Problems of Biology RAS – The Branch of Keldysh Institute of Applied Mathematics of the Russian Academy of Sciences, Pushchino, Russian Federation*

² *Marchuk Institute of Numerical Mathematics, Russian Academy of Sciences, Moscow, Russian Federation*

Abstract. We use a Markov chain Monte Carlo (MCMC) method to quantify uncertainty in parameters of the heterogeneous linear compartmental model of cell population growth, described by a system of ordinary differential equations. This model allows division number-dependent rates of cell proliferation and death and describes the rate of changes in the numbers of cells having undergone j divisions. The experimental data set specifies the following characteristics of the kinetics of human T lymphocyte proliferation assay in vitro: the total number of live cells and dead but not disintegrated cells and the number of cells divided j times. Our goal is to compare results of the MCMC analysis of the uncertainty in the best-fit parameter estimates with the ones obtained earlier, using the variance-covariance approach, the profile-likelihood based approach and the bootstrap technique. We show that the computed posterior probability density functions are Gaussian for most of the model parameters and they are close to Gaussian ones for other parameters except one. We present posterior uncertainty limits for the model solution and new observations.

Key words: *cell population dynamics, Markov chain Monte Carlo analysis, CFSE assay, heterogenous compartmental model, parameter estimation, uncertainty.*

2. Introduction

The turnover of cell populations is a central process underlying the functioning of immune system. Knowledge of the proliferation and death rates of the immune cells, e.g. T lymphocytes, is required to understand and predict the efficiency of the system in health and disease. For example, for HIV infection it was shown that small variations in parameter values of the virus-host immune system have a large impact on the viral load set-point, and therefore, the onset time of AIDS [1]. To evaluate the proliferative performance of lymphocytes various labeling techniques, mathematical models have been proposed, including deterministic ODEs, PDEs, as well as the stochastic branching process-based frameworks (see for a comprehensive overview [2, 3]).

Assessment of the robustness of the mathematical model-based predictions of cell systems dynamics and evaluation of the effect of drugs on lymphocyte dynamics (e.g., anti-PD1 treatment) require the computation of uncertainty for the parameter estimates. Two complementary types of approaches, i.e., the frequentist- and the Bayesian frameworks, can be used [4, 5] to quantify the uncertainties. The critical issue of which approach should one use in a given problem was investigated in [6].

We have previously presented a systematic comparison of three methods belonging to the class of the frequentist (maximum likelihood) methodology: the variance-covariance-, the

likelihood-profile- and the bootstrapping (both non- and parametric) methods cf. [7]. These are extensively used in quantitative modelling of cell population dynamics. During the last years Markov chain Monte Carlo (MCMC) approach has become rather popular, see, e.g. [4, 8–10]. We apply a MCMC method to quantify uncertainty in parameters of the heterogeneous linear compartmental model of the labeled cell population dynamics, formulated as system of ordinary differential equations (ODEs). The MCMC method, which we use, is implemented as the Matlab toolbox [11], which can be downloaded at <http://helios.fmi.fi/~lainema/mcmc/> with some examples.

Our goal is to compare results of the MCMC analysis with the ones obtained earlier, using the variance-covariance approach, the profile-likelihood based approach and the bootstrap technique, cf. [7]. We show that the computed posterior probability density functions are Gaussian for most of the parameters and they are close to Gaussian ones for other parameters except one. We present posterior limits of the model uncertainty and new observations.

In this section we briefly introduce a set of experimental data used for fitting parameters of the mathematical model and the model itself, see for details [7].

CFSE* based tracking of the T-lymphocyte proliferation using flow cytometry is a powerful experimental technique in immunology allowing for the tracing of labeled cell populations over time in terms of the number of divisions cells undergone. The proliferation of CFSE labeled cells is detected as progressive halving of cellular fluorescence with every cell division completed. CFSE histograms give information on the fraction of lymphocytes that have divided once, twice, etc. The percentage of T-cells having undergone from 1 up to 10 divisions can be quantified by the flow cytometry.

During the first three days of culture there is a significant number of live cells in the well, which are not lymphocytes. This implies that the dead cell data, at least initially, characterize the death of heterogeneous cell population, rather than the T lymphocyte population. Another confounding aspect of the data is that the transfer of the peripheral blood mononuclear cells (PBMCs) from physiological in vivo conditions into in vitro culture will cause a death of many cells due to the change in milieu, temperature, plastic walls, etc. Notice that the specificity of T-lymphocyte gating gets better with time, and from day 4 to day 7 the gate used covers almost all live T cells as counted.

Table 1 presents the set of CFSE data that we analyzed. These data specify the following characteristics of the kinetics of PHA[†]-induced T lymphocyte proliferation assay in vitro from day 3 to day 7: (i) the total number of live cells, $N(t_i)$, (ii) the total number of dead but not disintegrated cells, $D(t_i)$, and (iii) the number of cells divided j times, $N_j(t_i)$, $i = 0, 1, \dots, 4$, $j = 0, 1, \dots, 7$.

The heterogeneous compartmental model of cell population turnover assumes that the per capita proliferation and death rates of T-lymphocytes, α_j , respectively β_j , depend on the number of divisions lymphocytes undergone. The rates of change of $N_j(t)$ and $D(t)$ with time are represented by the set of ordinary differential equations

$$\begin{aligned} \frac{dN_0}{dt}(t) &= -(\alpha_0 + \beta_0)N_0(t), \\ \frac{dN_j}{dt}(t) &= 2\alpha_{j-1}N_{j-1}(t) - (\alpha_j + \beta_j)N_j(t), \quad j = 1, \dots, 7, \\ \frac{dD}{dt}(t) &= \sum_{j=0}^7 \beta_j N_j(t) - \delta D(t). \end{aligned} \tag{1}$$

*carboxyfluorescein diacetate succinimidyl ester

†phytohemagglutinin: A phytomitogen from plants that agglutinates red blood cells. The term is commonly used specifically for the lectin obtained from the red kidney bean (*Phaseolus vulgaris*) which is also a mitogen that stimulates T lymphocytes more vigorously than B lymphocytes

Table 1. Quantitative dynamics of human peripheral blood T lymphocytes following stimulation with PHA in vitro. At various times, CFSE distribution profiles of lymphocyte populations were obtained by flow cytometry. The total numbers of live, $N(t_i)$, and dead, $D(t_i)$, lymphocytes, and the distribution of lymphocytes with respect to the number of divisions they have undergone, $N_j(t_i)$, $j = 0, 1, \dots, 7$, were followed from day 3 to day 7 at the indicated times t_i , $i = 0, 1, \dots, 4$. Adapted by permission from Springer, J. of Math. Biology. Computational analysis of CFSE proliferation assay. T. Luzyanina, S. Mrusek, J. T. Edwards, D. Roose, S. Ehl, G. Bocharov. Copyright 2006

Time days t_i	Total number of live cells $N(t_i)$	Total number of dead cells $D(t_i)$	Numbers of cells w.r.t. the number of divisions (j) they undergone $N_j(t_i)$							
			0	1	2	3	4	5	6	7
3	1.4×10^5	1.6×10^4	29358	22876	43372	39970	5208	98	14	0
4	2.5×10^5	2.4×10^4	16050	12600	22650	57025	96350	46950	2500	25
5	4.4×10^5	6.0×10^4	14476	14784	25344	58652	141460	156290	32076	440
6	5.0×10^5	1.2×10^5	13500	12150	24150	55000	137850	188950	69450	2150
7	5.7×10^5	1.3×10^5	13509	12198	21603	51927	140560	232160	96102	3420

The first term on the right of equations for $N_j(t)$ represents the cell birth (influx from previous compartment because of division), while the last term on the right represents cell loss (outflux from the compartment) due to division and death. In the equation for dead cells, δ denotes the specific (fractional) decay rate of dead lymphocytes due to disintegration and phagocytosis.

Assuming that the population sizes at time t_0 are specified by initial data $N_j(t_0)$ and $D(t_0)$, and the condition $\alpha_j + \beta_j \neq \alpha_i + \beta_i$ is fulfilled for $i \neq j$, the solution of the model can be expressed in the form

$$\begin{aligned}
 N_j(t) &= \sum_{s=1}^j \left\{ 2^s N_{j-s}(t_0) \prod_{m=j-s}^{j-1} \alpha_m \sum_{i=j-s}^j e^{-c_i(t-t_0)} \prod_{k=j-s, k \neq i}^j (c_k - c_i)^{-1} \right\} + N_j(t_0) e^{-c_j(t-t_0)}, \\
 & \quad j = 0, 1, \dots, 7, \quad t \geq t_0, \\
 D(t) &= \sum_{j=0}^7 \beta_j \left\{ \sum_{s=1}^j 2^s N_{j-s}(t_0) \prod_{m=j-s}^{j-1} \alpha_m \sum_{i=j-s}^j \frac{e^{-c_i(t-t_0)} - e^{-\delta(t-t_0)}}{\delta - c_i} \prod_{k=j-s, k \neq i}^j (c_k - c_i)^{-1} \right\} \\
 & \quad + \sum_{j=0}^7 \beta_j N_j(t_0) \frac{e^{-c_j(t-t_0)} - e^{-\delta(t-t_0)}}{\delta - c_j} + D(t_0) e^{-\delta(t-t_0)}, \quad t \geq t_0,
 \end{aligned} \tag{2}$$

where $c_j := \alpha_j + \beta_j$. The availability of the closed-form solution to the model reduces the computational treatment of the model-driven data analysis, i.e., the parameter estimation and uncertainty evaluation.

The paper is structured as follows. In Sect. 2 we briefly present results on parameter estimation of the model and their confidence intervals, see for details [7]. Section 3 contains some useful information about the MCMC toolbox which we use. In Sect. 4 we apply the MCMC code to system (1) and present results. Section 5 contains conclusions.

3. Model parameters estimation and their confidence intervals

To estimate the division and death rates of cells, we fit a model to the given data set by maximizing the likelihood that the data did arise from the model. Applying the maximum likelihood approach to our problem, we assume that (i) the observational errors, i.e. the residuals defined as a difference between observed and model-predicted values, are normally distributed, (ii) the errors in observations at successive times are independent, (iii) the errors in the components of the state vector are independent, (iv) the variance of observation errors (σ^2) is the same for all the state variables and observation times.

Under these conditions, the log-likelihood function specifying the probability of observing the given data set is given by

$$\ln(\mathcal{L}(\mathbf{p}; \sigma)) = -0.5 \left(n_d \ln(2\pi) + n_d \ln(\sigma^2) + \sigma^{-2} \Phi(\mathbf{p}) \right), \quad (3)$$

where n_d is the total number of scalar measurements and $\Phi(\mathbf{p})$ is an ordinary least-squares function

$$\Phi(\mathbf{p}) = \sum_{i=1}^4 \left(\sum_{j=0}^7 (N_j^i - N_j(t_i; \mathbf{p}))^2 + (D^i - D(t_i; \mathbf{p}))^2 \right), \quad (4)$$

where \mathbf{p} is the vector of estimated parameters.

The problem of maximizing the likelihood function is equivalent to that of minimizing $\Phi(\mathbf{p})$, provided that σ^2 is assigned the value

$$\sigma^{*2} = \frac{1}{n_d} \Phi(\mathbf{p}^*), \quad (5)$$

which follows from the optimality condition $\partial(\ln(\mathcal{L}(\mathbf{p}^*; \sigma)))/\partial\sigma^2 = 0$. Here \mathbf{p}^* is the parameter vector, which gives a minimum to the ordinary least-squares function.

The experimental data and the solution of the model corresponding to the best-fit parameter estimates are shown in Fig. 1. Table 2 presents the best-fit values of the parameters of model (1), estimated using the data in Table 1, and computational estimates of 95% confidence intervals for them approximated by the variance-covariance, profile-likelihood and bootstrap methods, see for details [7].

Parameter estimation results obtained using the ordinary least-squared approach. The death rates appear to be close to zero (ranging between 10^{-15} and 10^{-11} , not shown in Table 2) until the division age of cells gets three and after that they start to increase. The prediction of no death in cells, that have made six and more divisions, might rather reflect that the data for corresponding populations are not informative enough for reliable estimation of the death rate. Biologically, these small values would imply zero death rate of the proliferating cells with division number age from zero to three, six and seven. The CFSE data set presented in Table 1 does not ensure a reliable estimation of the division rate for cells which have undergone more than six divisions. Indeed, the best-fit value of α_7 is very high. Because this parameter affect only the kinetics of cells which have made seven divisions, the unrealistic values might be attributed to a relatively small contribution of the experimental data, characterizing the number of cells having done seven divisions, to the objective function. Therefore we can reduce the vector of estimated parameters to the following components

$$\mathbf{p} = [\alpha_0, \alpha_1, \alpha_2, \dots, \alpha_7, \beta_4, \beta_5, \delta]. \quad (6)$$

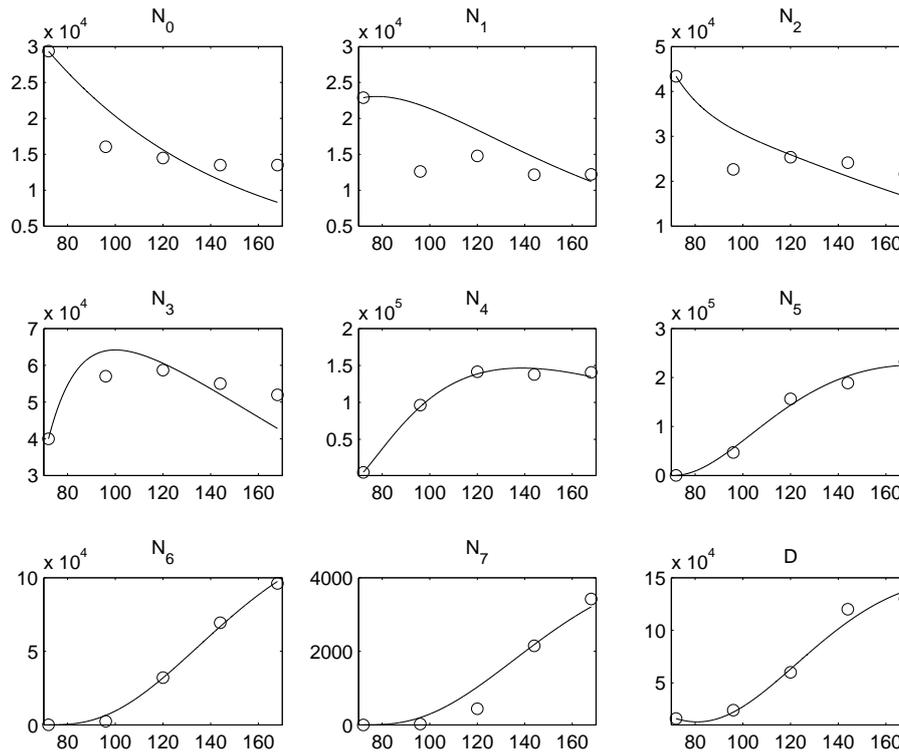


Fig. 1. Experimental data (\circ) and the best-fit solution of the model (1) (solid lines). Adapted by permission from Springer, *J. of Math. Biology*. Computational analysis of CFSE proliferation assay. T. Luzyanina, S. Mrusek, J. T. Edwards, D. Roose, S. Ehl, G. Bocharov. Copyright 2006.

Table 2. Computational estimates of 95% confidence intervals for the best-fit parameter values of the heterogenous model (1) approximated by the variance-covariance, profile-likelihood and bootstrap methods. Adapted by permission from Springer, *J. of Math. Biology*. Computational analysis of CFSE proliferation assay. T. Luzyanina, S. Mrusek, J. T. Edwards, D. Roose, S. Ehl, G. Bocharov. Copyright 2006

\mathbf{p} hr^{-1}	Best-fit values	Estimates of 95% confidence intervals		
		variance-covariance method	profile likelihood method	bootstrap method
α_0	1.31×10^{-2}	$[0.79, 1.8] \times 10^{-2}$	$[0.94, 1.8] \times 10^{-2}$	$[0.51, 2.1] \times 10^{-2}$
α_1	3.10×10^{-2}	$[2.1, 4.1] \times 10^{-2}$	$[2.4, 4.0] \times 10^{-2}$	$[2.0, 4.2] \times 10^{-2}$
α_2	5.21×10^{-2}	$[4.1, 6.4] \times 10^{-2}$	$[4.4, 6.3] \times 10^{-2}$	$[3.8, 6.7] \times 10^{-2}$
α_3	4.95×10^{-2}	$[4.2, 5.7] \times 10^{-2}$	$[4.4, 5.6] \times 10^{-2}$	$[4.1, 5.8] \times 10^{-2}$
α_4	2.94×10^{-2}	$[2.1, 3.7] \times 10^{-2}$	$[2.4, 3.5] \times 10^{-2}$	$[2.3, 3.6] \times 10^{-2}$
α_5	7.28×10^{-3}	$[0.24, 1.2] \times 10^{-2}$	$[0.42, 1.3] \times 10^{-2}$	$[0.32, 1.1] \times 10^{-2}$
α_6	2.26×10^{-2}	$[0, 5.5] \times 10^{-2}$	$[0.09, 5.8] \times 10^{-2}$	$[0, 4.8] \times 10^{-2}$
α_7	1.37	$[0, 6.8]$	$[0.016, \infty)$	$[0, 2.7]$
β_4	7.12×10^{-3}	$[0, 1.8] \times 10^{-2}$	$[0, 1.51] \times 10^{-2}$	$[0, 1.7] \times 10^{-2}$
β_5	2.69×10^{-2}	$[1.2, 4.2] \times 10^{-2}$	$[1.0, 3.8] \times 10^{-2}$	$[1.4, 3.9] \times 10^{-2}$
δ	4.52×10^{-2}	$[2.7, 6.4] \times 10^{-2}$	$[2.9, 6.0] \times 10^{-2}$	$[2.9, 6.1] \times 10^{-2}$

The general approach to characterize the reliability of parameter estimations is based upon evaluating their confidence intervals (CIs). There exist three major approaches to evaluate CIs: the technique based on the variance-covariance matrix [12], the profile-likelihood-based method [13] and the bootstrap method [14]. We applied these methods and assessed their relative performance by computing approximations to 95% confidence intervals for the estimated parameters, see Table 2, cf. [7]. During the last years statistical MCMC methods have been also advanced [5] to quantify uncertainty in model parameters. Below we apply the MCMC method to model (1).

4. MCMC toolbox

In statistics, MCMC methods are a class of algorithms for sampling from probability distributions based on constructing a Markov chain that has the desired distribution as its equilibrium distribution [6]. The state of the chain after a large number of steps is then used as a sample of the desired distribution. The quality of the sample improves as a function of the number of steps. The Metropolis-Hastings (MH) algorithm, implemented in the MCMC toolbox which we use [11], is a MCMC method, it generates a random walk using a proposal density and a method for rejecting proposed moves.

Since the MCMC Matlab toolbox [11] does not have a Manual, below we present some useful information collected from the code and the examples which can be downloaded with the code.

Proposal and prior distribution for the model parameters. The proposal distribution is the conditional probability of proposing a state x_{n+1} given x_n . The MCMC toolbox [11] uses multivariate Gaussian proposal distribution. The covariance matrix of the proposal distribution can be adapted during the simulation according to adaptive schemes described in [11, 15]. If the Hessian matrix can be easily computed and the best-fit values of the model parameters are known, we can compute the covariance matrix of the model parameters which is used (once) in the code as the initial covariance for the Gaussian proposal density for the MCMC samples. Otherwise, the initial covariance of the parameters is set to the prior covariance which is assigned by the user or is equal to infinity (as default in the code). In the latter case the initial proposal covariance for the parameter p_i is computed as $(p_i \times 0.05)^2$. The default prior distribution in the code is the Gaussian one. The user can assign prior means and standard deviations for all parameters. If they are not assigned, NaN, respectively, infinity are used as default. The user can also define his own prior distribution function.

How a new sample of the MCMC chain is computed in the code. First, a new candidate for all parameters is sampled from the Gaussian proposal. Then, using these values, new prior values are computed using Gaussian prior distribution (as default) with prior means and variances, or using a prior distribution, given by the user. These prior values are then checked whether they satisfy certain criterion(s). If so, then the new candidate is accepted and saved in the MCMC chain. If no, a certain algorithm (delayed rejection) is applied. If, after this, the new point is still not accepted, the previous sample is saved in the chain again (so a set of parameters is repeated twice in the chain).

Prior probability distribution for data. If the best-fit parameter values are known, we can compute and pass to the code the initial error variance. The user can also pass to the code the prior error variance. If the initial error variance is not defined, then it is taken equal to the prior error variance. If the latter is not defined, then the initial error variance is set by the code to 1 (default) and the prior variance is set equal to the initial error variance. If user sets the option "to update the error variance", then, for each sample of the chain, the variance is sampled as conjugate priors specified by the prior variance of the inverse gamma distribution, with the "noninformative" defaults equal to the prior values. Otherwise, the error variance is not updated

and the corresponding predictive envelope is not computed.

Posterior distribution. In Bayesian statistics, the posterior probability of a random event or an uncertain proposition is the conditional probability that is assigned after the relevant evidence is taken into account. Similarly, the posterior probability distribution is the distribution of an unknown quantity, treated as a random variable, conditional on the evidence obtained from an experiment or survey.

The output of the MCMC code includes, in particular, the posterior means and standard deviations of the parameters and a chain for the error variance (and, hence, error standard deviation). The predictive posterior distribution of the model: we can calculate the model fit for a randomly selected subset of the chain and calculate the predictive envelope of the model (e.g., 50%, 90%, 95%, 99% posterior regions).

One way to assess the accuracy of the posterior estimates is by calculating the Monte Carlo error for each parameter. This is an estimate of the difference between the mean of the sampled values (which we are using as our estimate of the posterior mean for each parameter) and the true posterior mean. As a rule of thumb, the simulation should be run until the Monte Carlo error for each parameter of interest is less than about 5% of the sample standard deviation.

Burn-in samples and acceptance rate. The Markov chain is started from an arbitrary initial value and the algorithm is run for many iterations until this initial state is "forgotten". These samples, which are discarded, are known as burn-in. The remaining set of accepted values (of the model parameters) represent a sample from the target (posterior) distribution.

The MH algorithm works best if the proposal density matches the shape of the target distribution from which direct sampling is difficult. If a Gaussian proposal density is used, the variance parameter σ^2 has to be tuned during the burn-in period. This is usually done by calculating the acceptance rate, which is the fraction of proposed samples that is accepted in a window of the last N samples. The desired acceptance rate depends on the target distribution, however it has been shown theoretically that the ideal acceptance rate for a one dimensional Gaussian distribution is about 50%, decreasing to about 23% for an N -dimensional Gaussian target distribution.

If σ^2 for parameters is too small the chain will mix slowly, i.e., the acceptance rate will be high but successive samples will move around the space slowly and the chain will converge only slowly to the posterior distribution. On the other hand, if σ^2 is too large the acceptance rate will be very low because the proposals are likely to land in regions of much lower probability density, the chain will converge very slowly. Ideally, the proposal distribution should have similar "shape" to posterior (target) distribution.

Kernel density. Kernel density estimation is one of the widely used non-parametric estimation techniques for estimating the probability density function of a random variable. For a univariate random variable X with unknown density $f(x)$, if we draw a sample of n independent and identically distributed observations x_1, x_2, \dots, x_n , the kernel density estimator is given by [16]

$$\hat{f}(x) = \frac{1}{n} \sum_{i=1}^n \frac{1}{h} K\left(\frac{x - x_i}{h}\right)$$

where h is the bandwidth that controls the amount of smoothness, and $K(\cdot)$ is the kernel function which is usually chosen to be a symmetric density function. In the MCMC code [11], the kernel function is the normal (Gaussian) density function.

What can do the MCMC toolbox. The MCMC code can do the following:

- Produce MCMC chain for user written $-2 * \log(\text{likelihood})$ and $-2 * \log(\text{prior})$ functions. These will be equal to sum-of-squares functions when using Gaussian likelihood and prior.
- In case of Gaussian error model, sample the model error variance from the inverse Gamma

distribution.

- Do plots and statistical analyses based on the chain, such as basic statistics, convergence diagnostics, chain timeseries plots, 2 dimensional clouds of points, kernel densities, and histograms.
- Calculate densities, cumulative distributions, quantiles, and random variates for some useful common statistical distributions without using Mathworks own statistics toolbox.

5. Results of the MCMC code applied to model (1)

The user has to define, at least, two functions, to run the MCMC code: the one to compute the model solution and another one to compute the objective function. Since the model solution in our case is given analytically (2), we compute the objective function (sum-of-squares) by (4). If a minimum of the objective function $\Phi(\mathbf{p}^*)$ is computed (e.g. by the `fminsearch` code), the estimate for the error variance σ^2 is obtained as

$$mse = \frac{\Phi(\mathbf{p}^*)}{n_d - n_p}, \tag{7}$$

where, in our case, $n_d = 36$ is the number of the data points and $n_p = 11$ is the number of the estimated parameters. In our case $mse \approx 5.07 \times 10^7$ and $\sqrt{(mse)} \approx 7.12 \times 10^3$. Then, we compute the covariance of the estimated parameters which is further used as the initial proposal covariance for the MCMC samples,

$$tcov = 2 \times H^{-1}(\mathbf{p}^*) \times mse, \tag{8}$$

where H is the Hessian matrix,

$$H(\mathbf{p}) := \left\{ \frac{\partial}{\partial \mathbf{p}} \right\} \left\{ \frac{\partial}{\partial \mathbf{p}} \right\}^T \Phi(\mathbf{p}) \in \mathbb{R}^{n_p \times n_p}, \quad H_{k,m}(\mathbf{p}) = \frac{\partial^2}{\partial p_k \partial p_m} \Phi(\mathbf{p}), \tag{9}$$

with $H_{k,m}$ being the (k, m) -th element of H . Note that the Hessian $H(\mathbf{p}^*)$ is not singular here since we set all "problematic" parameters β to zero. We use an exact Hessian $H(\mathbf{p}^*)$ since we have a code to compute analytically the first and second derivatives of the objective with respect to the estimated parameters.

We set the number of simulations to 20000, allow automatic sampling and estimation of the error variance, set the initial covariance for the Gaussian proposal density of the MCMC sampler to $tcov$ and we use the best-fit parameter values as the starting point for the MCMC chain. As a result of the MCMC simulation run, we obtain a structure that contains some information about the run and matrices that contain the actual MCMC chains for the parameters and for the observation error variance. Figure 2 shows the one-dimensional parameter chains. We used the positivity condition on the parameter values. The acceptance rate is about 81%. One can conclude that after the burn-in period (not shown) all the parameters, except α_7 , converge.

For convergence diagnostics we computed the Geweke criterion. Geweke's MCMC convergence diagnostic is a test for equality of the means of the first $a\%$ and last $b\%$ of a Markov chain [17]. We take the default values, $a = 10\%$ and $b = 50\%$, as Geweke suggested. The corresponding code gives vectors $z, c \in \mathbb{R}^{n_p}$,

$$z = [-10, 6.7, -6.7, -0.30, -0.56, 2.5, 0.91, -197, -5.6, -2.9, -5.7] \times 10^{-2},$$

$$c = [0.915, 0.946, 0.946, 0.998, 0.996, 0.980, 0.993, 0.049, 0.955, 0.977, 0.955].$$

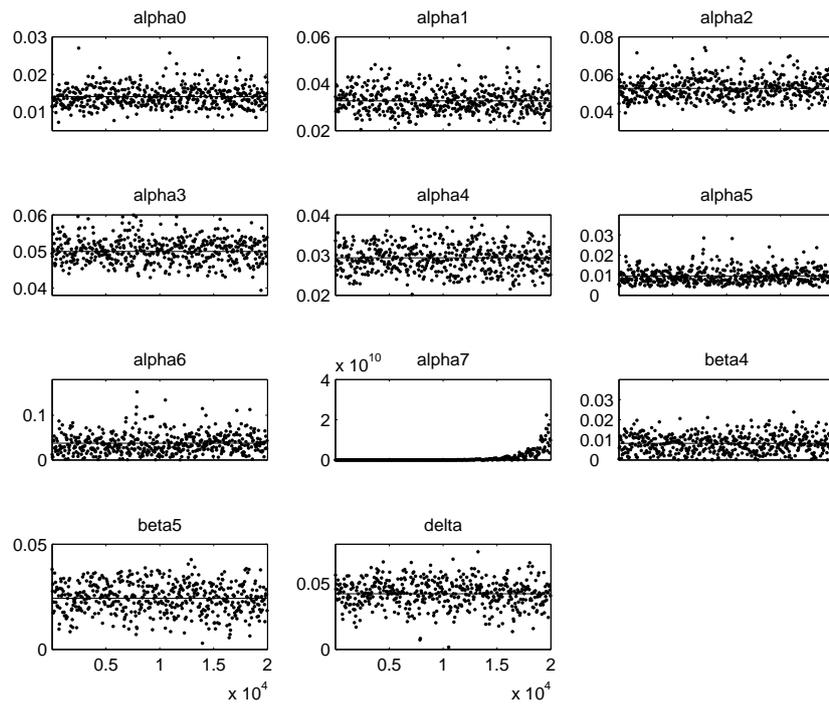


Fig. 2. One dimensional parameter chains. Curves indicate the mean value of the parameters.

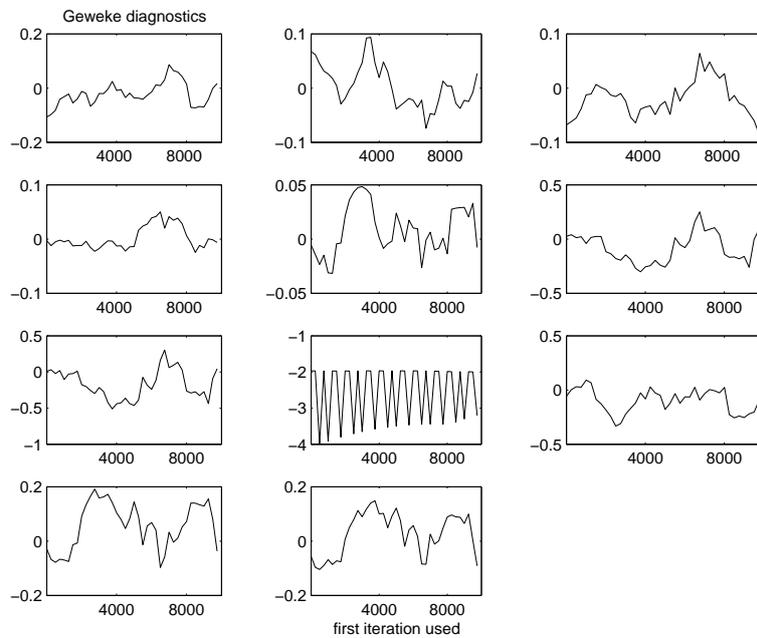


Fig. 3. Geweke's diagnostic. The first row: $\alpha_0, \alpha_1, \alpha_2$, the second row: $\alpha_3, \alpha_4, \alpha_5$, the third row: $\alpha_6, \alpha_7, \beta_4$, the fourth row: β_5, δ .

Table 3. MCMC statistics, number of simulations (samples) is equal to 20000

parameter	mean	std	MC error	geweke
α_0	0.014058	0.0026337	0.00010853	0.91519
α_1	0.032698	0.0050817	0.00022892	0.9464
α_2	0.052512	0.0055983	0.00024943	0.94594
α_3	0.05023	0.0033669	0.00015214	0.99762
α_4	0.029199	0.0032575	0.00014038	0.99551
α_5	0.00963	0.0036393	0.00016458	0.9799
α_6	0.037262	0.022393	0.0010226	0.99274
α_7	9.3479e+008	2.5245e+009	5.1731e+008	0.048973
β_4	0.0082371	0.0043493	0.00017712	0.95503
β_5	0.024307	0.0070822	0.00030413	0.97689
δ	0.042339	0.010258	0.00043598	0.9549

The order of the elements of these vectors corresponds to the order of parameters in the vector \mathbf{p} (6). The meaning of these vectors is the following. If the chain converges, then z_n tends to $\mathcal{N}(0, 1)$ as $n \rightarrow \infty$, where n is the number of samples and the values of a and b are fixed. We can use this result to test the null hypothesis of equal location, which, if it is rejected (i.e., $|Z_n|$ is large), indicates that the chain has not converged by time n_0 [17]. We observe that the largest element of the vector z corresponds to the parameters α_7 , which, as we obtained earlier, has a very large or infinite 95% confidence interval. In Fig. 3 we plot Geweke’s diagnostic z , which shows how this diagnostic satisfies $\mathcal{N}(0, 1)$ for all parameters, except α_7 . These plots are given by the code. We observe that the diagnostic z is closer to $\mathcal{N}(0, 1)$, except the parameter α_7 .

In the code, the vector c is computed as

$$c = 2 \times (1 - \text{normcdf}(\text{abs}(z))),$$

its value is given in Table 3 (last column). Here *normcdf* is the standard normal (Gaussian) cumulative distribution. Ideally, all components of the vector c should be equal to 1. Again, the vector c shows that its element corresponding to α_7 is equal to 0.049, what implies that α_7 does not converge.

Table 3 presents mean and standard deviation (std) from the chain, the Monte Carlo error estimates (MC error) and a simple test (geweke) for a null hypothesis that the chain has converged. Note that the Monte Carlo error for each parameter, except α_7 , is about 5% of the corresponding standard deviation or less.

Some of the estimates of 95% confidence intervals, computed using the mean value and the standard deviation (see Table 3), are more close to ones computed by the variance-covariance approach, while others are close to the ones computed by the profile-likelihood method and by the parametric bootstrap technique:

$$\alpha_0 \in [0.94, 1.9] \times 10^{-2}, \alpha_1 \in [2.2, 4.3] \times 10^{-2}, \alpha_2 \in [4.1, 6.4] \times 10^{-2},$$

$$\alpha_3 \in [4.3, 5.7] \times 10^{-2}, \alpha_4 \in [2.6, 3.6] \times 10^{-2}, \alpha_5 \in [0.21, 1.7] \times 10^{-2},$$

$$\alpha_6 \in [0, 8.3] \times 10^{-2}, \alpha_7 \in [0, 61.3] \times 10^9, \beta_4 \in [0, 1.7] \times 10^{-2},$$

$$\beta_5 \in [0.97, 3.9] \times 10^{-2}, \delta \in [2.1, 6.4] \times 10^{-2}.$$

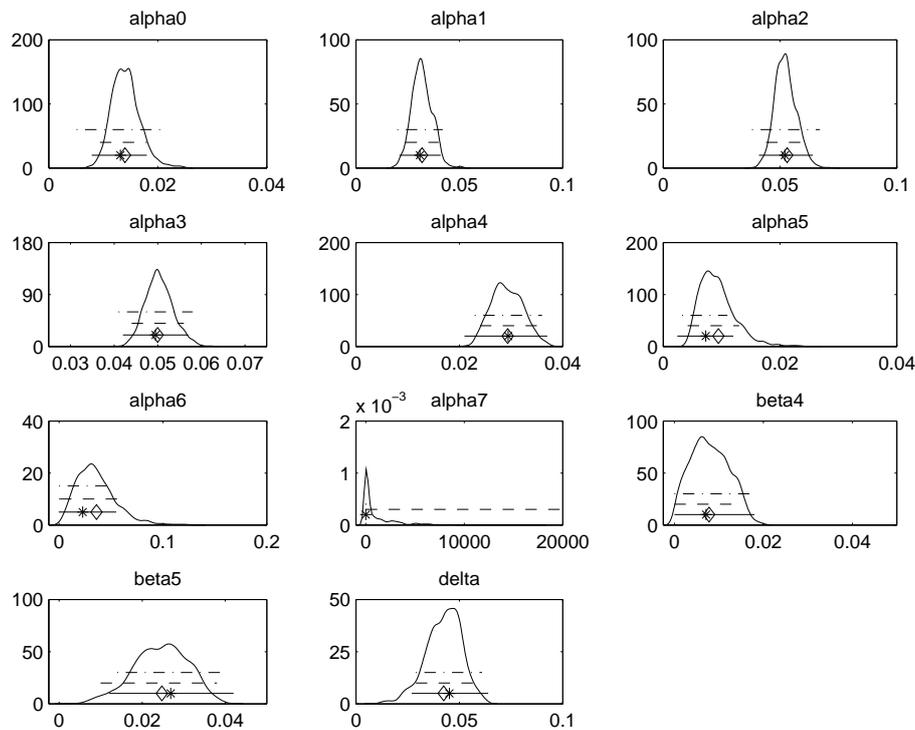


Fig. 4. The computed posterior probability density functions of the model parameters. The best-fit values of the model parameters, cf. Table 2, are indicated by (*), the means of the parameter chains computed by MCMC are indicated by (◊). The CIs computed by the variance-covariance method (solid lines), the profile-likelihood approach (dashed lines) and by the parametric bootstrapping (dot-dashed lines) are depicted (see Table 2).

Figure 4 shows the posterior probability density functions of the model parameters, computed from the chain values using kernel density estimator. Note that a very small left part of density curves for the parameters $\alpha_6, \alpha_7, \beta_4$ is located to the left from zero. In this figure, we also plotted, for each parameter, its best-fit value (cf. Table 2) with the corresponding CIs computed by the variance-covariance method, the profile-likelihood approach and by the parametric bootstrapping (see Table 2) and its mean value computed from the MCMC chain. We observe that the best-fit values of the parameters are very close to the mean values or coincide with the means.

Examples of two dimensional marginal posterior distributions for some of the model parameters are presented in Fig. 5. In this way we can directly examine parameter correlation. Note that all pairs of correlated parameters are depicted in this figure: $(\alpha_5, \alpha_6), (\alpha_i, \beta_i, i = 4, 5), (\alpha_{i-1}, \beta_i, i = 4, 5),$ and $(\alpha_6, \beta_5), (\beta_4, \beta_5), (\beta_5, \delta)$. This correlation is explained by the structure of the model (the model is linear and some parameters α_i and β_i compensate each other) and by a small number of the experimental data, especially for last generations of cells. As an example, in this figure are given non-correlated parameters: $(\alpha_0, \alpha_i, i = 3, \dots, 7), (\alpha_3, \beta_5), (\alpha_5, \beta_4)$ and (β_4, δ) .

Figure 6 shows the histogram of the chain for the error posterior standard deviation. Recall that we obtained, using (7), $\sqrt{(mse)} \approx 7.12 \times 10^3$. The mse value was used as the initial error variance to start the MCMC run.

Figure 7 presents (a) the best-fit of the given data using the posterior means of the model parameters and (b) the 95% posterior probability limits of the model solution uncertainty and predicted observations. Note that data for N_7 are not fitted. This is because there is no convergence for α_7 . Note also that, when plotting all the predictive envelopes in the same scale, we observe almost the same size of the envelopes (not shown).

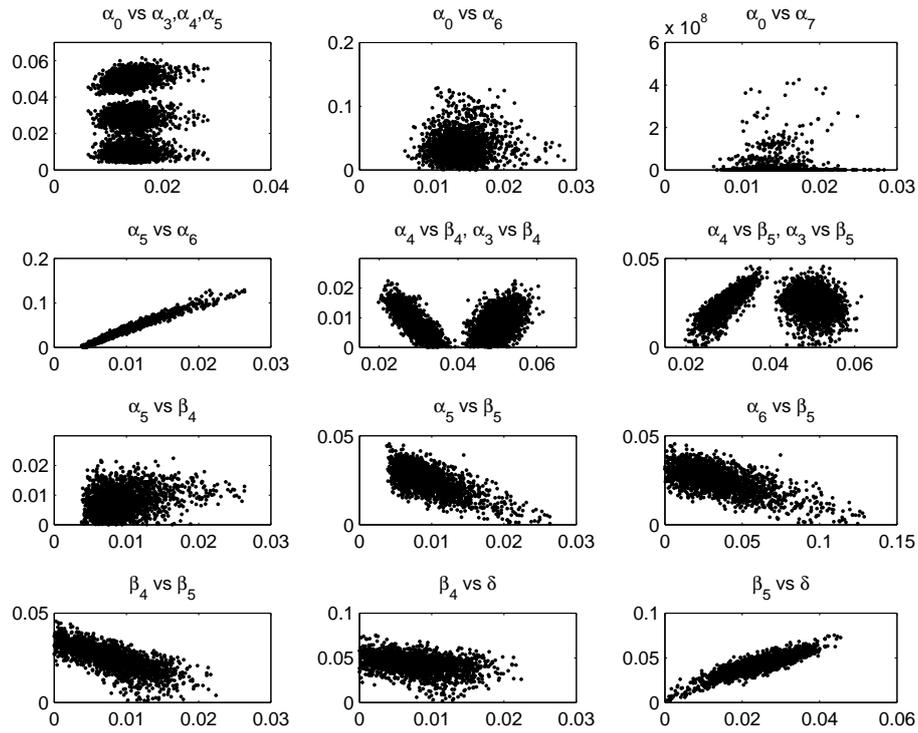


Fig. 5. Two dimensional marginal posterior distributions for some of the model parameters. The computed chain consists of 20000 samples, only every 10th sample is plotted.

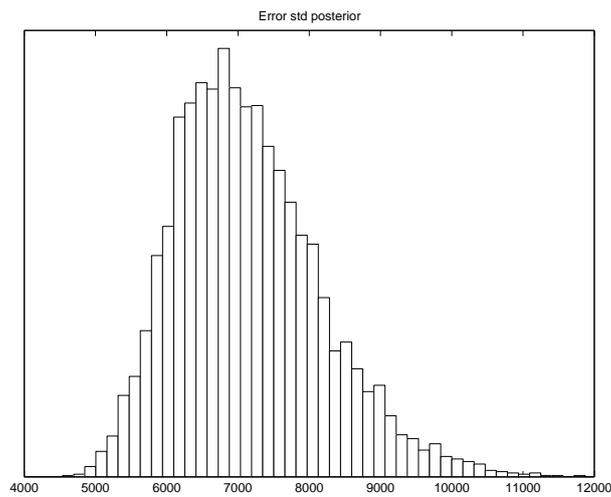


Fig. 6. The histogram of the chain for posterior error standard deviation.

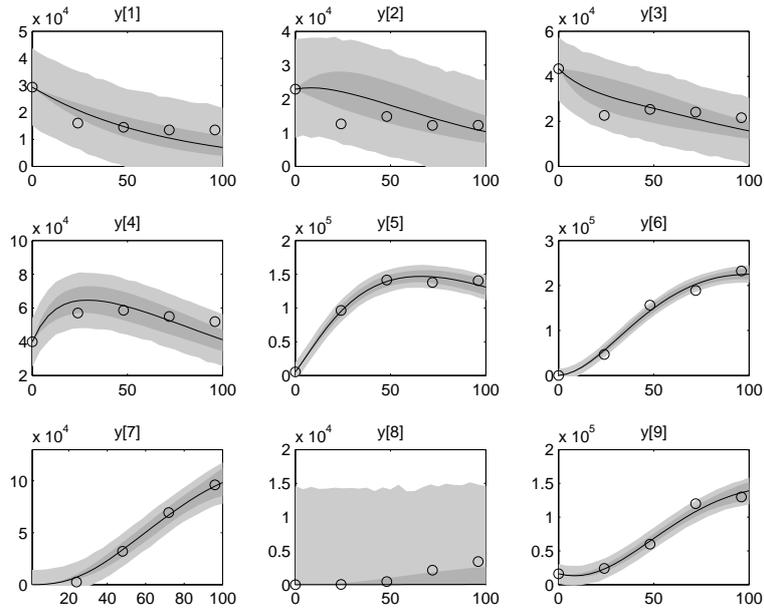


Fig. 7. Plots of the data-fitted model solutions together with the uncertainties. Circles present the data. The solid lines show the median fits (the posterior means of the model parameters). Darker areas correspond to the 95% posterior probability limits of the model uncertainty, while the lighter areas present the same uncertainty level in predicting new observations. The time scale is shifted by 72 hours. The variables $y(1), y(2), \dots, y(9)$ imply N_0, N_1, \dots, N_7, D .

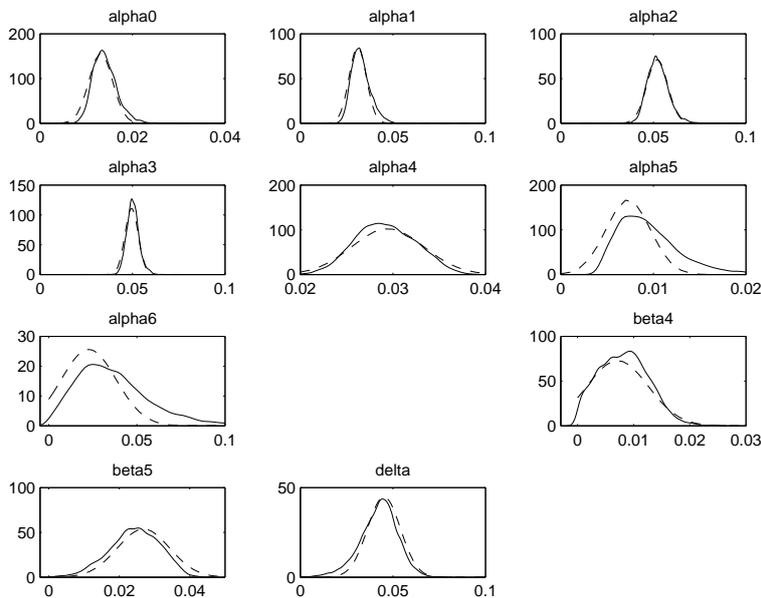


Fig. 8. Solid curves indicate the posterior probability density functions of the model parameters computed from the MCMC chain using kernel density estimate. Dashed curves indicate normal probability density functions computed using the best-fit parameter estimates and their variances, cf. Table 2.

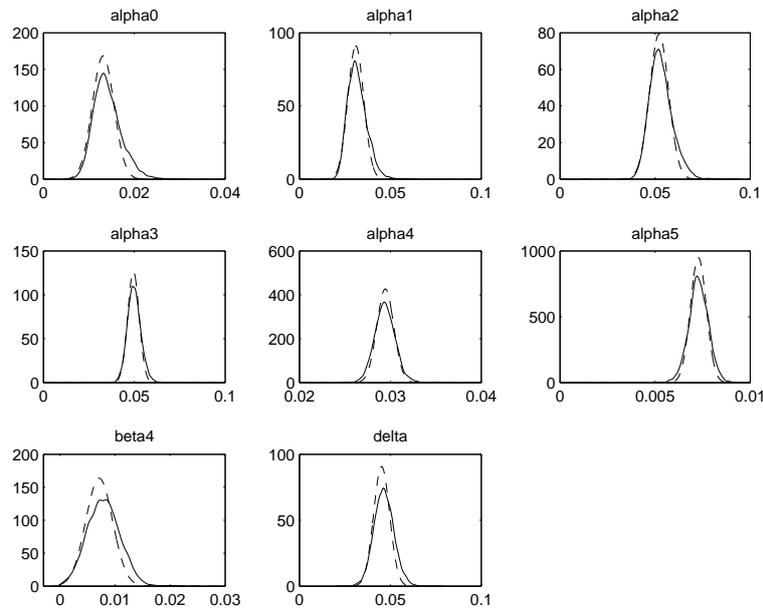


Fig. 9. Solid curves indicate the posterior probability density functions of the model parameters computed from the MCMC chain using kernel density estimate. Dashed curves indicate normal probability density functions computed using the best-fit parameter estimates and their variances, cf. Table 2.

We also investigated, fixing the non-identifiable parameter α_7 at some value, whether the posterior probability density functions are close to the Gaussian ones. Figure 8 shows, for each model parameter, except α_7 , (a) the posterior probability density functions computed from the MCMC chain using kernel density estimate and (b) normal probability density functions computed using the best-fit parameter estimates and their variances, cf. Table 2. We do not show here results for α_7 since, as we obtained above, the chain for this parameter does not converge and hence the MCMC estimation results for this parameter are not reliable. We see that the densities computed by the two methods are centered at the same location for all parameters. The estimated densities almost coincide for all parameters except α_5 , α_6 , β_4 . This implies that the posterior densities are Gaussian for most of the model parameters and they are close to Gaussian ones for the parameters α_5 , α_6 , β_4 .

As Fig. 5 indicates, if we fix parameters α_6 , β_5 and α_7 , then we can expect that correlation between the remaining model parameters will be minimal. So we fix these three parameters at their best-fit values, cf. Table 2, and compute the posterior probability density for the remaining parameters, see Fig. 9. We observe that now the posterior density for α_5 and β_4 are much closer to the Gaussian one.

6. Conclusions

The statistical MCMC method, being implemented as the Matlab toolbox [11], allows one to get much more information about the model behavior and the data: the posterior probability density functions of the model parameters and the posterior probability limits of the model uncertainty and uncertainty in predicting new observations. It also allows one to get easily two dimensional marginal posterior distributions for the model parameters and in this way directly examine parameter correlation.

As our particular example on the cell population dynamics model shows, some of the confidence intervals computed by the MCMC method are more close to ones computed by the variance-covariance approach, while others are close to the ones computed by the profile-likelihood method and by the bootstrap technique.

Our experience with the MCMC code suggests that an efficient methodology to proceed with the application of the MCMC analysis should consist of the following stages

- 1 compute the minimum of the data mismatch expressing objective function by, e.g., the fminsearch code, and
- 2 use the MCMC method to compute (i) posterior probability distributions for the estimated parameters and (ii) the 95% posterior probability limits for the model parameters uncertainty and predicted observations.

In this setting (i) we sequentially compute the initial estimate for the error variance (*mse*) to start the code with and (ii) thus, avoid the need the Hessian matrix because the code creates an initial proposal covariance for the MCMC method, using the best-fit values of the model parameters. Overall, the use of MCMC method for the identification of the cell growth models allows one to comprehensively explore the information about the model parameters and in parallel to express the uncertainty in the estimates without the need for a prior analysis of the model parameter identifiability. The method greatly extends the set of characteristics on immune cell turnover which can be extracted from the labelling assays using ODE models.

7. Acknowledgments

This work was supported by the Russian Foundation for Basic Research (Grant number 17-01-00636) to G.B.

REFERENCES

1. Muller V., Maree A.F., De Boer R.J. Small variations in multiple parameters account for wide variations in HIV-1 set-points: a novel modelling approach. *Proc. R. Soc. Lond. B.* 2001. V. 268. P. 235–242.
2. De Boer R.J., Perelson A.S. Quantifying T lymphocyte turnover. *J. Theor. Biol.* 2013. V. 327. P. 45–87.
3. Hross S., Hasenauer J. Analysis of CFSE time-series data using division-,age- and label-structured population models. *Bioinformatics.* 2016. V. 32. No. 15. P. 2321–2329.
4. Kenz Z.R., Banks H.T., Smith R.C. Comparison of frequentist and Bayesian confidence analysis methods on a viscoelastic stenosis model. *SIAM/ASA J. on Uncertainty Quantification.* 2013. V. 1. No. 1. P. 348–369.
5. Ballnus B., Hug S., Hatz K., Grlitz L., Hasenauer J., Theis F.J. Comprehensive benchmarking of Markov chain Monte Carlo methods for dynamical systems. *BMC Syst. Biol.* 2017. V. 11. No. 63. P. 1–18.
6. Samaniego F.J. *A comparison of the Bayesian and frequentist approaches to estimation.* New York: Springer-Verlag, 2010. (Springer Series in Statistics).
7. Luzyanina T., Mrusek S., Edward J.T., Roose D., Ehl S., Bocharov G. Computational analysis of CFSE proliferation assay. *J. of Math. Biology.* 2007. V. 54. No. 1. P. 57–89.
8. Klinke D.J. An empirical Bayesian approach for model-based inference of cellular signaling networks. *BMC Bioinformatics.* 2009. V. 10. P. 371–387.
9. Solonen A., Ollinaho P., Laine M., Haario H., Tamminen J., Jarvinen H. Efficient MCMC for climate model parameter estimation: Parallel adaptive chains and early rejection. *Bayesian Analysis.* 2012. V. 7. No. 3. P. 715–736.
10. Hug S., Raue A., Hasenauer J., Klingmuller U., Timmer J., Theis F.J. High-dimensional Bayesian parameter estimation: Case study for a model of JAK2/STAT5 signaling. *Math. Biosciences.* 2013. V. 246. P. 293–304.
11. Haario H., Laine M., Mira A., Saksman E. DRAM: Efficient adaptive MCMC. *Stat. Comput.* 2006. V. 16. P. 539–354.

12. Bard Y. *Nonlinear parameter estimation*. New York: Academic Press, 1974.
13. Venzon D.J., Moolgavkar S.H. A method for computing profile-likelihood-based confidence intervals. *Appl. Statist.* 1988. V. 37. No. 1. P. 87–94.
14. Efron B., Tibshirani R. Bootstrap methods for standard errors, confidence intervals, and other measures of statistical accuracy. *Stat. Sci.* 1986. V. 1. No. 1. P. 54–77.
15. Haario H., Saksman E., Tamminen J. An adaptive Metropolis algorithm. *Bernoulli*. 2001. V. 7. No. 2. P. 223–242.
16. Wand M.P., Jones M.C. *Kernel smoothing*. London: Chapman & Hall, 1995
17. Brooks P.S., Roberts G.O. Assessing convergence of Markov chain Monte Carlo algorithms. *Stat. Comput.* 1998. V. 8. P. 319–335.

Accepted 14.09.2018.

Published 03.10.2018.