

MCMC_CLIB—an advanced MCMC sampling package for ODE models

Andrei Kramer^{1,*}, Vassilios Stathopoulos², Mark Girolami³ and Nicole Radde¹

¹Institute for Systems Theory and Automatic Control, University of Stuttgart, 70569 Stuttgart, Germany, ²Department of Statistical Science, University College, London WC1E 6BT, UK and ³Department of Statistics, University of Warwick, Coventry CV4 7AL, UK

Associate Editor: Jonathan Wren

ABSTRACT

Summary: We present a new C implementation of an advanced Markov chain Monte Carlo (MCMC) method for the sampling of ordinary differential equation (ODE) model parameters. The software MCMC_CLIB uses the simplified manifold Metropolis-adjusted Langevin algorithm (SMMALA), which is locally adaptive; it uses the parameter manifold's geometry (the Fisher information) to make efficient moves. This adaptation does not diminish with MC length, which is highly advantageous compared with adaptive Metropolis techniques when the parameters have large correlations and/or posteriors substantially differ from multivariate Gaussians. The software is standalone (not a toolbox), though dependencies include the GNU scientific library and sundials libraries for ODE integration and sensitivity analysis.

Availability and implementation: The source code and binary files are freely available for download at http://a-kramer.github.io/mcmc_clib/. This also includes example files and data. A detailed documentation, an example model and user manual are provided with the software.

Contact: andrei.kramer@ist.uni-stuttgart.de

Received on February 2, 2014; revised on June 28, 2014; accepted on June 30, 2014

1 INTRODUCTION

Modeling intracellular reaction networks by ordinary differential equations (ODEs) has become a standard approach in systems biology. ODEs provide a quantitative and dynamic description of the underlying biochemical processes. However, parameter estimation for these models is a challenging task, as typically only few data points are available, and the respective optimization problem is ill-posed. That is, the data do not contain sufficient information to identify parameter values uniquely. In a statistical Bayesian framework, this results in high correlations of parameters and/or multiple modes in the posterior distribution. Standard sampling algorithms frequently fail when facing those problems. The Metropolis algorithm in its original form (Hastings, 1970; Metropolis *et al.*, 1953), for example, is elegant and easy to implement but uses an isotropic distribution to suggest the next set of parameters and struggles with complicated posteriors. In the case of strongly correlated parameters, some directions are highly prohibitive because of low likelihoods, whereas others are permissive because of low sensitivity of the model output in these directions, i.e. flat likelihoods. This results in highly auto-correlated samples and extremely slow convergence rates. Adaptive variants of Metropolis that try to amend

this problem do exist; they use diminishing adaptation that can deal with strong correlations of sampling variables in some cases. One such implementation is described in Haario *et al.*, 2006. The adaptive Metropolis algorithm adapts to the posterior by estimating its covariance from previously sampled points. While this can drastically increase sampling efficiency for densities that are similar to Gaussians, it struggles with more complex cases. To date, sampling techniques that put more computational effort into the proposal step are often implemented in high-level programming languages only and so are not efficient computationally. This prompted our efforts to provide a highly efficient specialized tool that spends most of its running time on the solution of initial value problems.

We introduce MCMC_CLIB, an implementation of SMMALA (simplified manifold Metropolis-adjusted Langevin algorithm), a sampling algorithm, described in Girolami and Calderhead, 2011 that has been shown to outperform simpler sampling algorithms.

2 APPROACH

Our approach is motivated by modeling practices in systems biology, where model parameters ρ_k are typically required to be of fixed sign for model stability. Otherwise, the model structure is fairly generic. Although the software will always supply $\rho_k = \exp(\theta_k) > 0$ to the model (see Availability and Implementation section), the framework can be manipulated to accommodate sign switching. Using θ for sampling, i.e. sampling in logarithmic space, has the benefit of easily covering several orders of magnitude. An important novel feature is the ability of the implemented likelihood function to deal with relative data like Western blots in arbitrary units.

2.1 Modeling

We assume an ODE model given in state-space representation:

$$\dot{x} = f(x; \rho, u_k), \quad x \in \mathbb{R}^n, \quad \rho \in \mathbb{R}_+^r, \quad (1)$$

$$y_{jk} = Cx(t_j; \rho, u_k) + \delta_{jk}, \quad y_{jk} \in \mathbb{R}^m, \quad (2)$$

with additive Gaussian noise $\delta_{jk} \in \mathbb{R}^m \sim \mathcal{N}(0, \Sigma_{jk})$. We assume the relation between state and measurement to be linear, characterized by $C \in \mathbb{R}^{m \times n}$ (it may, however, be scaled by an unknown parameter, i.e. data may be in arbitrary units). Index $k = 1, \dots, n_E$ enumerates different experiments (described by different inputs $u_k \in \mathbb{R}^q$, e.g. drug dosage), whereas j marks different measurement time instances. For simplicity, we assume

*To whom correspondence should be addressed.

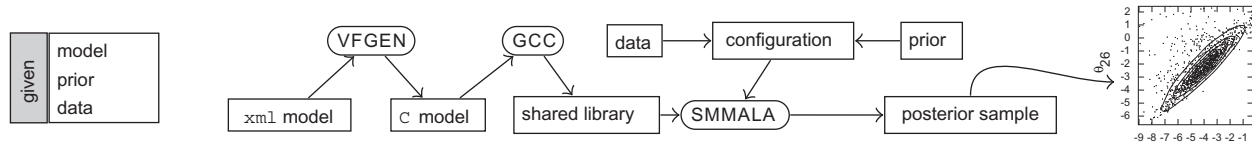


Fig. 1. Flowchart of the usage structure

independent noise with standard deviation σ_{ijk} for each experiment, each output variable and each time point, such that $\Sigma_{jk} = \text{diag}(\sigma_{jk})$ contains variances of outputs y_{ijk} ($i = 1, \dots, m$). Given data $D = \{y_{ijk}, \sigma_{ijk}^2\}_{i,j,k}$, the likelihood reads

$$L_D(\theta) = \exp\left(-\frac{1}{2} \sum_{i,j,k} \left(\frac{y_{ijk} - \sum_l C_{il} x_l(t_j; \theta, u_k)}{\sigma_{ijk}} \right)^2\right). \quad (3)$$

Using Bayesian learning, we construct the posterior distribution $p(\theta|D) \propto L_D(\theta)p(\theta)$ with prior distribution $p(\theta) \sim \mathcal{N}(\mu, \Xi)$ that is implemented as a multivariate Gaussian distribution with mean μ and covariance matrix Ξ .

3 METHODS

The original and the SMMALA algorithms are comprehensively described in Girolami and Calderhead, 2011 and Calderhead, 2012, respectively, and we only give a glimpse of its structure here.

3.1 The simplified manifold MALA algorithm SMMALA

The Markov chain (MC) is a discrete time approximation of the SDE

$$d\theta = \frac{1}{2} G^{-1}(\theta) \nabla_{\theta} V(\theta; D) dt + \text{chol}(G^{-1}(\theta)) db(t), \quad (4)$$

where $V(\theta; D) = \log(p(\theta|D))$ and $db(t)$ is a Wiener process. The natural metric $G(\theta)$ of the parameter space is the Fisher information; chol denotes the Cholesky decomposition, a function that returns the Cholesky factor. We can write the MC proposal $\theta \rightarrow \theta'$ with step size $\epsilon \in \mathbb{R}_+$ as

$$p(\theta'|\theta) = \mathcal{N}(\mathbf{v}(\theta, \epsilon), M(\theta, \epsilon)) \quad (5)$$

$$\mathbf{v}(\theta, \epsilon) = \theta + \frac{\epsilon}{2} G^{-1}(\theta) \nabla_{\theta} V(\theta; D) \quad (6)$$

$$M(\theta, \epsilon) = \epsilon^2 G^{-1}(\theta). \quad (7)$$

This step proposes a parameter vector from a Gaussian distribution with mean and covariance that take information about the local shape of the posterior distribution via the Fisher matrix and the gradient into account. The metric tensor G is calculated using sensitivity analysis ($S_y := dC\mathbf{x}/d\theta$) of the solution to the initial value problem,

$$G(\theta) = \sum_{j,k} S_y(t_j; \theta, \mathbf{u}_k)^T \Sigma_{jk}^{-1} S_y(t_j; \theta, \mathbf{u}_k). \quad (8)$$

3.2 Implementation details

Figure 1 shows a flowchart of the typical MCMC_CLIB usage.

Input The ODE model has to be provided as an XML file in VFGENS vf format, together with a data file that contains measurement time points and a data set D . Additionally, this data file also specifies sample size, initial step size, desired acceptance rate and the parameters μ and Ξ^{-1} for the prior distribution.

Sampling algorithm To evaluate the likelihood function, ODEs are numerically integrated using CVODES with fixed relative and absolute tolerance. For further details, we refer to the supplement in the Availability and Implementation section (manual.pdf and documentation.pdf).

Table 1. Sample properties for the provided example model

Property	Value
Sample size	$N = 1 \times 10^6$
Target acceptance	0.500
Observed acceptance	0.498
Auto-correlation length $\tau_{\text{int},1}$	340 ± 30
Sampling time t_s	74 h
Effective sampling speed v	$(5.50 \pm 0.49) \times 10^{-3} \text{ s}^{-1}$

Note: The effective sampling speed is defined as $v = N/(2\tau_{\text{int},1}t_s)$.

Output The user can select a text or binary file output, which contains the results of the sampling procedure: the sampled log-parameters and their log-posterior values. The sample can be further processed with other numerical software (e.g. GNU OCTAVE, MATLAB).

3.3 Test results for the provided demo

For demonstration purposes, we supply a model with 11 state variables, 26 parameters and 4 input variables. The data, obtained in a real experiment, are given in arbitrary units, stem from $n_E = 4$ experiments and are only meaningful in relation to the reference experiment (also included). The initial conditions are unknown stable steady states of the unperturbed system. At $t = 0$, a constant disturbance, parametrized by \mathbf{u} , is activated and perturbs the system indefinitely. Then measurements are taken. The activation is modeled by a logistic function centered at $t = 0$.

Performance of the sampling algorithm is summarized in Table 1. Further insight into this example is provided as part of the software package. The analysis reveals that not all data points can be fitted equally well by the model; a considerable uncertainty remains for several parameters and consequently model predictions. Hence, we consider this as a realistic test case, which covers many of the difficulties that emerge in current research.

Funding: A.K. and N.R. acknowledge support by the German Research Foundation (DFG) within the Cluster of Excellence in Simulation Technology (EXC 310/1) at the University of Stuttgart.

Conflict of interest: none declared.

REFERENCES

- Calderhead, B. (2012) Differential geometric MCMC methods and applications. PhD Thesis, Department of Computer Science, University of Glasgow, Glasgow, UK.
- Girolami, M. and Calderhead, B. (2011) Riemann manifold Langevin and Hamiltonian Monte Carlo methods. *J. R. Stat. Soc. B*, **73**, 123–214.
- Haario, H. et al. (2006) DRAM: efficient adaptive MCMC. *Stat. Comput.*, **16**, 339–354.
- Hastings, W.K. (1970) Monte Carlo sampling methods using Markov chain and their applications. *Biometrika*, **57**, 97–109.
- Metropolis, N. et al. (1953) Equation of state calculations by fast computing machines. *J. Chem. Phys.*, **21**, 1087–1092.