



# CHALMERS

## Chalmers Publication Library

### **A continuous-time adaptive particle filter for estimations under measurement time uncertainties with an application to a plasma-leucine mixed effects model.**

This document has been downloaded from Chalmers Publication Library (CPL). It is the author's version of a work that was accepted for publication in:

**BMC systems biology (ISSN: 1752-0509)**

Citation for the published paper:

Krengel, A. ; Hauth, J. ; Taskinen, M. (2013) "A continuous-time adaptive particle filter for estimations under measurement time uncertainties with an application to a plasma-leucine mixed effects model.". BMC systems biology, vol. 7(1), pp. 8- (53 sidor).

<http://dx.doi.org/10.1186/1752-0509-7-8>

Downloaded from: <http://publications.lib.chalmers.se/publication/175027>

Notice: Changes introduced as a result of publishing processes such as copy-editing and formatting may not be reflected in this document. For a definitive version of this work, please refer to the published source. Please note that access to the published version might require a subscription.

Chalmers Publication Library (CPL) offers the possibility of retrieving research publications produced at Chalmers University of Technology. It covers all types of publications: articles, dissertations, licentiate theses, masters theses, conference papers, reports etc. Since 2006 it is the official tool for Chalmers official publication statistics. To ensure that Chalmers research results are disseminated as widely as possible, an Open Access Policy has been adopted. The CPL service is administrated and maintained by Chalmers Library.

(article starts on next page)

METHODOLOGY ARTICLE

Open Access

# A continuous-time adaptive particle filter for estimations under measurement time uncertainties with an application to a plasma-leucine mixed effects model

Annette Krengel<sup>1\*</sup>, Jan Hauth<sup>1</sup>, Marja-Riitta Taskinen<sup>2</sup>, Martin Adiels<sup>3,4</sup> and Mats Jirstrand<sup>5</sup>

## Abstract

**Background:** When mathematical modelling is applied to many different application areas, a common task is the estimation of states and parameters based on measurements. With this kind of inference making, uncertainties in the time when the measurements have been taken are often neglected, but especially in applications taken from the life sciences, this kind of errors can considerably influence the estimation results. As an example in the context of personalized medicine, the model-based assessment of the effectiveness of drugs is becoming to play an important role. Systems biology may help here by providing good pharmacokinetic and pharmacodynamic (PK/PD) models. Inference on these systems based on data gained from clinical studies with several patient groups becomes a major challenge. Particle filters are a promising approach to tackle these difficulties but are by itself not ready to handle uncertainties in measurement times.

**Results:** In this article, we describe a variant of the standard particle filter (PF) algorithm which allows state and parameter estimation with the inclusion of measurement time uncertainties (MTU). The modified particle filter, which we call MTU-PF, also allows the application of an adaptive stepsize choice in the time-continuous case to avoid degeneracy problems. The modification is based on the model assumption of uncertain measurement times. While the assumption of randomness in the measurements themselves is common, the corresponding measurement times are generally taken as deterministic and exactly known. Especially in cases where the data are gained from measurements on blood or tissue samples, a relatively high uncertainty in the true measurement time seems to be a natural assumption. Our method is appropriate in cases where relatively few data are used from a relatively large number of groups or individuals, which introduce mixed effects in the model. This is a typical setting of clinical studies. We demonstrate the method on a small artificial example and apply it to a mixed effects model of plasma-leucine kinetics with data from a clinical study which included 34 patients.

**Conclusions:** Comparisons of our MTU-PF with the standard PF and with an alternative Maximum Likelihood estimation method on the small artificial example clearly show that the MTU-PF obtains better estimations. Considering the application to the data from the clinical study, the MTU-PF shows a similar performance with respect to the quality of estimated parameters compared with the standard particle filter, but besides that, the MTU algorithm shows to be less prone to degeneration than the standard particle filter.

**Keywords:** Particle filter, Sequential Monte Carlo methods, Nonlinear filtering, Parameter estimation, Measurement time uncertainties, PK/PD, Mixed effects, Leucine kinetics

\*Correspondence: krengel@mathematik.uni-kl.de

<sup>1</sup>Fraunhofer-Institut für Techno- und Wirtschaftsmathematik (ITWM);  
Fraunhofer Institute for Industrial Mathematics, Kaiserslautern, Germany  
Full list of author information is available at the end of the article

## Background

### Measurement time uncertainties

Uncertainty in the time at which a measurement is taken is an often neglected source of random error. While in many application areas, this kind of error is generally small and indeed neglectable (due to automated measurements and precise timings), in others it may be of real influence, especially in the life sciences. As a prominent example, one may consider pharmacokinetic and pharmacodynamic (PK/PD) models which are used to describe the metabolic interactions and the effects of a chemical agent (like a drug or a labelled substance) over time inside an organism, respectively.

A typical population experiment in the PK/PD context consists in the analysis of the contents of the blood plasma of several individuals with respect to concentrations of certain molecules of interest. For this purpose, blood probes have to be taken from each individual at certain (fixed) time points after a certain event has occurred (e.g. a drug or a labelled substance has been applied). It is clear from the setting of the experiments that there is some variation in the real point in time when the blood probe has been taken: the true time when the measurement value has been obtained might be shortly before or after the intended time, and this true measurement time is not known to us. Since the inclusion of those time uncertainties in the model usually makes the analysis more difficult, it is standard to lump the time uncertainties with the measurement error. But especially at early times when concentrations change quickly, this may easily lead to wrong estimations, even if one assumes very high variances of the measurement error (we will demonstrate this later on a simple example). On the other hand, the inclusion of measurement time uncertainties (MTU) in algorithms aiming at inference making in complex models is not straightforward. In this article, we will present a modification of the Particle Filter (PF) algorithm (which we call MTU-PF) which is able to fully include a statistical model of the time uncertainties.

### Inference in complex systems

The assessment of the effectiveness of a drug in a clinical study has been done in the past by the direct computation of relatively simple statistical values. The enormous increase in complexity of the underlying models, due to present developments in medicine and biology, for instance in the areas of personalized medicine or systems biology, increases also the need for more sophisticated model-based inference methods.

The estimation of unobservable internal variables or model parameters from data which have been obtained from blood or tissue samples at several time points can reveal information on the concentrations and effectiveness of the substance under question. If these data come

from individuals which belong to two different (or even more) groups, e.g. test and control group, mixed effects are introduced in the underlying models. The inherent non-linearity and high variability of biological processes adds considerably to the difficulties one faces during the inference step. Inference in connection with dynamic models plays a major role in many other application areas. State and parameter estimation as well as model discrimination and validation are most common, but also optimal control problems should be mentioned.

It is often not enough to consider (independent) measurement noise [1]. Correlations between residuals are not uncommon, and the violation of this statistical assumption may lead to wrong estimates. A natural way to include correlated noise is to model two different types of noise: the dynamic (process or system) noise which is present in the dynamics of the system states and originates either from true random fluctuations in the system or from unmodelled dynamics in the system, and the measurement noise which is introduced by the measurement procedure or equipment and modelled by independent residuals. One possible approach is to use state space models which consist of a time-continuous model for the system states, e.g. based on Stochastic Differential Equations (SDEs), and a separate model for the time-discrete measurements.

### Parameter estimation with Maximum Likelihood approach

Parameter estimation in state space systems is a difficult problem. In a context where the system dynamics are modelled by Ordinary Differential Equations (ODEs) without correlated noise, the problem is most often considered as a (deterministic) optimization problem based on a Maximum Likelihood (ML) formulation. An overview of these approaches can be found in [2] and [3]; see also [4], which consider also other aspects like identifiability. A generalization of the ML approach including more flexible cost functions is given by the prediction error estimation method ([5]). The introduction of system noise in the state variables leads to optimization problems with SDE constraints. In this case, internal system states which cannot be directly observed need to be estimated jointly with the parameters, given the data. For this purpose, the parameter estimation methods must be augmented by appropriate state filtering methods. An overview of ML parameter estimation in these types of models is given in [6]. If the SDEs are non-linear, linearizations to the Kalman Filter, like the Extended Kalman Filter (EKF) or the Unscented Kalman Filter (UKF), are used to establish approximations to the means and covariances of the filter distributions over time. All those approximations suffer from the fact that they approximate the filtering distributions of the states by a Gaussian distribution at all time points and cannot adequately approximate skewed

or multimodal distributions. Better approximations are provided by simulation based methods like Sequential Monte Carlo (SMC) algorithms where good convergence results have been established ([7]). Nevertheless, they suffer from several drawbacks when applied to the joint estimation of dynamic states and fixed parameters ([8-10], see also [11]).

#### **Parameter estimation in a Bayesian context**

In a Bayesian context, in contrast to the “classical” ML approach, a prior distribution is assigned to the parameter vector, hence the parameters can be treated as random variables. In this sense, parameter estimation is done by evaluating the so-called posterior distribution which can be computed (at least theoretically) by Bayes’ theorem given the observations (measurements) and the prior distribution. In the context of high-dimensional spaces, this requires the computation of high-dimensional integrals which is not possible to do analytically. For this purpose, Markov Chain Monte Carlo (MCMC) methods provide powerful tools for the computation of simulation-based approximations to the posterior distribution. Again, in the context of the joint estimation of dynamic states and fixed parameters, the design of good proposal densities is a very difficult problem which renders the use of standard MCMC methods like the Metropolis-Hastings sampler impractical for the purposes of parameter estimation in state space systems.

It has long been a wish to combine both (dynamic) SMC and (static) MCMC methods to provide a general tool for the joint estimation of dynamic states and static parameters. Only recently, Andrieu et al. [11] proposed a very promising combination of both types of Monte Carlo approaches called Particle Markov Chain Monte Carlo (PMCMC) which is generally applicable and where also convergence has been proved.

In the present article, even though the PMCMC approach might be the preferred method for parameter estimation in state space systems, we will concentrate solely on the SMC methods, since our modification affects only this part. However, to be able to do parameter estimation in a pure SMC context, we rely on an approach that is very often used to avoid problems with the estimation of constant parameters. This approach consists in the introduction of artificial dynamics in the parameters, that means the parameters are allowed to slightly change their values over time. In this way, and in a Bayesian context, the parameters can be treated exactly in the same way as the system states. After building an augmented system state by concatenating the parameter vector and the state vector, the joint estimation of states and parameters reduces to filtering of the augmented state vector which makes SMC methods directly applicable to the problem.

#### **Particle filters for state and parameter estimation**

Particle filters ([12-14]) belong to the class of SMC methods for state filtering in state space models. Using the state augmentation approach, the method is also capable of estimating system parameters. The standard particle filter is designed for discrete, non-linear, and non-Gaussian models and can routinely be adapted to the continuous case with measurements at discrete times. The idea of the particle filter is that, at each time point, there is a sample based representation (the weighted particles) of the current estimate of the inner states and parameters which is based on the measurements that have been obtained up to the current time point. The particle cloud is then propagated through time, and the particles and weights are updated accordingly at each time point where measurements are available.

#### **Non-Linear Mixed Effects models**

Estimation in a Non-linear Mixed Effects model (NLME) involves the estimation of both global and individual parameters. With classical maximum likelihood estimation, the individual parameters are random variables equipped with a distribution while the global parameters remain constants with a “true” but unknown value. If the underlying model equations are non-linear, this leads to likelihood functions which are not analytically accessible and one has to rely on approximations. In the context where the system dynamics are modelled by ODEs, the most popular algorithm for NLME parameter estimation in the PK/PD context is the tool NONMEM ([15]). In [1] an estimation algorithm for NLME models based on Stochastic Differential Equations (SDEs) was proposed that uses the First-Order Conditional Estimation (FOCE) method to approximate the likelihood in combination with the EKF estimation in the SDEs. This has been added to NONMEM ([16]). In [17], a comparison between ODE and SDE based parameter estimation has been performed which showed that the interindividual variabilities were in general estimated to be smaller for the SDE model. Donnet and Samson ([18]) proposed a stochastic version of the Expectation-Maximization (SAEM) algorithm (for the estimation of the global parameters) in combination with MCMC methods (for the estimation of states and individual parameters). However, since MCMC exhibits slow mixing properties in the context of the estimation of states and parameters in state space models, in [19] MCMC has been replaced by the more promising PMCMC approach of Andrieu et al. ([11]).

On the other hand, in a Bayesian context, also the global parameters are equipped with a (prior) probability distribution, and the conceptual difference between global and individual parameters vanishes. The mixed effects model can then be considered as a hierarchical model with dependent parameters ([20,21], see also [22] for a more

recent population-based Bayesian approach to PK/PD modelling). Simulation-based (Monte Carlo) methods can easily be adapted to this case. Nevertheless, the above mentioned challenges to both SMC and MCMC methods are even higher due to the increased number of states and parameters in NLME models (the number of states and individual parameters has to be multiplied by the number of individuals).

### Aim of the article

Our goal is two-fold: Firstly, we want to show that the particle filter algorithm is applicable (with our modifications) also to more complex models when time uncertainties are formulated explicitly. Secondly, we want to show that the modification may even provide the possibility for further enhancement of the performance of the algorithm by presenting an adaptive time-stepping scheme which is only possible in the context of the new algorithm.

We do not claim that our MTU algorithm generally performs better or worse than the standard filter, nor that it should be the preferred method for estimation in non-linear mixed effects models. Rather, we provide a method which is usable for models where time uncertainties may play a major role. In these cases, it may indeed lead to better estimations. On the other hand, our method transfers the time-discrete particle filter approach, where updates based on the measurements very strictly depend on the measurement times, to a truly time-continuous approach, where updates to the filtering distributions can be performed at every point on the time-scale. Since we want to focus on the time uncertainties, we neglect discussing further issues like identifiability, model evaluation and model discrimination. In our application to the model of plasma-leucine kinetics, we try to avoid these issues by providing ad-hoc values to some of the parameters (especially to the variances of the system states).

### Motivating example

Let us have a look at an example for illustrating the benefits of a separate modelling of measurement time uncertainties. Let us consider a state space system given by the ODE

$$dq(t) = (-\alpha q(t) + \beta) dt$$

with parameters  $\alpha, \beta \in \mathbb{R}$ . Here  $q(t) \in \mathbb{R}$  denotes the state of the system at time  $t$ . We call the state trajectories obtained by this deterministic system the nominal evolutions of the states. We add noise to the system in a standard way by introducing an additional term  $\sigma dW_t$ , with a standard Wiener process  $(W_t)_{t \in \mathbb{R}^{\geq 0}}$  and a diffusion parameter  $\sigma \in \mathbb{R}$ . This leads to the following SDE describing the evolution of the state  $q$  over time:

$$dq(t) = (-\alpha q(t) + \beta) dt + \sigma dW_t.$$

Furthermore, let the initial state  $q(0)$  be given by a log-normal distribution with parameters  $\mu_{q_0}$  and  $\sigma_{q_0}^2$  (mean and variance of the logarithm of  $q(0)$ , respectively). The parameters chosen in our implementation of this example are shown in Table 1.

We assume that  $M$  measurements of the state  $q(t)$  will be taken at times  $t_j$  and that each measurement  $j$ ,  $j = 1, \dots, M$  is disturbed by normal noise with mean  $q(t_j)$  and with a fixed variance  $\sigma_y^2$ , i.e. measurement  $j$  is distributed according to  $y_j \sim \mathcal{N}(q(t_j), \sigma_y^2)$ . Usually, the times  $t_j$  are assumed to be known. In contrast, we will assume that in addition to the measurement value error, there is some uncertainty about the exact times where the measurements have been taken. If we attempt to take the  $j$ th measurement at the intended (or nominal) time  $\hat{t}_j$ , the measurement in fact takes place at time  $t_j$  which may be shortly before or after the intended time  $\hat{t}_j$ . A natural way to model these uncertainties is to assume that the measurement time  $t_j$  is given as a realization of a random variable  $T_j$ . In our example, we assume that  $T_j$  follows a truncated normal distribution given by the density

$$\gamma_j(t_j) := \frac{1}{\Gamma_j} \begin{cases} \exp\left(-\frac{(t_j - \hat{t}_j)^2}{0.3^2}\right) & \text{if } \max\{0, \hat{t}_j - 1\} \\ & \leq t_j \leq \hat{t}_j + 1, \\ 0 & \text{otherwise} \end{cases}$$

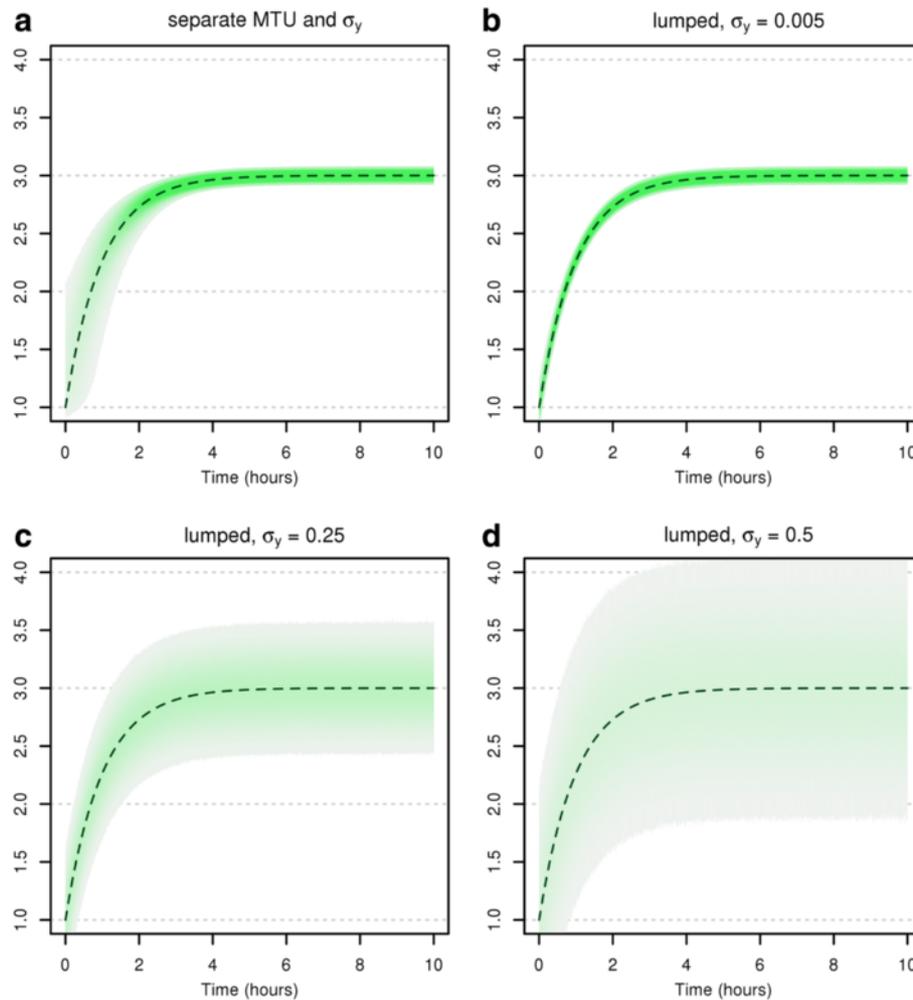
with normalizing constant

$$\Gamma_j := \int_{\max\{0, \hat{t}_j - 1\}}^{\hat{t}_j + 1} \exp\left(-\frac{(t_j - \hat{t}_j)^2}{0.3^2}\right)$$

and given intended measurement times  $\hat{t}_j$ . Figure 1 shows the different distributions for one measurement  $j$  for all possible intended times  $\hat{t}_j$ . In each of the subfigures (a)-(d), the shaded green area gives an impression of the "density" of the distribution of the measurement value  $y_j$  in dependence of the intended measurement time  $\hat{t}_j$  on the  $x$ -axis, while the dark-green dashed line depicts the nominal evolution of the state  $q$  over time. Subfigure (a) shows the distribution of the measurement values with time uncertainties, while (b)-(d) depict the distribution of the measurement values with known measurement times (in this case  $\hat{t}_j = t_j$ ), for several standard deviations  $\sigma_y$ : in

**Table 1 Parameters for the motivating example**

|                        |   |
|------------------------|---|
| true value of $\alpha$ | 1   |
| true value of $\beta$  | 3   |
| $\sigma$               | 0.05  |
| $\mu_{q_0}$            | $\log(1)$   |
| $\sigma_{q_0}$         | 0.1   |
| $\sigma_y$             | 0.005   |
| distribution of $T_j$  | $\mathcal{N}(\hat{t}_j, 0.3^2)$ truncated at $\hat{t}_j \pm 1$ and at $t_0$ |



**Figure 1 Assumed measurement distributions for the motivating example.** Measurement distribution resulting from (a) separate modelling of measurement time uncertainties and measurement value uncertainties, with  $\sigma_y = 0.005$ , and (b-d) lumped time-and-value uncertainties with several different assumed lumped measurement variances  $\sigma_y$ . The dashed dark-green line depicts the nominal evolution of the state  $q$  over time. The green shaded area depicts the region where the measurements are expected.

(b), the original standard deviation is used, while in (c) and (d), higher standard deviations are used which correspond to the cases with lumped value and time variations.

Comparing Figures 1(a) and 1(b-d), we observe that the distributions of the measurements exhibit clearly different shapes. For the “true” model depicted in Figure 1(a), if we consider a single point in time that lies in a time segment where the state values change quickly, the distribution of the measurement at this certain point in time is quite broad. The variance in the measured value is very high, whereas it is small in time segments where the state values change slowly. In contrast, for the standard particle filter, the measurement variance is constant and hence the assumed measurement distributions differ remarkably from the “true” distributions, howsoever the value of  $\sigma_y^2$  is chosen. It must be expected that this leads to difficulties

when inference on the states and parameters needs to be done based on these models. We will resume our example after having presented the MTU particle filter and will show that this is indeed the case.

## Methods

We divide this section into three subsections. In the first subsection, we fix the state and observation model we want to consider. In the second subsection entitled “Standard case” we outline the standard particle filter algorithm in the context of time-continuous states with time-discrete measurements, and the various probability distributions involved. Although nothing is new in this subsection, it serves several purposes. Firstly, the time-continuous case is relatively rarely considered in the literature; secondly, the derivation of our modification needs

a slightly more general formulation than it is standard for the discrete-time filter; and lastly, the comparison of our modified version with the standard case might more clearly reveal the differences between the two approaches. In the third subsection entitled “MTU particle filter”, we present our new modification of the particle filter. In the following section “Results and Discussion”, we compare the new MTU particle filter to the standard particle filter and to an alternative Maximum Likelihood estimation method on a simple artificial example. We also present an application of our MTU-PF method to a PK/PD study in a non-linear mixed-effects setting in direct comparison with the standard particle filter.

Note: a list of all used symbols with a short explanation can be found at the end of this paper.

### The model

#### State process

Let  $(\Omega, \mathcal{A}, \mathbb{P})$  be a probability space and for each  $t \in [t_0, \infty)$  with  $t_0 \in \mathbb{R}$  let  $(\mathcal{X}_t, \mathcal{B}_{\mathcal{X}_t})$  be an arbitrary measurable space. For each  $t \in [t_0, \infty)$  let further  $X_t : \Omega \rightarrow \mathcal{X}_t$  be an  $\mathcal{A}$ - $\mathcal{B}_{\mathcal{X}_t}$  measurable random variable such that  $X_{[t_0, \infty)} := (X_t)_{t \in [t_0, \infty)}$  is a continuous-time Markov process with general state space

$$\mathcal{X}_{[t_0, \infty)} := \prod_{t_0 \leq s} \mathcal{X}_s.$$

For each  $t \in [t_0, \infty)$ , denote by  $\mathcal{L}_{X_t}$  the pushforward measure of  $\mathbb{P}$  under  $X_t$ , i.e.  $\mathcal{L}_{X_t}(B) := \mathbb{P}(X_t^{-1}(B))$  for all  $B \in \mathcal{B}_{\mathcal{X}_t}$ . Further, denote by  $\mathcal{L}_{X_{[t_0, \infty)}}$  the pushforward measure of  $\mathbb{P}$  under  $X_{[t_0, \infty)} := (X_s)_{s \in [t_0, \infty)}$  (with the corresponding product algebra). Analogously, denote by

$$\mathcal{X}_{[t_0, t]} := \prod_{t_0 \leq s \leq t} \mathcal{X}_s \quad \text{for each } t \geq t_0$$

the state space restricted to the interval  $[t_0, t]$ , and denote by  $\mathcal{L}_{X_{[t_0, t]}}$  the corresponding pushforward measure. For each  $s$  and  $t$  with  $t > s \geq t_0$ , let  $K_{s,t}(x_s, dx_t)$  be the Markov kernel of the process  $X_{[t_0, \infty)}$  from time  $s$  to time  $t$ .

An important special case for  $X_{[t_0, \infty)}$  is given by a multidimensional Itô process on  $\mathcal{X}_t = \mathbb{R}^n$  (equipped with the corresponding Borel  $\sigma$ -algebra) defined through a stochastic differential equation (SDE)

$$dX_t = a(X_t, t) dt + B(X_t, t) dW_t$$

with drift  $a(x, t)$ , diffusion matrix  $B(x, t)$ , multidimensional standard Wiener process  $W_t$ , and initial variable  $X_{t_0}$ . In this case, it is possible to sample directly (at least approximately) from the kernels  $K_{s,t}$  when a suitable discretization method is applied, for instance the Euler-Maruyama method.

### Observations / measurements

Let the process  $X_{[t_0, \infty)}$  be observed via  $M$  random variables  $Y_{1:M}$  with values in measurable spaces  $(\mathcal{Y}_j, \mathcal{B}_{\mathcal{Y}_j})$ . Each single observation (measurement)  $Y_j$  depends on the state variable  $X_{t_j}$  at some time  $t_j$  and on the observation time (measurement time)  $t_j$  itself. We assume that, given the observation time  $t_j$  and the state  $X_{t_j} = x_{t_j}$ , the variable  $Y_j$  is independent of all other variables, and the conditional measure can be expressed via some conditional probability density  $g_j(y_j | x_{t_j}, t_j)$  with respect to a reference measure  $\mu_{\mathcal{Y}_j}$  on  $(\mathcal{Y}_j, \mathcal{B}_{\mathcal{Y}_j})$ . We do not require any further restrictions on  $g$  such as linear dependence on the states or Gaussianity.

### Observation / measurement times

The observation times (measurement times)  $t_j$  for  $j = 1, \dots, M$  are usually assumed to be deterministically given and known. Our variant of the particle filter will be based on the assumption that the observation times  $t_j$  are themselves realizations of random variables  $T_j$ . These variables model the uncertainty about exact observation times. In contrast to the observation variables  $Y_j$ , the observation times  $T_j$  are never observed (measured). We assume that all information available to us is their probability distribution on the half axis  $[t_0, \infty)$ , while in the case of the observations  $Y_j$ , we know both the densities  $g_j(y_j | x_{t_j}, t_j)$  and the observed values  $y_j$ .

In this article, we will only consider the simplest case where each variable  $T_j$  is independent of all others. Dependencies between the  $T_j$ 's, especially concerning the order of the observation times, may be considered natural but would lead to more complicated algorithms. However, order dependencies can easily be introduced via restrictions on the support of the variables. In general, the probability distribution of every single variable  $T_j$  shall be given by a density  $\gamma_j(t_j)$  with respect to the Lebesgue measure  $\lambda_{[t_0, \infty)}$  on the interval  $[t_0, \infty)$ .

In the following, we will consider the two cases mentioned, where either all  $t_j$  are deterministic and known or all  $t_j$  are random and unknown. Note that the first case formally coincides with the case that  $t_j$  is random but observed. We will therefore stick to the notation  $g_j(y_j | x_{t_j}, t_j)$  for the observation densities in both cases.

### Standard case: measurement times deterministic and known

We will first consider the standard case, where the observation times  $t_j$  are known. For simplicity, we assume here that the observation times  $t_{1:M}$  are strictly ordered increasingly, i.e.  $t_0 < t_1 < \dots < t_M$ .

The standard case of the particle filter is usually formulated for discrete-time Markov processes  $X_{t_{0:M}} := (X_{t_j})_{j \in \{0, \dots, M\}}$  with general state space where the state variables are only defined at the initial time  $t_0$  and at the times

$t_1, \dots, t_M$  when measurements occur. Nevertheless, this case is included in our more general framework where  $X_t$  is defined for all  $t \geq t_0$ . One just focuses on the state variables for those times only. In view of the later generalization to random observation times, we will consider the fixed values  $t_j$  as realizations of random variables  $T_j$  and condition all occurring densities on them. As mentioned above this assumption leads to the same results as if we assumed the values  $t_j$  to be given deterministically.

### Full model and filter model

The full model is given by the joint density of the variables  $X_{t_0:M}$  and  $Y_{1:M}$  (conditioned on the observation times  $T_{1:M} = t_{1:M}$ ) with respect to the product measure  $\mathcal{L}_{X_{t_0:M}} \prod_{j=1}^M \mu_{y_j}$ :

$$f^{X_{t_0:M}, Y_{1:M} | T_{1:M}}(x_{t_0:M}, y_{1:M} | t_{1:M}) := \prod_{j=1}^M g_j(y_j | x_{t_j}, t_j). \quad (1)$$

The filter at a given time  $t_k$  is based on a reduced model. This model is given by the joint density of the variables  $X_{t_0:k}$  and  $Y_{1:k}$  (conditioned on  $T_{1:M} = t_{1:M}$ ) with respect to the product measure  $\mathcal{L}_{X_{t_0:k}} \prod_{j=1}^k \mu_{y_j}$ :

$$f^{X_{t_0:k}, Y_{1:k} | T_{1:M}}(x_{t_0:k}, y_{1:k} | t_{1:M}) := \prod_{j=1}^k g_j(y_j | x_{t_j}, t_j). \quad (2)$$

This density is based on the state sequence  $X_{t_0:k}$ . In contrast, we can focus on the single state  $X_{t_k}$  by considering the joint density of the variables  $X_{t_k}$  and  $Y_{1:k}$  (given  $T_{1:M} = t_{1:M}$ ) with respect to  $\mathcal{L}_{X_{t_k}} \prod_{j=1}^k \mu_{y_j}$ . It can be computed by marginalization as follows:

$$\begin{aligned} & f^{X_{t_k}, Y_{1:k} | T_{1:M}}(x_{t_k}, y_{1:k} | t_{1:M}) \\ & := \int_{\{\tilde{x}_{t_0:k} \in \mathcal{X}_{t_0:k} : \tilde{x}_{t_k} = x_{t_k}\}} f^{X_{t_0:k}, Y_{1:k} | T_{1:M}}(\tilde{x}_{t_0:k}, y_{1:k} | t_{1:M}) \\ & \quad \times d\mathcal{L}_{X_{t_0:k}}(\tilde{x}_{t_0:k}) \end{aligned} \quad (3)$$

and the filter density at time  $t_k$  with respect to  $\mathcal{L}_{X_{t_k}}$  can then be computed with Bayes' theorem:

$$f^{X_{t_k} | Y_{1:k}, T_{1:M}}(x_{t_k} | y_{1:k}, t_{1:M}) := \frac{f^{X_{t_k}, Y_{1:k} | T_{1:M}}(x_{t_k}, y_{1:k} | t_{1:M})}{f^{Y_{1:k} | T_{1:M}}(y_{1:k} | t_{1:M})} \quad (4)$$

with

$$\begin{aligned} & f^{Y_{1:k} | T_{1:M}}(y_{1:k} | t_{1:M}) \\ & := \int_{\mathcal{X}_{t_0:k}} f^{X_{t_0:k}, Y_{1:k} | T_{1:M}}(x_{t_0:k}, y_{1:k} | t_{1:M}) d\mathcal{L}_{X_{t_0:k}}(x_{t_0:k}). \end{aligned} \quad (5)$$

For general (non-linear) models, the practical computation of the filter density is very difficult. Nevertheless,

the particle filter computes a Monte Carlo approximation using the fact that the filter densities  $f^{X_{t_k} | Y_{1:k}, T_{1:M}}$  can be computed recursively. This is done in two steps. We consider the filter distribution at time  $t_{k-1}$  given by the probabilities

$$\begin{aligned} & P(X_{t_{k-1}} \in B | Y_{1:k-1} = y_{1:k-1}, T_{1:M} = t_{1:M}) \\ & = \int_B f^{X_{t_{k-1}} | Y_{1:k-1}, T_{1:M}}(x_{t_{k-1}} | y_{1:k-1}, t_{1:M}) d\mathcal{L}_{X_{t_{k-1}}}(x_{t_{k-1}}) \end{aligned} \quad (6)$$

for each set  $B \in \mathcal{B}_{\mathcal{X}_{t_{k-1}}}$ . We then get first the prediction distribution, i.e. the distribution of  $X_{t_k}$  given the data  $Y_{1:k-1}$  (and  $T_{1:M}$ ), by use of the kernel  $K_{t_{k-1}, t_k}$ :

$$\begin{aligned} & P(X_{t_k} \in B | Y_{1:k-1} = y_{1:k-1}, T_{1:M} = t_{1:M}) \\ & = \int_B \int_{\mathcal{X}_{t_{k-1}}} f^{X_{t_{k-1}} | Y_{1:k-1}, T_{1:M}}(x_{t_{k-1}} | y_{1:k-1}, t_{1:M}) \\ & \quad \times d\mathcal{L}_{X_{t_{k-1}}}(x_{t_{k-1}}) K_{t_{k-1}, t_k}(x_{t_{k-1}}, dx_{t_k}) \end{aligned} \quad (7)$$

for each set  $B \in \mathcal{B}_{\mathcal{X}_{t_k}}$ . Then we use Bayes' theorem to get the filter distribution at time  $t_k$ :

$$\begin{aligned} & P(X_{t_k} \in B | Y_{1:k} = y_{1:k}, T_{1:M} = t_{1:M}) \\ & = \int_B \frac{g_k(y_k | x_{t_k}, t_k)}{f^{Y_k | Y_{1:k-1}, T_{1:M}}(y_k | y_{1:k-1}, t_{1:M})} \\ & \quad \int_{\mathcal{X}_{t_{k-1}}} f^{X_{t_{k-1}} | Y_{1:k-1}, T_{1:M}}(x_{t_{k-1}} | y_{1:k-1}, t_{1:M}) \\ & \quad \times d\mathcal{L}_{X_{t_{k-1}}}(x_{t_{k-1}}) K_{t_{k-1}, t_k}(x_{t_{k-1}}, dx_{t_k}) \end{aligned} \quad (8)$$

for each set  $B \in \mathcal{B}_{\mathcal{X}_{t_k}}$ , with normalizing constant

$$\begin{aligned} & f^{Y_k | Y_{1:k-1}, T_{1:M}}(y_k | y_{1:k-1}, t_{1:M}) \\ & := \int_{\mathcal{X}_{t_k}} g_k(y_k | x_{t_k}, t_k) \\ & \quad \int_{\mathcal{X}_{t_{k-1}}} f^{X_{t_{k-1}} | Y_{1:k-1}, T_{1:M}}(x_{t_{k-1}} | y_{1:k-1}, t_{1:M}) \\ & \quad \times d\mathcal{L}_{X_{t_{k-1}}}(x_{t_{k-1}}) K_{t_{k-1}, t_k}(x_{t_{k-1}}, dx_{t_k}). \end{aligned} \quad (9)$$

### Importance sampling

Another ingredient for the particle filter is sequential importance sampling. We assume that a second Markov chain  $\tilde{X}_{t_0:M}$  on the same state space is given with push-forward measures  $\mathcal{L}_{\tilde{X}_{t_j}}$  and kernels  $\tilde{K}_{t_{j-1}, t_j}(x_{t_{j-1}}, dx_{t_j})$  for  $j = 1, \dots, M$ . We assume that for each  $x_{t_{j-1}} \in \mathcal{X}_{t_{j-1}}$ , the measure  $\tilde{K}_{t_{j-1}, t_j}(x_{t_{j-1}}, \cdot)$  is absolutely continuous with respect to the measure  $\tilde{K}_{t_{j-1}, t_j}(x_{t_{j-1}}, \cdot)$ . It follows that

the Radon-Nikodym derivative (written as conditional density)

$$\varrho_{t_j | t_{j-1}}(x_{t_j} | x_{t_{j-1}}) := \frac{K_{t_{j-1}, t_j}(x_{t_{j-1}}, dx_{t_j})}{\tilde{K}_{t_{j-1}, t_j}(x_{t_{j-1}}, dx_{t_j})}$$

exists. We further assume that the pushforward measure  $\mathcal{L}_{X_{t_0}}$  under  $X_{t_0}$  is absolutely continuous with respect to the corresponding pushforward measure  $\mathcal{L}_{\tilde{X}_{t_0}}$  under  $\tilde{X}_{t_0}$  with Radon-Nikodym derivative

$$\varrho_{t_0}(x_{t_0}) := \frac{d\mathcal{L}_{X_{t_0}}(x_{t_0})}{d\mathcal{L}_{\tilde{X}_{t_0}}(x_{t_0})}.$$

For sequential importance sampling, we need to be able to sample from the initial measure  $\mathcal{L}_{\tilde{X}_{t_0}}$  and from the kernels

$$\tilde{K}_{t_{j-1}, t_j}(x_{t_{j-1}}, \cdot)$$

for each  $x_{t_{j-1}} \in \mathcal{X}_{t_{j-1}}$ , and to compute  $\varrho_{t_0}(x_{t_0})$  as well as  $\varrho_{t_j | t_{j-1}}(x_{t_j} | x_{t_{j-1}})$  pointwise.

Using

$$K_{t_{k-1}, t_k}(x_{t_{k-1}}, dx_{t_k}) = \varrho_{t_k | t_{k-1}}(x_{t_k} | x_{t_{k-1}}) \tilde{K}_{t_{k-1}, t_k}(x_{t_{k-1}}, dx_{t_k}),$$

we can then write the recursive formula (8) for the filter distribution at time  $t_k$  as

$$\begin{aligned} & P(X_{t_k} \in B | Y_{1:k} = y_{1:k}, T_{1:M} = t_{1:M}) \\ &= \int_B \frac{g_k(y_k | x_{t_k}, t_k)}{f^{Y_k | Y_{1:k-1}, T_{1:M}}(y_k | y_{1:k-1}, t_{1:M})} \\ & \int_{\mathcal{X}_{t_{k-1}}} f^{X_{t_{k-1}} | Y_{1:k-1}, T_{1:M}}(x_{t_{k-1}} | y_{1:k-1}, t_{1:M}) \varrho_{t_k | t_{k-1}}(x_{t_k} | x_{t_{k-1}}) \\ & \quad \times d\mathcal{L}_{X_{t_{k-1}}}(x_{t_{k-1}}) \tilde{K}_{t_{k-1}, t_k}(x_{t_{k-1}}, dx_{t_k}) \end{aligned} \quad (10)$$

for each  $B \in \mathcal{B}_{\mathcal{X}_{t_k}}$ . The direct computation of the normalizing constants  $f^{Y_k | Y_{1:k-1}, T_{1:M}}(y_k | y_{1:k-1}, t_{1:M})$  (while  $y_{1:M}$  is assumed to be fixed) is not necessary. Sequential importance sampling is performed as follows. Draw a number  $N$  of realizations  $x_{t_0}^i$  from  $\mathcal{L}_{\tilde{X}_{t_0}}$  and compute the corresponding unnormalized weights

$$w_{t_0}^i := \varrho_{t_0}(x_{t_0}^i) \quad \text{for all } i = 1, \dots, N.$$

Then, for all  $k = 1, \dots, M$ , sample realizations  $x_{t_k}^i$  from the kernel  $\tilde{K}_{t_{k-1}, t_k}(x_{t_{k-1}}^i, dx_{t_k})$  for each  $i = 1, \dots, N$  and compute the unnormalized weights

$$w_{t_k}^i := \varrho_{t_k | t_{k-1}}(x_{t_k}^i | x_{t_{k-1}}^i) g_k(y_k | x_{t_k}^i, t_k) w_{t_{k-1}}^i \quad \text{for all } i = 1, \dots, N.$$

For suitable integrable functions  $h$  (e.g. fulfilling some mild restrictions on how fast  $h$  may increase with  $x$ , see [23] for details), the expectation of  $h$  with respect to the

filter density conditioned on the observations  $Y_{1:k} = y_{1:k}$ , given by

$$\begin{aligned} & E[h(X_{t_k}) | Y_{1:k} = y_{1:k}, T_{1:M} = t_{1:M}] \\ &:= E_{f^{X_{t_k} | Y_{1:k}, T_{1:M}}(\cdot | y_{1:k}, t_{1:M})}[h(X_{t_k})] \\ &= \int f^{X_{t_k} | Y_{1:k}, T_{1:M}}(x_{t_k} | y_{1:k}, t_{1:M}) h(x_{t_k}) d\mathcal{L}_{X_{t_k}}(x_{t_k}), \end{aligned} \quad (11)$$

can then be approximated by

$$E_{t_k, N}[h(X_{t_k}) | Y_{1:k} = y_{1:k}, T_{1:M} = t_{1:M}] := \frac{\sum_{i=1}^N w_{t_k}^i h(x_{t_k}^i)}{\sum_{i=1}^N w_{t_k}^i} \quad (12)$$

where  $N$  is the number of particles. In fact, it can be shown that as  $N$  approaches infinity, these empirical expectations converge to the filter expectations:

$$\begin{aligned} & \lim_{N \rightarrow \infty} E_{t_k, N}[h(X_{t_k}) | Y_{1:k} = y_{1:k}, T_{1:M} = t_{1:M}] \\ &= E[h(X_{t_k}) | Y_{1:k} = y_{1:k}, T_{1:M} = t_{1:M}]. \end{aligned} \quad (13)$$

Note that if we can sample from the Markov kernels of  $X_{t_j}$ , we can choose  $\tilde{X}_{t_j} = X_{t_j}$  (at least in law), whence  $\varrho_{t_0}(x_{t_0}) \equiv 1$  and  $\varrho_{t_j | t_{j-1}}(x_{t_j} | x_{t_{j-1}}) \equiv 1$ . This is a standard choice, but in terms of efficiency of the particle filter algorithm not always the best one. On the other hand, finding a suitable Markov chain  $\tilde{X}_{t_0:M}$  different from  $X_{t_0:M}$  is not an easy task.

### Resampling

If the number  $N$  of samples through time is fixed, the samples obtained by sequential importance sampling quickly degenerate since most of the normalized weights decrease rapidly towards 0. The degree of degeneracy is often measured by an estimate of the so-called effective sample size (ESS). This estimate at time  $t$  is given by

$$n_{\text{ESS}} := \frac{1}{\sum_{i=1}^N (\tilde{w}_t^i)^2} \quad (14)$$

where

$$\tilde{w}_t^i := \frac{w_t^i}{\sum_{v=1}^N w_t^v} \quad (15)$$

are the normalized weights. It obtains its maximal value  $N$  if all weights are equal, and it approaches 1 if the variance of the weights and thus the degree of degeneracy increases. To avoid this degeneration of the samples, a resampling step needs to be done when the ESS drops below a threshold  $N_{\text{Threshold}}$  (which is usually chosen to be  $N/2$ ).

Resampling at some time  $s_\ell$  is based on given non-negative (unnormalized) selection weights  $v_{s_\ell}^i$  for each

particle index  $i$ : One repeatedly selects particles with probabilities  $p_\ell^i$  given by the normalized selection weights

$$p_\ell^i := \frac{v_{s_\ell}^i}{\sum_{v=1}^N v_{s_\ell}^v}. \quad (16)$$

This is multinomial resampling. There exist procedures where each single particle is still selected with probability  $p_\ell^i$ , but with reduced overall variance, for instance stratified resampling or systematic resampling which should be preferred [24,25]. In any case, resampling defines a selection function  $\iota_\ell : I \rightarrow I$  on the index set  $I := \{1, \dots, N\}$ . Resampling is then done by replacing the state samples  $(x_{s_\ell}^i)_{i=1, \dots, N}$  by the selected state samples  $(x_{s_\ell}^{\iota_\ell(i)})_{i=1, \dots, N}$ . Since before selection the probability that the particle  $i$  will be chosen is  $p_\ell^i$  for each draw, the expected number of times that particle  $i$  has been chosen after  $N$  draws is  $Np_\ell^{\iota_\ell(i)}$ . To correct for the introduced bias, the normalized weight  $\tilde{w}_{s_\ell}^i$  for each selected particle  $i$  needs then to be corrected by replacing it by the weight

$$\frac{\tilde{w}_{s_\ell}^{\iota_\ell(i)}}{Np_\ell^{\iota_\ell(i)}} \bigg/ \sum_{v=1}^N \frac{\tilde{w}_{s_\ell}^{\iota_\ell(v)}}{Np_\ell^{\iota_\ell(v)}} = \frac{w_{s_\ell}^{\iota_\ell(i)}}{v_{s_\ell}^{\iota_\ell(i)}} \bigg/ \sum_{v=1}^N \frac{w_{s_\ell}^{\iota_\ell(v)}}{v_{s_\ell}^{\iota_\ell(v)}} \quad (17)$$

(using (16)). The necessary correction is therefore achieved if the unnormalized weights  $(w_{s_\ell}^i)_{i=1, \dots, N}$  are replaced by the corrected unnormalized weights  $(w_{s_\ell}^{\iota_\ell(i)} / v_{s_\ell}^{\iota_\ell(i)})_{i=1, \dots, N}$ .

Note that in the original particle filter, the selection weights  $v_{s_\ell}^i$  at time  $s_\ell$  are chosen to be the particle weights (before the replacement), i.e.

$$v_{s_\ell}^i = w_{s_\ell}^i \quad \text{for } i = 1, \dots, N,$$

such that after the resampling step the unnormalized weights are all equal to 1. Nevertheless, in general their choice is free and may be based on the observations (which is used in the so-called auxiliary particle filter [26]).

### Particle filter algorithm

The particle filter computes the state realizations and weights recursively through time. In its standard form, the particle filter can be stated as in algorithm 1. Note that if one chooses  $\tilde{X}_{[t_0, \infty)} = X_{[t_0, \infty)}$  (in law), then  $Q_{t_k | t_{k-1}}(x_{t_k}^i | x_{t_{k-1}}^i) \equiv 1$  and the update of the weights simplifies to

$$w_{t_k}^i = g_k(y_k | x_{t_k}^i, t_k) w_{t_{k-1}}^i.$$

---

### Algorithm 1 Standard particle filter

---

- 1: {At time  $t_0$ ;}
    - 2: Sample  $N$  state realizations  $(x_{t_0}^i)_{i=1, \dots, N}$  of  $\tilde{X}_{t_0}$ , with  $N$  large.
    - 3: **for**  $i = 1, \dots, N$  **do**
    - 4:     Set the weight  $w_{t_0}^i = Q_{t_0}(x_{t_0}^i)$ .
    - 5:     **end for**
    - 6: **for all** times  $t_k, k = 1, \dots, M$  **do**
    - 7:     {Resample the particle set  $(x_{t_{k-1}}^i)_{i=1, \dots, N}$  if necessary (e.g. if the ESS drops below a threshold):}
    - 8:     Generate a selection function  $\iota$  according to some selection weights  $(v_{t_{k-1}}^i)_{i=1, \dots, N}$ .
    - 9:     **for**  $i = 1, \dots, N$  **do**
    - 10:         Replace the state sample  $x_{t_{k-1}}^i$  by the selected state sample  $x_{t_{k-1}}^{\iota(i)}$ .
    - 11:         Replace the unnormalized weight  $w_{t_{k-1}}^i$  by the corrected weight  $w_{t_{k-1}}^{\iota(i)} / v_{t_{k-1}}^{\iota(i)}$ .
    - 12:         **end for**
    - 13:     **for**  $i = 1, \dots, N$  **do**
    - 14:         Sample a realization  $x_{t_k}^i$  from the Markov kernel  $\tilde{K}_{t_{k-1}, t_k}(x_{t_{k-1}}^i, \cdot)$ .
    - 15:         Update the weight  $w_{t_k}^i = Q_{t_k | t_{k-1}}(x_{t_k}^i | x_{t_{k-1}}^i) g_k(y_k | x_{t_k}^i, t_k) w_{t_{k-1}}^i$ .
    - 16:     **end for**
    - 17:     For given suitable integrable functions  $h$ , compute estimates  $E_{t_k, N}[h(X_{t_k}) | Y_{1:k}, T_{1:M} = t_{1:M}] := \frac{\sum_{i=1}^N w_{t_k}^i h(x_{t_k}^i)}{\sum_{i=1}^N w_{t_k}^i}$ .
    - 18: **end for**
-

### Data Likelihood

Model validation or discrimination is generally based on the data likelihood

$$\begin{aligned} Z_{t_k}(t_{1:M}) &:= f^{Y_{1:k} | T_{1:M}}(y_{1:k} | t_{1:M}) \\ &= \int_{\mathcal{X}_{t_0:k}} f^{X_{t_0:k}, Y_{1:k} | T_{1:M}}(x_{t_0:k}, y_{1:k} | t_{1:M}) d\mathcal{L}_{X_{t_0:k}}(x_{t_0:k}) \\ &= \mathbb{E} \left[ f^{X_{t_0:k}, Y_{1:k} | T_{1:M}}(\cdot, y_{1:k} | t_{1:M}) \right] \end{aligned} \quad (18)$$

for given observations  $y_{1:k}$ . Without resampling, the data likelihood could be approximated by the empirical mean of the unnormalized weights, i.e. by

$$\hat{Z}_{t_k}(t_{1:M}) := \frac{1}{N} \sum_{i=1}^N w_{t_k}^i \quad (19)$$

because this is the empirical estimate for the above expectation. After a resampling step, this is not valid any longer. Nevertheless, in any case, the data likelihood can be computed recursively by the following estimate of the ratio  $Z_{t_k}(t_{1:M})/Z_{t_{k-1}}(t_{1:M})$ :

$$\frac{\widehat{Z}_{t_k}(t_{1:M})}{\widehat{Z}_{t_{k-1}}(t_{1:M})} := \frac{\sum_{i=1}^N \varrho_{t_k | t_{k-1}}(x_{t_k}^i | x_{t_{k-1}}^i) g_k(y_k | x_{t_k}^i, t_k) w_{t_{k-1}}^i}{\sum_{i=1}^N w_{t_{k-1}}^i}, \quad (20)$$

with initial estimate  $\hat{Z}_{t_0}(t_{1:M}) = 1$  (see e.g. [27]).

### MTU particle filter: Uncertain measurement times

We now assume that each observation time  $t_j$  is a realization of a random variable  $T_j$ . Its distribution is expressed via densities  $\gamma_j$  with respect to the Lebesgue measure  $\lambda_{[t_0, \infty)}$ . The observation times  $t_j$  themselves are not observed.

### Full model

The full model in this case will include complete continuous state paths, since the observation times are now distributed over the complete time axis  $[t_0, \infty)$ , and the observations may potentially depend on every state  $x_{t_j}$  for  $t_j \in [t_0, \infty)$ . Consider therefore the joint density of the variables  $X_{[t_0, \infty)}$ ,  $Y_{1:M}$  and  $T_{1:M}$ , with respect to the product measure  $\mathcal{L}_{X_{[t_0, \infty)}} \prod_{j=1}^M \mu \gamma_j \prod_{j=1}^M \lambda_{[t_0, \infty)}$ :

$$f^{X_{[t_0, \infty)}, Y_{1:M}, T_{1:M}}(x_{[t_0, \infty)}, y_{1:M}, t_{1:M}) = \prod_{j=1}^M g_j(y_j | x_{t_j}, t_j) \prod_{j=1}^M \gamma_j(t_j). \quad (21)$$

### Filter model

The filter at a given time  $t \geq t_0$  is again based on a reduced model. This model is given by the joint density of the following variables:  $X_{[t_0, t]}$ , denoting the state paths until time  $t$ ; further only those variables  $Y_j$  for which  $T_j \leq t$ ;

and finally  $T_{1:M}$ . This density is given with respect to the product measure  $\mathcal{L}_{X_{[t_0, t]}} \prod_{j=1}^M \mu \gamma_j \prod_{j=1}^M \lambda_{[t_0, \infty)}$  by:

$$\hat{f}_t^{X_{[t_0, t]}, Y_{1:M}, T_{1:M}}(x_{[t_0, t]}, y_{1:M}, t_{1:M}) = \prod_{\substack{j=1 \\ t_j \leq t}}^M g_j(y_j | x_{t_j}, t_j) \prod_{j=1}^M \gamma_j(t_j). \quad (22)$$

Note that we cannot use the simple notation of the standard case where for filtering only the first  $k$  observations are taken into consideration at time  $t_k$ , since neither the observations are ordered in time nor the times  $t_j$  are fixed in advance. For this reason we have to include all measurements  $Y_{1:M}$  also into the filter model. Note that even though we use the complete data  $Y_{1:M} = y_{1:M}$  in the notation, only those  $y_j$  have to be known at time  $t$  for which  $t_j \leq t$  holds. To avoid confusion, we mark all densities connected to the filter model at time  $t$  by a hat superscript (and by the index  $t$ ).

We will now derive formulas for the filter density. Since we assume that the observation times  $T_{1:M}$  are not observed, we use marginalization to get the joint density for  $X_{[t_0, t]}$  and  $Y_{1:M}$  only, which is

$$\begin{aligned} \hat{f}_t^{X_{[t_0, t]}, Y_{1:M}}(x_{[t_0, t]}, y_{1:M}) &= \int_{t_0}^{\infty} \cdots \int_{t_0}^{\infty} \hat{f}_t^{X_{[t_0, t]}, Y_{1:M}, T_{1:M}}(x_{[t_0, t]}, y_{1:M}, t_{1:M}) dt_1 \cdots dt_M \\ &= \int_{t_0}^{\infty} \cdots \int_{t_0}^{\infty} \prod_{\substack{j=1 \\ t_j \leq t}}^M g_j(y_j | x_{t_j}, t_j) \prod_{j=1}^M \gamma_j(t_j) dt_1 \cdots dt_M \end{aligned} \quad (23)$$

with respect to the product measure  $\mathcal{L}_{X_{[t_0, t]}} \prod_{j=1}^M \mu \gamma_j$ . We will further simplify this density. If we define

$$\bar{g}_{j,t}(y_j | x_{t_j}, t_j) := \begin{cases} g_j(y_j | x_{t_j}, t_j) & \text{if } t_j \leq t \\ 1 & \text{otherwise,} \end{cases}$$

then

$$\prod_{\substack{j=1 \\ t_j \leq t}}^M g_j(y_j | x_{t_j}, t_j) = \prod_{j=1}^M \bar{g}_{j,t}(y_j | x_{t_j}, t_j)$$

and further

$$\begin{aligned} \hat{f}_t^{X_{[t_0, t]}, Y_{1:M}}(x_{[t_0, t]}, y_{1:M}) &= \int_{t_0}^{\infty} \cdots \int_{t_0}^{\infty} \prod_{\substack{j=1 \\ t_j \leq t}}^M g_j(y_j | x_{t_j}, t_j) \prod_{j=1}^M \gamma_j(t_j) dt_1 \cdots dt_M \\ &= \int_{t_0}^{\infty} \cdots \int_{t_0}^{\infty} \prod_{j=1}^M (\bar{g}_{j,t}(y_j | x_{t_j}, t_j) \gamma_j(t_j)) dt_1 \cdots dt_M \\ &= \prod_{j=1}^M \int_{t_0}^{\infty} \bar{g}_{j,t}(y_j | x_{t_j}, t_j) \gamma_j(t_j) dt_j, \end{aligned} \quad (24)$$

where the last step is possible because the factor indexed by  $j$  does not depend on  $t_{j'}$  for  $j' \neq j$ . For each  $j$ , we can split the integration by  $t_j$  at the time point  $t$  into two parts and get

$$\begin{aligned} & \int_{t_0}^{\infty} \bar{g}_{j,t}(y_j | x_{t_j}, t_j) \gamma_j(t_j) \\ &= \int_{t_0}^t g_j(y_j | x_{t_j}, t_j) \gamma_j(t_j) dt_j + \int_t^{\infty} \gamma_j(t_j) dt_j \\ &= 1 + \int_{t_0}^t (g_j(y_j | x_{t_j}, t_j) - 1) \gamma_j(t_j) dt_j \end{aligned}$$

where the last step follows from the fact that  $\gamma_j$  is a probability density and therefore

$$\int_{t_0}^t \gamma_j(t_j) dt_j + \int_t^{\infty} \gamma_j(t_j) dt_j = \int_{t_0}^{\infty} \gamma_j(t_j) dt_j = 1$$

holds. Inserting this into (24), we get

$$\begin{aligned} & \hat{f}_t^{X_{[t_0,t]}, Y_{1:M}}(x_{[t_0,t]}, y_{1:M}) \\ &= \prod_{j=1}^M \left( 1 + \int_{t_0}^t (g_j(y_j | x_{t_j}, t_j) - 1) \gamma_j(t_j) dt_j \right). \end{aligned}$$

With a further marginalization, we get the joint density of  $X_t$  and  $Y_{1:M}$  for the filter model,

$$\begin{aligned} & \hat{f}_t^{X_t, Y_{1:M}}(x_t, y_{1:M}) \\ &:= \int_{\{\tilde{x}_{[t_0,t]} \in \mathcal{X}_{[t_0,t]} : \tilde{x}_t = x_t\}} \hat{f}_t^{X_{[t_0,t]}, Y_{1:M}}(\tilde{x}_{[t_0,t]}, y_{1:M}) d\mathcal{L}_{X_{[t_0,t]}}(\tilde{x}_{[t_0,t]}) \\ &= \int_{\{\tilde{x}_{[t_0,t]} \in \mathcal{X}_{[t_0,t]} : \tilde{x}_t = x_t\}} \prod_{j=1}^M \left( 1 + \int_{t_0}^t (g_j(y_j | \tilde{x}_{t_j}, t_j) - 1) \gamma_j(t_j) dt_j \right) d\mathcal{L}_{X_{[t_0,t]}}(\tilde{x}_{[t_0,t]}) \end{aligned} \quad (25)$$

which is with respect to the product measure  $\mathcal{L}_{X_t} \prod_{j=1}^M \mu_{\mathcal{Y}_j}$ . From this density, we finally can compute the filter density with respect to  $\mathcal{L}_{X_t}$  by applying Bayes' theorem:

$$\hat{f}_t^{X_t | Y_{1:M}}(x_t | y_{1:M}) = \frac{\hat{f}_t^{X_t, Y_{1:M}}(x_t, y_{1:M})}{\hat{f}_t^{Y_{1:M}}(y_{1:M})} \quad (26)$$

where

$$\hat{f}_t^{Y_{1:M}}(y_{1:M}) = \int_{\mathcal{X}_t} \hat{f}_t^{X_t, Y_{1:M}}(\tilde{x}_t, y_{1:M}) d\mathcal{L}_{X_t}(\tilde{x}_t) \quad (27)$$

is the data likelihood with respect to the measure  $\prod_{j=1}^M \mu_{\mathcal{Y}_j}$ .

### Effective computation of the filter distributions

In the following paragraph, we will show how the densities of the filter distributions given by (26) can be effectively computed. This is the basis for the formulation of our MTU particle filter method.

Let the observations  $y_{1:M} \in \mathcal{Y}_{1:M}$  with  $f^{Y_{1:M}}(y_{1:M}) > 0$  be given. For each time  $t \in [t_0, \infty)$  and for each  $j \in \{1, \dots, M\}$ , we define random variables

$$W_{j,t} : \Omega \rightarrow \mathbb{R}_{\geq 0} \quad \text{and} \quad W_t : \Omega \rightarrow \mathbb{R}_{\geq 0}$$

by the following system of ODEs

$$\begin{aligned} dW_{1,t}(\omega) &= (g_1(y_1 | X_t(\omega), t) - 1) \gamma_1(t) dt \\ &\vdots \\ dW_{M,t}(\omega) &= (g_M(y_M | X_t(\omega), t) - 1) \gamma_M(t) dt \end{aligned} \quad (28)$$

for each  $\omega \in \Omega$  with initial values

$$W_{1,t_0}(\omega) = \dots = W_{M,t_0}(\omega) = 1, \quad (29)$$

and by

$$W_t = \prod_{j=1}^M W_{j,t}. \quad (30)$$

We will show that for each set  $A$  in the  $\sigma$ -algebra generated by the variable  $X_t$ , it holds that

$$\int_{X_t(A)} \hat{f}_t^{X_t | Y_{1:M}}(x_t | y_{1:M}) d\mathcal{L}_{X_t}(x_t) = \frac{\int_A W_t(\omega) dP(\omega)}{\int_{\Omega} W_t(\omega) dP(\omega)} \quad (31)$$

where  $\hat{f}_t^{X_t | Y_{1:M}}$  is the filter density. That means we can use the processes  $W_{j,t}$  and  $W_t$  to compute the filter distributions through time. From this, it follows immediately that we can also compute filter expectations. Indeed, for any real-valued measurable function  $h$  on  $\mathcal{X}$  such that  $E[|h(X_t)|] < \infty$ , it holds that the expectation of  $h(X_t)$  given  $Y_{1:M} = y_{1:M}$  with respect to the filtered state  $X_t$  defined by

$$\begin{aligned} \hat{E}_t[h(X_t) | Y_{1:M} = y_{1:M}] &:= E_{\hat{f}_t^{X_t | Y_{1:M}}(\cdot | y_{1:M})}[h(X_t)] \\ &= \int_{\mathcal{X}_t} \hat{f}_t^{X_t | Y_{1:M}}(x_t | y_{1:M}) h(x_t) d\mathcal{L}_{X_t}(x_t), \end{aligned} \quad (32)$$

is given by the following equation:

$$\hat{E}_t[h(X_t) | Y_{1:M} = y_{1:M}] = \frac{\int_{\Omega} W_t(\omega) h(X_t(\omega)) dP(\omega)}{\int_{\Omega} W_t(\omega) dP(\omega)}. \quad (33)$$

To show our assertion, we consider the processes  $W_{j,t}$  for  $j = 1, \dots, M$ . According to (28) and (29), each  $W_{j,t}$  is defined as

$$W_{j,t} = 1 + \int_{t_0}^t (g_j(y_j | X_{t_j}, t_j) - 1) \gamma_j(t_j) dt_j, \quad (34)$$

so, with (30),

$$W_t = \prod_{j=1}^M W_{j,t} = \prod_{j=1}^M \left( 1 + \int_{t_0}^t (g_j(y_j | X_{t_j}, t_j) - 1) \gamma_j(t_j) dt_j \right) \quad (35)$$

holds. Thus, to prove (31), we have to show that for each set  $A$  from the  $\sigma$ -algebra generated by  $X_t$ ,

$$\frac{\int_A W_t(\omega) dP(\omega)}{\int_\Omega W_t(\omega) dP(\omega)} = \left( \int_{X_t(A)} \int_{\{\tilde{x}_{[t_0,t]} \in \mathcal{X}_{[t_0,t]} : \tilde{x}_t = x_t\}} \prod_{j=1}^M \left( 1 + \int_{t_0}^t (g_j(y_j | \tilde{x}_{t_j}, t_j) - 1) \gamma_j(t_j) dt_j \right) \times d\mathcal{L}_{X_{[t_0,t]}}(\tilde{x}_{[t_0,t]}) d\mathcal{L}_{X_t}(x_t) \right) / \hat{f}_t^{Y_{1:M}}(y_{1:M})$$

holds (see (25) and (26)). It is enough to show the equality for the numerator, i.e.

$$\int_A W_t(\omega) dP(\omega) = \int_{X_t(A)} \int_{\{\tilde{x}_{[t_0,t]} \in \mathcal{X}_{[t_0,t]} : \tilde{x}_t = x_t\}} \prod_{j=1}^M \left( 1 + \int_{t_0}^t (g_j(y_j | \tilde{x}_{t_j}, t_j) - 1) \gamma_j(t_j) dt_j \right) \times d\mathcal{L}_{X_{[t_0,t]}}(\tilde{x}_{[t_0,t]}) d\mathcal{L}_{X_t}(x_t)$$

since the equality of the denominator follows then immediately from the special case  $A = \Omega$  and from the fact that

$$\hat{f}_t^{Y_{1:M}}(y_{1:M}) = \int_{\mathcal{X}_t} \int_{\{\tilde{x}_{[t_0,t]} \in \mathcal{X}_{[t_0,t]} : \tilde{x}_t = x_t\}} \prod_{j=1}^M \left( 1 + \int_{t_0}^t (g_j(y_j | \tilde{x}_{t_j}, t_j) - 1) \gamma_j(t_j) dt_j \right) \times d\mathcal{L}_{X_{[t_0,t]}}(\tilde{x}_{[t_0,t]}) d\mathcal{L}_{X_t}(x_t).$$

Using the variable transformation  $(X_{[t_0,t]}(\omega), X_t(\omega)) = (\tilde{x}_{[t_0,t]}, x_t)$ , we get

$$\begin{aligned} & \int_A W_t(\omega) dP(\omega) \\ &= \int_A \prod_{j=1}^M \left( 1 + \int_{t_0}^t (g_j(y_j | X_{t_j}(\omega), t_j) - 1) \gamma_j(t_j) dt_j \right) dP(\omega) \\ &= \int_{X_t(A)} \int_{\{\tilde{x}_{[t_0,t]} \in \mathcal{X}_{[t_0,t]} : \tilde{x}_t = x_t\}} \prod_{j=1}^M \left( 1 + \int_{t_0}^t (g_j(y_j | \tilde{x}_{t_j}, t_j) - 1) \gamma_j(t_j) dt_j \right) \\ & \quad \times d\mathcal{L}_{X_{[t_0,t]}}(\tilde{x}_{[t_0,t]}) d\mathcal{L}_{X_t}(x_t). \end{aligned}$$

This is what we wanted to show.

### Weights

Since for each  $t$  and for each  $j = 1, \dots, M$  the random variables  $W_{j,t}$  and  $W_t$  depend only on the process  $X_{[t_0,t]}$  until time  $t$ , we can define functions  $w_{j,t} : X_{[t_0,t]}(\Omega) \rightarrow$

$\mathbb{R}_{\geq 0}$  and  $w_t : X_{[t_0,t]}(\Omega) \rightarrow \mathbb{R}_{\geq 0}$  by setting for each  $x_{[t_0,t]} \in X_{[t_0,t]}(\Omega) \subset \mathcal{X}_{[t_0,t]}$ :

$$w_{j,t}(x_{[t_0,t]}) := W_{j,t}(\omega) \quad \text{and} \quad w_t(x_{[t_0,t]}) := W_t(\omega) \\ \text{for some } \omega \in \Omega \text{ with } X_{[t_0,t]}(\omega) = x_{[t_0,t]}.$$

It follows from (34) that

$$w_{j,t}(x_{[t_0,t]}) = 1 + \int_{t_0}^t (g_j(y_j | x_{t_j}, t_j) - 1) \gamma_j(t_j) dt_j \quad (36)$$

for each  $j$  and from (35) that

$$w_t(x_{[t_0,t]}) = \prod_{j=1}^M w_{j,t}(x_{[t_0,t]}). \quad (37)$$

The values of  $w_t$  will serve as weights in the MTU particle filter. We will call  $w_{j,t}$  the partial weights. Since in each discretization scheme which is applied to solve the integral in the formula (36) for  $w_{j,t}$  the integrand has to be evaluated, we may run into practical problems if we use it as it is written in the formula. If the density  $g_j(y_j | x_{t_j}, t_j)$  evaluates to a very small value and if we work with fixed-precision numbers, the subtraction of 1 will result in a value which may be practically equal to  $-1$ . If this error accumulates over time, we may end up with wrong values for  $w_{j,t}(x_{[t_0,t]})$ . To reduce the computational error, the integral could be split up in the following way:

$$\begin{aligned} w_{j,t}(x_{[t_0,t]}) &= 1 - \int_{t_0}^t \gamma_j(t_j) dt_j + \int_{t_0}^t g_j(y_j | x_{t_j}, t_j) \gamma_j(t_j) dt_j \\ &= 1 - \bar{\gamma}_{j,t} + \bar{w}_{j,t}(x_{[t_0,t]}) \end{aligned} \quad (38)$$

where the cumulative distribution function

$$\bar{\gamma}_{j,t} := \int_{t_0}^t \gamma_j(t_j) dt_j \quad (39)$$

is independent of  $x_{[t_0,t]}$  and where the part

$$\bar{w}_{j,t}(x_{[t_0,t]}) := \int_{t_0}^t g_j(y_j | x_{t_j}, t_j) \gamma_j(t_j) dt_j \quad (40)$$

depends on the path  $x_{[t_0,t]}$ . It is even more convenient to compute  $\bar{\gamma}_{j,t}$  by evaluating the antiderivative of  $\gamma_j$ , if it is computationally available.

Note that the definition of the filter distribution is dependent on the reference measure  $\mu_{y_j}$ . A suitable change of this measure may help to further increase the efficiency of the algorithm. This issue still has to be explored.

### Resampling

Special attention is needed for the computation of the weights after resampling steps have been applied. As mentioned earlier, resampling at time  $s_\ell$  is done by randomly generating a selection function  $\iota_\ell : I \rightarrow I$  (with index

set  $I = \{1, \dots, N\}$  based on given non-negative (unnormalized) selection weights  $v_{s_\ell}^i$  for each particle index  $i$ . The state samples  $(x_{s_\ell}^i)_{i=1, \dots, N}$  have then to be replaced by the selected state samples  $(x_{s_\ell}^{\ell(i)})_{i=1, \dots, N}$ , and the unnormalized weights  $(w_{s_\ell}^i)_{i=1, \dots, N}$  by the corrected weights  $(w_{s_\ell}^{\ell(i)} / v_{s_\ell}^{\ell(i)})_{i=1, \dots, N}$ . We assume that resampling steps at times  $s_1, \dots, s_\ell$  with  $t_0 \leq s_1 < \dots < s_\ell \leq t$  have occurred, states and weights have been replaced at these times, and within this paragraph, we denote them by  $x_t^{s_1, \dots, s_\ell; i}$  and  $w_t^{s_1, \dots, s_\ell} (x_{[t_0:t]}^{s_1, \dots, s_\ell; i})$ , respectively, for each particle  $i$ . By definition, we have at  $t = s_\ell$

$$x_{s_\ell}^{s_1, \dots, s_\ell; i} = x_{s_\ell}^{s_1, \dots, s_{\ell-1}; \ell(i)},$$

and

$$w_{s_\ell}^{s_1, \dots, s_\ell} (x_{[t_0:s_\ell]}^{s_1, \dots, s_\ell; i}) = \frac{w_{s_\ell}^{s_1, \dots, s_{\ell-1}} (x_{[t_0:s_\ell]}^{s_1, \dots, s_\ell; i})}{v_{s_\ell}^{\ell(i)}}.$$

For each time  $t \geq s_\ell$  and for each particle  $i$ , the corrected weights are then recursively given by

$$w_t^{s_1, \dots, s_\ell} (x_{[t_0:t]}^{s_1, \dots, s_\ell; i}) = \frac{w_t^{s_1, \dots, s_{\ell-1}} (x_{[t_0:t]}^{s_1, \dots, s_\ell; i})}{v_{s_\ell}^{\ell(i)}} = \frac{w_t (x_{[t_0:t]}^{s_1, \dots, s_\ell; i})}{\prod_{\lambda=1}^{\ell} v_{s_\lambda}^{\lambda(i)}} \quad (41)$$

with

$$v_{\lambda; \ell}(i) = v_\lambda \circ \dots \circ v_\ell(i).$$

Since the process  $W_t$  computes the uncorrected weights  $w_t(x_{[t_0:t]}^{s_1, \dots, s_\ell; i})$ , we have to correct the weights explicitly in the algorithm by dividing them by the cumulative product

$$\bar{v}_t^{s_1, \dots, s_\ell; i} := \prod_{\lambda=1}^{\ell} v_{s_\lambda}^{\lambda; \ell}(i).$$

Note that we have to select these products during a resampling step similarly to the selection of the states, i.e. at  $t = s_\ell$ ,

$$\bar{v}_{s_\ell}^{s_1, \dots, s_\ell; i} = \bar{v}_{s_\ell}^{s_1, \dots, s_{\ell-1}; \ell(i)} v_{s_\ell}^{\ell(i)}.$$

### The MTU particle filter algorithm

A practical MTU particle filter can be obtained by any discretization scheme based on an (arbitrary) time discretization  $t_0 = \tau_0 < \tau_1 < \dots < \tau_D$ . Similar to the standard particle filter, sampling need not necessarily be done from the state process  $X_{[t_0, \infty)}$  directly. One can instead sample from another process  $\tilde{X}_{[t_0, \infty)}$ , provided that the Radon-Nikodym derivatives

$$\varrho_{t_0}(x_{t_0}) := \frac{d\mathcal{L}_{X_{t_0}}(x_{t_0})}{d\mathcal{L}_{\tilde{X}_{t_0}}(x_{t_0})} \quad \text{and} \quad \varrho_{t|s}(x_t | x_s) := \frac{K_{s,t}(x_s, dx_t)}{\tilde{K}_{s,t}(x_s, dx_t)}$$

for each  $s, t \in [t_0, t]$  with  $s < t$  exist and can be evaluated pointwise. (In fact, it suffices that  $\varrho_{t|s}(x_t | x_s)$  exists for all

states  $x_s \in \mathcal{X}_s$  which are reachable from some initial state  $x_{t_0} \in \mathcal{X}_{t_0}$  with  $\varrho_{t_0}(x_{t_0}) \neq 0$ ). In the special case that we sample from the Markov kernels of  $X_{[t_0, \infty)}$  directly,  $\varrho_{t|s} \equiv 1$  for all  $s < t$ . Our MTU particle filter is described in algorithm 2. Here we suppress the indices  $s_1, \dots, s_\ell$  in the notations of states and weights (see last paragraph).

The algorithm as it is written here has to be enriched with concrete discretization methods for the sampling and update steps. For instance, if the process  $\tilde{X}_{[t_0, \infty)}$  is a multidimensional Itô process defined through a stochastic differential equation (SDE)

$$d\tilde{X}_t = a(\tilde{X}_t, t) dt + B(\tilde{X}_t, t) d\mathcal{W}_t$$

with drift  $a(x, t)$ , diffusion matrix  $B(x, t)$ , multidimensional standard Wiener process  $\mathcal{W}_t$ , and initial variable  $\tilde{X}_{t_0}$ , then the Euler-Maruyama method can be used for discretization. We set  $\Delta\tau_d := \tau_d - \tau_{d-1}$ . The sampling is then done by

$$x_{\tau_d}^i = a(x_{\tau_{d-1}}^i, \tau_{d-1}) \Delta\tau_d + B(x_{\tau_{d-1}}^i, \tau_{d-1}) \sqrt{\Delta\tau_d} \eta^i \quad (42)$$

where  $\eta^i$  is a sample from a standard normal distribution (with mean 0 and variance 1). Further, the update step can be done in the simplest case using Euler discretization:

$$\bar{y}_{j, \tau_d} = \bar{y}_{j, \tau_{d-1}} + \gamma_j(\tau_{d-1}) \Delta\tau_d \quad (43)$$

and

$$\bar{w}_{j, \tau_d}^i = \bar{w}_{j, \tau_{d-1}}^i + g_j(y_j | x_{\tau_{d-1}}^i, \tau_{d-1}) \gamma_j(\tau_{d-1}) \Delta\tau_d. \quad (44)$$

Of course, better discretization schemes are possible. As we have already mentioned, if the antiderivative  $G_j$  with  $G_j(t_0) = 0$  (which is in fact the distribution function) of  $\gamma_j$  is available, then one should rather use

$$\bar{y}_{j, \tau_d} = G_j(\tau_d)$$

for the computation of the values  $\bar{y}_{j, \tau_d}$ .

### Adaptive stepsize

To be able to fully exploit our MTU particle filter method, the discretization stepsize must be chosen appropriately. One simple possibility is to use a very small stepsize throughout the complete procedure. A quite high computation time will result from that. This can be reduced if an adaptive stepsize is chosen. We propose to determine the stepsize  $\Delta\tau_d$  online depending on the ESS estimate. The stepsize should decrease when the ESS drops rapidly, and it should increase again if the ESS estimate changes only marginally. In detail, the following procedure can be applied.

In each step of the algorithm, we obtain an initial guess of the stepsize by a linear interpolation between a maximal stepsize if the ESS had not changed since the last step, and a minimal stepsize if the ESS had dropped by the number  $N$  of samples (actually, the maximal difference that can be

---

**Algorithm 2** MTU particle filter

---

```

1: {At time  $\tau_0 = t_0$ :}
2:   Sample  $N$  state realizations  $x_{\tau_0}^i, i = 1, \dots, N$ , of  $\tilde{X}_{\tau_0}$ , with  $N$  large.
3:   for  $j = 1, \dots, M$  do
4:     Set  $\bar{\gamma}_{j, \tau_0} = 0$ .
5:   end for
6:   for  $i = 1, \dots, N$  do
7:     Set the weight  $w_{\tau_0}^i = \varrho_{t_0}(x_{\tau_0}^i)$ .
8:     for  $j = 1, \dots, M$  do
9:       Set  $\bar{w}_{j, \tau_0}^i = 0$ .
10:    end for
11:    Set the cumulative product  $\bar{v}^i$  of the selection weights to 1.
12:  end for
13: for all times  $\tau_d, d = 1, \dots, D$ : do
14:   {Resample the particle set  $(x_{\tau_{d-1}}^i)_{i=1, \dots, N}$  if necessary (e.g. if the ESS drops below a threshold):}
15:   Generate a selection function  $\iota$  according to some selection weights  $v_{\tau_{d-1}}^i$ .
16:   for  $i = 1, \dots, N$  do
17:     Replace the state sample  $x_{\tau_{d-1}}^i$  by the selected state samples  $x_{\tau_{d-1}}^{\iota(i)}$ .
18:     for  $j = 1, \dots, M$  do
19:       Replace the partial weight  $\bar{w}_{j, \tau_{d-1}}^i$  by the selected partial weight  $\bar{w}_{j, \tau_{d-1}}^{\iota(i)}$ .
20:     end for
21:     Replace the cumulative product of the selection weights  $\bar{v}^i$  by  $\bar{v}^{\iota(i)} v_{\tau_{d-1}}^{\iota(i)}$ .
22:   end for
23:   for  $i = 1, \dots, N$  do
24:     Sample (at least approximately) a realization  $x_{\tau_d}^i$  from the Markov kernel
25:      $\tilde{K}_{\tau_{d-1}, \tau_d}(x_{\tau_{d-1}}^i, \cdot)$ .
26:     for  $j = 1, \dots, M$  do
27:       Compute (at least approximately)
28:       
$$\bar{\gamma}_{j, \tau_d} = \bar{\gamma}_{j, \tau_{d-1}} + \int_{\tau_{d-1}}^{\tau_d} \gamma_j(t) dt \quad \text{or (if available)} \quad \bar{\gamma}_{j, \tau_d} = \int_{t_0}^{\tau_d} \gamma_j(t) dt$$

29:       and
30:       
$$\bar{w}_{j, \tau_d}^i = \bar{w}_{j, \tau_{d-1}}^i + \int_{\tau_{d-1}}^{\tau_d} g_j(y_j | x_t^i, t) \gamma_j(t) dt.$$

31:     end for
32:     Set the total unnormalized weight to
33:     
$$w_{\tau_d}^i = \frac{\varrho_{\tau_d | \tau_{d-1}}(x_{\tau_d}^i | x_{\tau_{d-1}}^i)}{\bar{v}^i} \prod_{j=1}^M (1 - \bar{\gamma}_{j, \tau_d} + \bar{w}_{j, \tau_d}^i).$$

34:   end for
35:   For given functions  $h$ , compute approximations to the expectations of the filtered process:
36:   
$$\hat{E}_{\tau_d}[h(X_{\tau_d}) | Y_{1:M} = y_{1:M}] \approx \frac{\sum_{i=1}^N w_{\tau_d}^i h(x_{\tau_d}^i)}{\sum_{i=1}^N w_{\tau_d}^i}.$$

37: end for

```

---

obtained is  $N - 1$ ). From this initial guess, we compute the increments of the partial weights, and from them we predict the ESS in the next step based on the current stepsize guess. If the difference between this predicted ESS and the current ESS drops by more than a certain relative amount (we use 10%), then a new guess of the stepsize is computed by dividing the current guess by 2. With this new guess, a new predicted ESS is computed, and the test can be applied again. This procedure will be applied iteratively until either the difference between predicted and current ESS drops by less than the prescribed amount, or the stepsize guess falls below a prescribed minimal stepsize. The current stepsize or the minimal stepsize, respectively, is then accepted, and the algorithm proceeds with this stepsize. See algorithm 3 for a formal description of the stepsize determination.

If we sample from the Markov kernels of  $X_{[t_0, \infty)}$  directly, then  $Q_{\tau_d^* | \tau_{d-1}}(x_{\tau_d^*}^i | x_{\tau_{d-1}}^i) \equiv 1$  and the update of the weights does not depend on the new states  $x_{\tau_d^*}^i$ . Hence we only need to compute the weight update and the corresponding ESS estimate until we find an adequate stepsize. In this case, it is not necessary to sample the new states in each iteration which renders the algorithm computationally more effective.

Note that this procedure cannot be performed when the measurement times are fixed (i.e. in the standard particle filter). In this case the ESS does not depend on the stepsize, and a reduction of it will not improve the ESS. The application of the MTU particle filter with distributed measurement times is essential to be able to use this adaptive stepsize procedure.

#### Data Likelihood

As mentioned earlier for the standard case of the particle filter algorithm, without resampling, the data likelihood could be approximated by the empirical mean of the unnormalized weights (see (19)). In our case, this would be

$$\hat{Z}_t := \frac{1}{N} \sum_{i=1}^N w_t^i = \frac{1}{N} \sum_{i=1}^N w_t(x_t^i).$$

After each resampling step, we have to correct this formula. Using the same notation as in the paragraph on resampling, for each time  $t \geq s_\ell$  and for each particle  $i$ , the corrected estimate is given by

$$\begin{aligned} \hat{Z}_t &:= \frac{1}{N^{\ell+1}} \left( \prod_{\lambda=1}^{\ell} \sum_{i=1}^N v_{s_\lambda}^i \right) \sum_{i=1}^N w_t^{s_1, \dots, s_\ell} (x_{[t_0:t]}^{s_1, \dots, s_\ell; i}) \\ &= \frac{1}{N^{\ell+1}} \left( \prod_{\lambda=1}^{\ell} \sum_{i=1}^N v_{s_\lambda}^i \right) \sum_{i=1}^N \frac{w_t(x_{[t_0:t]}^{s_1, \dots, s_\ell; i})}{\prod_{\lambda=1}^{\ell} v_{s_\lambda}^{\iota_{\lambda:\ell}(i)}} \end{aligned} \quad (45)$$

#### Algorithm 3 Determination of the adaptive stepsize

1: Compute an initial guess of the stepsize:

$$\Delta \tau_d^* = \Delta \tau_{\max} - (\Delta \tau_{\max} - \Delta \tau_{\min}) \cdot \frac{|n_{\text{ESS}, \tau_{d-2}} - n_{\text{ESS}, \tau_{d-1}}|}{N}.$$

2: Set  $\tau_d^* = \tau_{d-1} + \Delta \tau_d^*$ .

3: **loop**

4: **for**  $i = 1, \dots, N$  **do**

5: Sample (at least approximately) a realization  $x_{\tau_d^*}^i$  from the Markov kernel

$$\tilde{K}_{\tau_{d-1}, \tau_d^*}(x_{\tau_{d-1}}^i, \cdot).$$

6: **for**  $j = 1, \dots, M$  **do**

7: Update the weights:

$$\bar{y}_{j, \tau_d^*} = \bar{y}_{j, \tau_{d-1}} + \gamma_j(\tau_{d-1}) \Delta \tau_d^*$$

and

$$\bar{w}_{j, \tau_d^*}^i = \bar{w}_{j, \tau_{d-1}}^i + g_j(x_{\tau_{d-1}}^i, \tau_{d-1}) \gamma_j(\tau_{d-1}) \Delta \tau_d^*.$$

8: **end for**

9: Set the total unnormalized weights to

$$w_{\tau_d^*}^i = \frac{Q_{\tau_d^* | \tau_{d-1}}(x_{\tau_d^*}^i | x_{\tau_{d-1}}^i)}{\bar{v}^i} \prod_{j=1}^M (1 - \bar{y}_{j, \tau_d^*} + \bar{w}_{j, \tau_d^*}^i).$$

10: **end for**

11: Compute the tentative ESS estimate:

$$n_{\text{ESS}, \tau_d^*} = \frac{1}{\sum_{i=1}^N (\bar{w}_{\tau_d^*}^i)^2}$$

where

$$\bar{w}_{\tau_d^*}^i = \frac{w_{\tau_d^*}^i}{\sum_{i=1}^N w_{\tau_d^*}^i}$$

are the normalized weights.

12: **if**  $\frac{|n_{\text{ESS}, \tau_{d-1}} - n_{\text{ESS}, \tau_d^*}|}{n_{\text{ESS}, \tau_{d-1}}} > 0.1$  and  $\Delta \tau_d^* > \Delta \tau_{\min}$  **then**

13: Set  $\Delta \tau_d^*$  to  $\max\left(\frac{\Delta \tau_d^*}{2}, \Delta \tau_{\min}\right)$  and repeat the loop.

14: **else**

15: Set  $\Delta \tau_d = \Delta \tau_d^*$  and  $\tau_d = \tau_d^*$  and **break**.

16: **end if**

17: **end loop**

with

$$\iota_{\lambda:\ell}(i) = \iota_\lambda \circ \dots \circ \iota_\ell(i)$$

(see (41)). This can be seen by considering recursively the correction at the time of the resampling step  $\ell$ . Since

$$\frac{N v_{s_\ell}^{\iota_\ell(i)}}{\sum_{v=1}^N v_{s_\ell}^v}$$

is the expected number of times the particle  $i$  has been selected after  $N$  draws, we have to correct the weights  $w_{s_\ell}^{s_1, \dots, s_{\ell-1}}(x_{[t_0:s_\ell]}^{s_1, \dots, s_\ell; i})$  in selection step  $\ell$  by dividing them by

this number. The following computation then proves the correctness of the formula via induction:

$$\begin{aligned} \hat{Z}_{s_\ell} &= \frac{1}{N^\ell} \left( \prod_{\lambda=1}^{\ell-1} \sum_{i=1}^N v_{s_\lambda}^i \right) \\ &\quad \times \sum_{i=1}^N \left( w_{s_\ell}^{s_1, \dots, s_{\ell-1}} \left( x_{[t_0:s_\ell]}^{s_1, \dots, s_\ell; i} \right) / \frac{N v_{s_\ell}^{\ell(i)}}{\sum_{v=1}^N v_{s_\ell}^v} \right) \\ &= \frac{1}{N^{\ell+1}} \left( \prod_{\lambda=1}^{\ell} \sum_{i=1}^N v_{s_\lambda}^i \right) \sum_{i=1}^N \frac{w_{s_\ell}^{s_1, \dots, s_{\ell-1}} \left( x_{[t_0:s_\ell]}^{s_1, \dots, s_\ell; i} \right)}{v_{s_\ell}^{\ell(i)}} \\ &= \frac{1}{N^{\ell+1}} \left( \prod_{\lambda=1}^{\ell} \sum_{i=1}^N v_{s_\lambda}^i \right) \sum_{i=1}^N w_{s_\ell}^{s_1, \dots, s_\ell} \left( x_{[t_0:s_\ell]}^{s_1, \dots, s_\ell; i} \right). \end{aligned}$$

We get algorithm 4 for the computation of the empirical estimate of the data likelihood, which needs to be done in parallel to the MTU particle filter algorithm.

#### Offline and online estimation

Two main cases of estimation procedures may be distinguished: the offline estimation procedure used for example for parameter estimation, and the online estimation procedure for control purposes. While in the offline case the measurements are completely available before estimation begins, we know only some of the measurement values at some certain time  $t$  during the online procedure. In this latter case, an online computation where all measurement times have been modelled with probability densities with infinite support is impossible, because in this situation all measurement values must already be known at time  $t_0$ . Online estimation is nevertheless possible if the

---

#### Algorithm 4 Estimation of the data likelihood

---

- 1: At time  $\tau_0 = t_0$ :
- 2: Set the correction factor for the data likelihood to  $\bar{Z}_{\tau_0} := 1/N$ .
- 3: **for all** times  $\tau_d, d = 1, \dots, D$  **do**
- 4:     **if** the particle set has been resampled **then**
- 5:         Set the correction factor for the data likelihood to

$$\bar{Z}_{\tau_d} := \bar{Z}_{\tau_{d-1}} \frac{1}{N} \sum_{i=1}^N v_{\tau_{d-1}}^{(i)}.$$

- 6:     **else**
- 7:         Set
- 8:          $\bar{Z}_{\tau_d} := \bar{Z}_{\tau_{d-1}}$ .
- 9:     **end if**
- 9:     Compute the empirical estimate for the data likelihood at time  $\tau_d$  by

$$\hat{Z}_{\tau_d} := \bar{Z}_{\tau_d} \sum_{i=1}^N w_{\tau_d}^i.$$

- 10: **end for**
- 

support of the measurement time densities is finite. The online estimation must then be delayed by the diameter of the respective supports.

#### Implementation

We have implemented the proposed algorithm in Mathematica as part of a Parameter Estimation Toolbox for Systems Biology developed by the Systems Biology group at the Fraunhofer Chalmers Centre (FCC) in Göteborg (Sweden). Furthermore, we have implemented it also in the statistical computing language R [28]. All figures in this article have been created using this R implementation.

## Results and Discussion

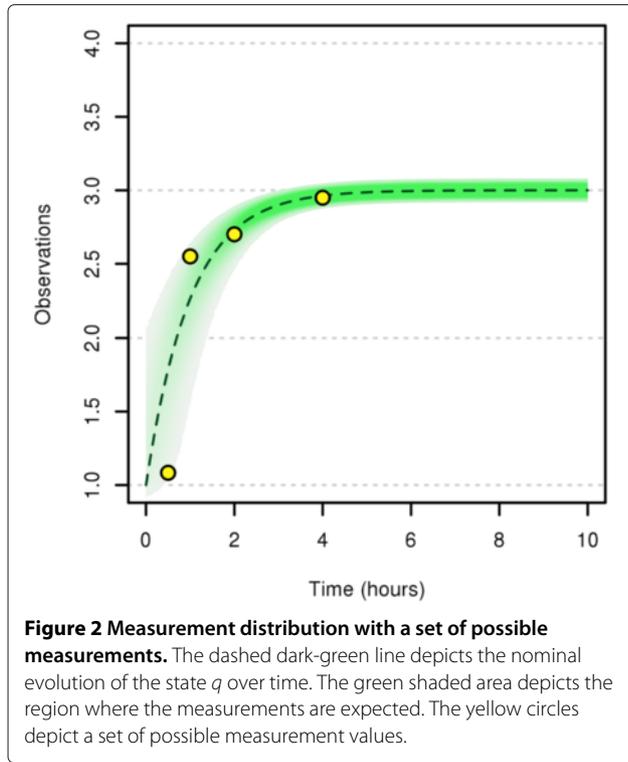
### Motivating example - results

In this section, we resume our motivating example and use it in a parameter estimation setting to compare our MTU filter to both the standard particle filter and to a state-of-the-art Maximum Likelihood (ML) method which is not based on Monte Carlo techniques. To this aim, we first create virtual “measurements” at four intended measurement times  $\hat{t}_j, j = 1, \dots, 4$  by simulation runs with our true model (see Table 1). This is done as follows: We simulate a single state path  $(q(t))_{t \in [0,10]}$  based on the “true” parameter values  $\alpha = 1$  and  $\beta = 3$ . Then, for each intended measurement time  $\hat{t}_j$ , we sample an actual measurement time  $t_j$  according to the density  $\gamma_j$ . Now, for each  $t_j$ , we sample a measurement value  $y_j$  from the distribution  $\mathcal{N}(q(t_j), \sigma_y^2)$ . Figure 2 shows one set of measurement values obtained in this way. The intended measurement times are 0.5, 1, 2, 4 and the measurement values we got from one simulation run are 1.083346, 2.550290, 2.700863, 2.949450. We will use these values in the following estimation runs. Our aim is to estimate the parameter vector  $\theta = (\alpha, \beta)$  with the following three different estimation procedures and compare the results:

1. ML estimation on “lumped” measurements,
2. Standard PF on “lumped” measurements, and
3. MTU-PF.

Note that in both the standard PF and the ML estimation method, it is not possible to include the uncertainties in the measurement times directly. Rather, we have to lump these uncertainties with the uncertainties in the measurement values by increasing their variances  $\sigma_y^2$  and to treat the  $\hat{t}_j$  like true measurement times. In both cases, we tested several values of “lumped” variances  $\sigma_y^2$ .

We have implemented the standard particle filter and the MTU particle filter in the statistical computing language R [28]. The ML estimations have been done using the Parameter Estimation Toolbox for Systems Biology developed by the Systems Biology group at the Fraunhofer Chalmers Centre (FCC) in Göteborg (Sweden).



### ML estimation

The ML estimation is a standard method based on a Maximum Likelihood approach, that is, on the maximization of the likelihood function  $L(\theta; y_{1:M})$  with respect to the parameter vector  $\theta$ . The computation of the likelihood requires the use of estimates of the state mean and covariance matrix which are routinely obtained by the use of a non-linear derivation of the Kalman filter.

Given measurements  $y_{1:M}$  at a total of  $M$  time points  $t_j$  which are assumed to be fixed and known, the likelihood function can be written as

$$L(\theta; y_{1:M}) := f^{Y_{1:M}, T_{1:M}, \theta}(y_{1:M}; \theta) = \prod_{j=1}^M f^{Y_j | Y_{1:j-1}, T_{1:M}, \theta}(y_j | y_{1:j-1}, t_{1:M}; \theta) \quad (46)$$

where  $f^{Y_j | Y_{1:j-1}, T_{1:M}, \theta}$  is the conditional probability density function of the  $j$ -th observation  $y_j$  given all previous observations  $y_{1:j-1}$  as given in (9), here explicitly based on the given parameter vector  $\theta$ . As already mentioned, analytical solutions are not generally available. An exception to this is the case that all measurements are conditionally Gaussian, that means that all conditional densities  $f^{Y_j | Y_{1:j-1}, T_{1:M}, \theta}$  are Gaussian. In this case, the Kalman filter yields the correct solution. The assumption that the measurements are conditionally Gaussian is commonly used as an approximation even in those cases where this is not

true. If one accepts that this assumption gives an approximation which is close enough to the true model, then it is only necessary to propagate the means and covariances of the measurements, since the Gaussian probability distribution is completely characterized by the first two moments. We introduce the notation

$$\hat{y}_{j|j-1} = E[y_j | y_{1:j-1}; \theta]$$

and

$$R_{j|j-1} = \mathbf{Cov}[y_j | y_{1:j-1}; \theta]$$

for the prediction of the mean and covariance of the observation variables, respectively. The computation of these values can be achieved by some derivatives of the Kalman filter, commonly used are the Extended Kalman Filter (EKF) or the Unscented Kalman Filter (UKF), more exactly those versions of these filters which are time-continuous in the states and time-discrete in the measurements (continuous/discrete EKF and UKF, respectively). The residuals  $\epsilon_j$  are then defined by the differences between the measurements  $y_j$  and their estimations  $\hat{y}_{j|j-1}$ :

$$\epsilon_j := y_j - \hat{y}_{j|j-1}.$$

The assumption of normally distributed observations leads to an approximation of the likelihood function of the following form

$$L(\theta; y_{1:M}) \approx \hat{L}(\theta; y_{1:M}) := \prod_{j=1}^M \frac{\exp\left(-\frac{1}{2}\epsilon_j^T R_{j|j-1}^{-1} \epsilon_j\right)}{\sqrt{|R_{j|j-1}|} (\sqrt{2\pi})^l}$$

where  $l$  is the number (dimension) of the observation variables. The negative logarithm of this approximated likelihood function is

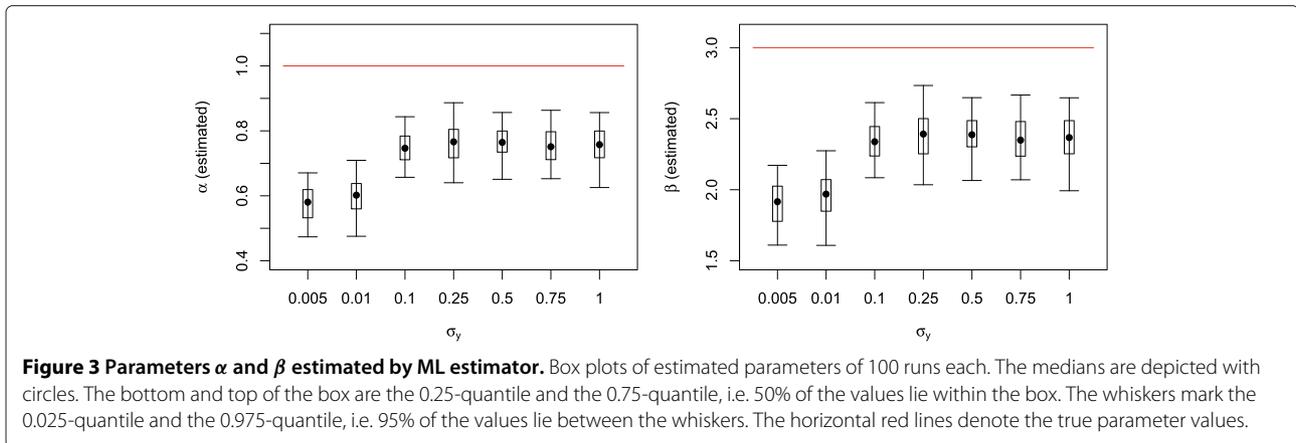
$$-\log \hat{L}(\theta; y_{1:M}) = \frac{1}{2} \sum_{j=1}^M \left( \log(\det(R_{j|j-1})) + \epsilon_j^T R_{j|j-1}^{-1} \epsilon_j \right) + \frac{Ml}{2} \log(2\pi).$$

The problem to finding maximum likelihood estimates of the model parameters takes the form of a nonlinear optimization problem:

$$\hat{\theta} = \arg \min_{\theta \in \Theta} \{-\log \hat{L}(\theta; y_{1:M})\}.$$

Roughly, the estimation with this approach is done as follows:

1. Choose an initial guess  $\theta^*$  for the parameter vector  $\theta$ .
2. Based on the current parameter vector  $\theta^*$  and for each (intended) measurement time  $t_j$ , compute residuals  $\epsilon_j$  and estimates of the covariance matrices  $R_{j|j-1}$  using (an approximation to) the Kalman filter.
3. Compute the likelihood  $L$  based on the state estimates  $\epsilon_j$  and  $R_{j|j-1}$ .



4. Use one step of some local optimization technique to find an improved parameter vector  $\theta_{\text{new}}^*$  that decreases the negative log-likelihood function; replace  $\theta^*$  by this improved parameter vector  $\theta_{\text{new}}^*$ .
5. Repeat steps 2 to 4 until the parameter vector shows convergence.

The results of several runs of parameter estimations for different values of the lumped measurement variances  $\sigma_y^2$  are shown in Figure 3. For each  $\sigma_y^2$ , we performed 100 runs with different initial state and parameter values. Note that the choice of an initial parameter value is mandatory for all approaches based on a local optimization method, and that different choices may lead to different results. On the other hand, in our example, we assume a log-normal prior distribution for the initial states (see Table 1), while the Kalman filter based approximations always assume normally distributed initial state values. Conditioned on the initial state (and current parameter), our example system is Gaussian linear which means that the Kalman filter yields exact estimates. We therefore decided to use the following procedure: Before each run, we sample the starting values for the parameters from the same priors as are available to the particle filter (see Table 2), and we also sample initial values  $q_0$  for the states from the correct distributions (see Table 1). Then we start the estimation procedure based on these initial parameter and state samples, and we keep the resulting estimated parameters as one “sample”. We repeat this procedure for all of the 100 runs. In this way, the standard procedure uses exactly the same information as the particle filters, with the only exception of the measurement time distributions  $\gamma_j$ . Figure 3 shows the distributions of the estimated parameter values as box plots.

The estimated parameters of all estimation runs are far away from the true values; even quite large variances  $\sigma_y^2$  do not improve the performance of the estimations. (We also tested the ML approach on a different data set of 4 measurements which was created *without* randomness in the

measurement times: the ML approach performed much better in this case since the estimated parameters varied around the true values; data not shown).

#### Estimation with particle filters

Tests with the particle filters (both standard and MTU) have been performed in the following way: First we perform an *estimation run* where we use these measurements to estimate the parameters  $\alpha$  and  $\beta$  of the system equation  $dq(t) = (-\alpha q(t) + \beta) dt + \sigma dW_t$ . We perform this estimation with the MTU particle filter and with the standard particle filter under the same conditions and with the same seed for the random number generator. Thus the initial distribution of the particles is the same in both cases. For the standard filter, we use different levels of variance  $\sigma_y^2$  for the measurement noise. Afterwards, the empirical medians of the final parameter distributions are used to perform a *simulation run* in each case, where we estimate the state  $q$  of the system based on the measurements and these fixed parameters. For both runs, we compute an estimate of the data likelihood over time. The choices of stepsizes and artificial parameter dynamics can be found in Table 2 (the values for the artificial parameter dynamics are computed by interpolation in the same way as we will describe later in the application of the MTU-PF to the plasma-leucine model).

Table 3 shows a comparison of the estimated parameters and data log-likelihood values. The estimated parameters

**Table 2 Choices for the PFs for the motivating example**

|  |                                 |
|--|---------------------------------|
| prior for estimation of $\alpha$                               | $\log\mathcal{N}(\log(2), 1^2)$ |
| prior for estimation of $\beta$                                | $\log\mathcal{N}(\log(6), 1^2)$ |
| stepsize for standard particle filter                          | $10^{-2}$                       |
| maximal stepsize for MTU particle filter                       | $10^{-2}$                       |
| minimal stepsize for MTU particle filter                       | $10^{-6}$                       |
| diffusion parameter for artificial parameter noise at time $t$ | $5.43/(t + 3.29)^2$             |

**Table 3 Estimated parameters and data log-likelihood values for the MTU particle filter and the standard particle filter with different standard deviations  $\sigma_y$  for the measurement noise**

|          | $\sigma_y$ | $\alpha$ | $\beta$ | Data log-likelihood |                |
|----------|------------|----------|---------|---------------------|----------------|
|          |            |          |         | Estimation run      | Simulation run |
| MTU      | 0.005      | 1.012    | 3.010   | -4.327              | 2.398          |
| standard | 0.005      | 7.031    | 20.712  | -24.084             | -744.970       |
| standard | 0.01       | 9.102    | 26.919  | -23.936             | -890.746       |
| standard | 0.1        | 4.709    | 13.847  | -9.081              | -140.117       |
| standard | 0.25       | 1.425    | 4.171   | -6.714              | -4.618         |
| standard | 0.5        | 1.156    | 3.287   | -5.538              | -2.170         |
| standard | 0.75       | 1.318    | 3.604   | -5.807              | -3.160         |
| standard | 1          | 1.450    | 3.733   | -6.227              | -4.100         |

The true parameters are  $\alpha = 1$  and  $\beta = 3$ .

(empirical medians) of the MTU particle filter ( $\alpha = 1.012$  and  $\beta = 3.010$ ) are very close to the true values ( $\alpha = 1$  and  $\beta = 3$ ) whereas the estimated values of the standard particle filter are significantly worse. The data log-likelihood in both estimation and simulation run is considerably higher for the MTU filter, compared to any of the standard filter runs.

In Figure 4, we show the distributions of the filtered states obtained by simulation runs with the estimated parameters. The violet shaded area in Figure 4(a) depicts the estimated state distributions of the state  $q$  for the MTU particle filter with the light-blue line marking its median. The dashed dark-green line depicts the nominal evolution of the state  $q$  over time which should be approximated by the filter. Figures 4(b-d) show the estimated state distributions of the state  $q$  for the standard particle filter with different assumed lumped measurement variances. As can be seen from these figures, the approximation by the MTU particle filter is very close to the evolution of the state with the true parameters. On the contrary, however we choose the measurement variance in the standard case, the algorithm is not able to adequately approximate the correct state evolution.

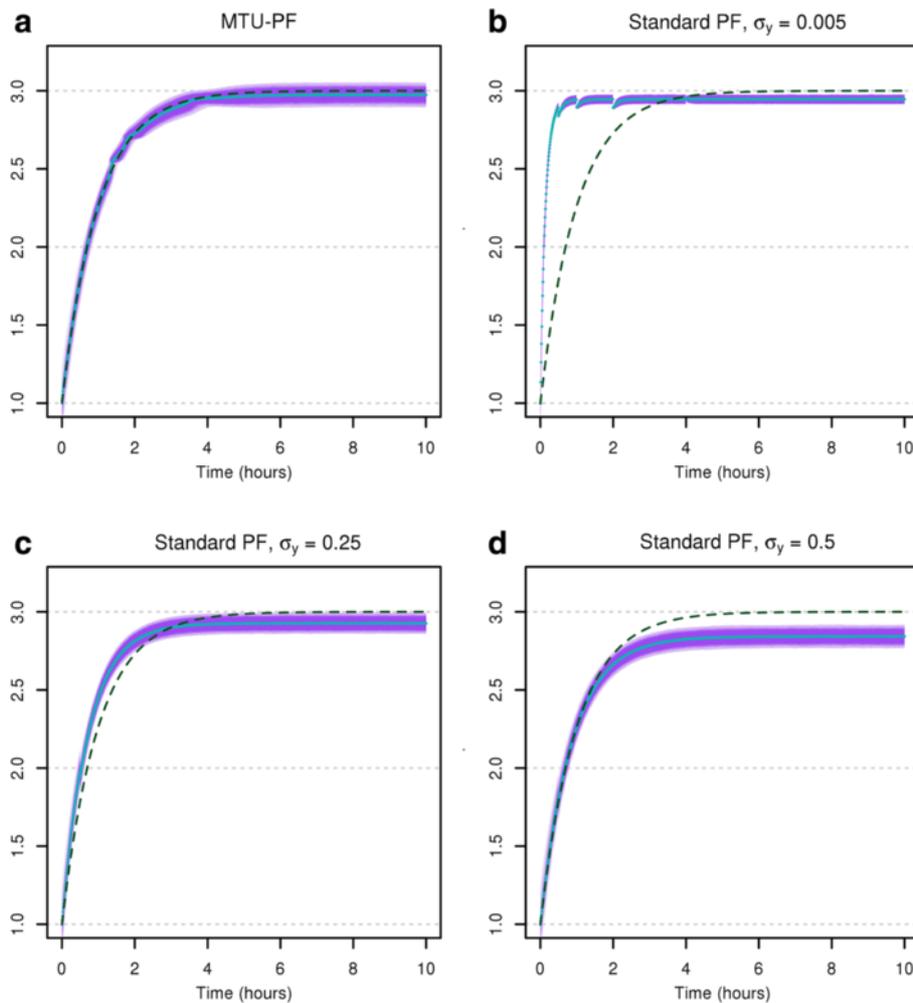
Figure 5 displays the simulated measurement distributions corresponding to the filtered state distributions based on the simulation runs with estimated parameters. Here we notice again that the MTU particle filter can adapt well to the situation, whereas the measurement distributions which can be realized by the standard filter do not fit well with the data points.

In addition to the significant improvement in the parameter and state estimation, the MTU particle filter has another benefit. Figure 6 shows the development of the effective sample size estimate during the estimation runs. In the standard particle filter runs where the lumped measurement variance is assumed to be small, the predicted values for states and parameters do not fit well with

the measurements. At the measurement times, when the predicted states are compared with the measurements and weighted accordingly, most weights decrease rapidly. This leads to a high variance of the weights and the effective sample size estimate drops severely, indicating that only very few particles effectively contribute to the estimation. This degeneration of the particle cloud is prevented by the MTU particle filter, as can be seen in Figure 6(a). For the standard filter runs where  $\sigma_y^2$  is assumed to be larger, the ESS estimate does not drop as severely as with a small  $\sigma_y^2$ . This is due to the fact that, with a measurement variance which is assumed to be very high, more or less all possible simulated measurements are considered to fit well with the true measurements. However, these filters are not able to establish reasonably good estimates of the states and parameters and are clearly outperformed by the MTU particle filter.

#### Application to the Plasma-Leucine Model with Population Data

In this section, we apply our MTU particle filter algorithm to a leucine kinetics model (Demant et al. [29] based on Cobelli et al. [30]) with data taken from a clinical study on diabetes patients ([31,32], see Additional file 1). We perform Bayesian population-based (i.e. NLME) parameter estimation with this model. The model and the data have been previously used in a maximum likelihood estimation context in [17]. It should be noted that the original ODE model needs to be extended by some kind of stochastic process variability in order to turn it into an SDE model. The approach taken in [17] is different from our approach in the way that in [17] the stochastic fluctuations are assumed to be in the tracee (plasma leucine) while we here assume the variability to be in the tracer (labelled leucine). Both assumptions are plausible; a final decision on the best way to model the process dynamics has not yet been made.



**Figure 4** Filtered state distributions based on simulation runs with estimated parameters for the motivating example. MTU particle filter (a) and standard particle filter with several different assumed lumped measurement variances  $\sigma_y$  (b-d). Violet shaded area: Filtered state distributions based on empirical quantiles. Solid blue line: Median of filtered states. Dashed dark-green line: nominal evolution of the state  $q$  over time.

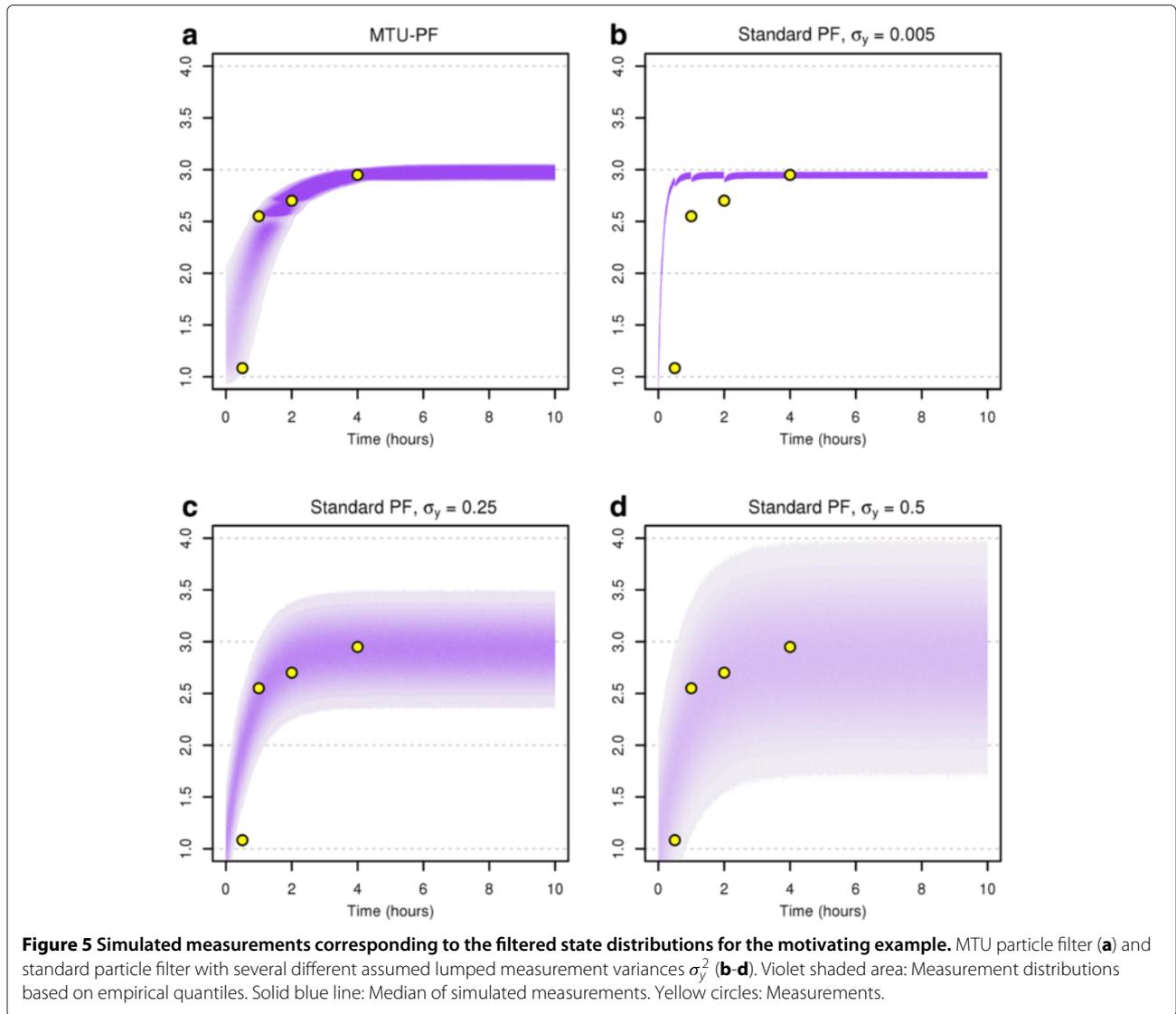
### The Leucine model

In [31] (see also the thesis [33]), a new combined multicompartamental model for apolipoprotein B-100 (apoB) and triglyceride metabolism in very low density lipoprotein (VLDL) subfractions has been developed, see Figure 7. VLDL are transporters of triglycerides and cholesterol from the liver to the periphery, and elevated levels are associated with increased risk for cardiovascular events. Each VLDL particle has exactly one apoB molecule attached which makes apoB a suitable marker for triglyceride transport.

The secreted particles become denser and denser as triglycerides are delivered to target organs such as muscles and adipose tissue and the relative protein content is increased. As the density increases, the VLDL becomes an intermediate density lipoprotein (IDL) and finally a low density lipoprotein (LDL).

For our purposes we use only the part of the model concerning the leucine pool (compartments 1-4), see Figure 8. This subsystem was first used for apoB kinetic studies by Demant et al. [29] as an implementation of the work by Cobelli et al. [30]. The output is at compartment 1 and at compartment 2. The influx into compartment 1 will be denoted  $U_1$ .

The data are obtained from tracer/tracee experiments. Here the tracee (i.e. the concentration we are actually interested in) is the leucine amino acids in the apoB molecules. Additional labelled leucine (tracer) is injected as a bolus infusion. Knowledge about the kinetics (fluxes between compartments) of the tracee can be gained by studying the kinetics of the tracer. In the restricted model four compartments are considered: plasma leucine (1), intra-hepatic leucine (2), and two plasma protein pools (3 and 4). In the full model, additional compartments



represent VLDL subfractions (compartments 5-11 in Figure 7). The four-compartment system is linked to compartments 5-11 through compartment 2.

For each compartment  $i$  where  $i = 1, \dots, 4$ , let  $Q_i$  and  $q_i$  denote the mass of the tracee and the tracer, respectively. Similarly, let  $U_i$  and  $u_i$  denote the input for the tracee and the tracer, respectively. For tracer/tracee experiments,  $Q_i$  is assumed to be in steady state. If the concentration level of the labelled injection is small compared with the overall concentration levels, and if the model is linear, then approximately

$$\frac{dq(t)}{dt} = K(t)q(t) + u(t)$$

where  $q(t) = (q_i(t))_{i=1,2,3,4}^T$ ,  $u(t) = (u_1(t), 0, 0, 0)^T$  and

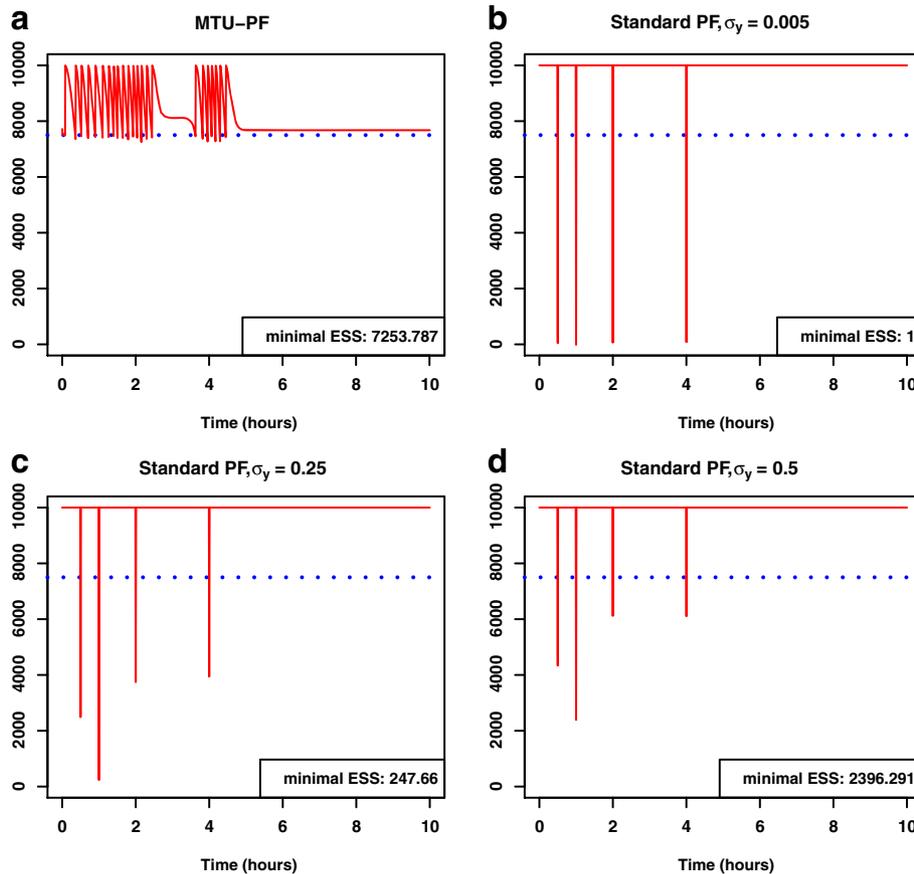
$$K(t) = (k_{j,i})_{j,i=1,2,3,4}$$

where  $k_{j,i}$  for  $i \neq j$  is the transfer coefficient of the tracers from compartment  $i$  to compartment  $j$  (compartments 0 and 11 are considered to be output compartments), and for each  $i = 1, \dots, 4$

$$k_{i,i} := - \sum_{\substack{j=0,1,2,3,4,11 \\ j \neq i}} k_{j,i}.$$

Throughout this paper, the time unit used is hours, all fractional transfer coefficients are given in the unit  $h^{-1}$ , and the amount of material in compartments is given in mg. In our model, only  $k_{0,1}$ ,  $k_{1,2}$ ,  $k_{1,3}$ ,  $k_{2,1}$ ,  $k_{3,1}$ ,  $k_{3,4}$ ,  $k_{4,3}$  and  $k_{11,2}$  are assumed to be non-zero, while additionally the following dependencies of the transfer coefficients are assumed to be valid:

$$\begin{aligned} k_{1,2} &= k_{2,1} \\ k_{3,4} &= 0.1 \cdot k_{4,3} \end{aligned}$$



**Figure 6** Estimated effective sample size during estimation run. MTU particle filter (a) and standard particle filter with several different assumed lumped measurement variances  $\sigma_y$  (b-d). The dotted blue line denotes the resampling threshold.

The fractional transfer coefficient  $k_{11,2}$  has to be fixed for the system to be identifiable. We set  $k_{11,2} = 0.01\text{h}^{-1}$ , as an estimated average from current results. We build stochastic differential equations (SDEs) from the resulting ordinary differential equations by adding noise terms which are given by standard Wiener processes  $\mathcal{W}_{1,t}, \dots, \mathcal{W}_{4,t}$  multiplied by diffusion parameters  $\sigma_1, \dots, \sigma_4$ , respectively. The leucine pool subsystem which we consider here (compartments 1-4) interacts with the surroundings only via an initial input into compartment 1 at time  $t = 0$ , an output from compartment 1, and another output from compartment 2 (towards compartment 11). All other flows are processes acting inside the subsystem and hence should follow the principle of mass conservation. Therefore we add the stochastic terms to the ODE system in the following way:

$$\begin{aligned} dq_1(t) &= - (k_{1,2} + k_{0,1} + k_{3,1}) (q_1(t) dt + \sigma_1 d\mathcal{W}_1(t)) \\ &\quad + k_{1,2} (q_2(t) dt + \sigma_2 d\mathcal{W}_2(t)) \\ &\quad + k_{1,3} (q_3(t) dt + \sigma_3 d\mathcal{W}_3(t)), \\ dq_2(t) &= - (k_{11,2} + k_{1,2}) (q_2(t) dt + \sigma_2 d\mathcal{W}_2(t)) \\ &\quad + k_{1,2} (q_1(t) dt + \sigma_1 d\mathcal{W}_1(t)), \end{aligned}$$

$$\begin{aligned} dq_3(t) &= - (k_{1,3} + k_{4,3}) (q_3(t) dt + \sigma_3 d\mathcal{W}_3(t)) \\ &\quad + 0.1k_{4,3} (q_4(t) dt + \sigma_4 d\mathcal{W}_4(t)) \\ &\quad + k_{3,1} (q_1(t) dt + \sigma_1 d\mathcal{W}_1(t)), \\ dq_4(t) &= - 0.1k_{4,3} (q_4(t) dt + \sigma_4 d\mathcal{W}_4(t)) \\ &\quad + k_{4,3} (q_3(t) dt + \sigma_3 d\mathcal{W}_3(t)). \end{aligned}$$

We fix the diffusion parameters to be  $\sigma_1 = \sigma_2 = \sigma_3 = \sigma_4 = 3$ . Initial conditions are given by

$$q_1(0) = q_2(0) = q_3(0) = q_4(0) = 0.$$

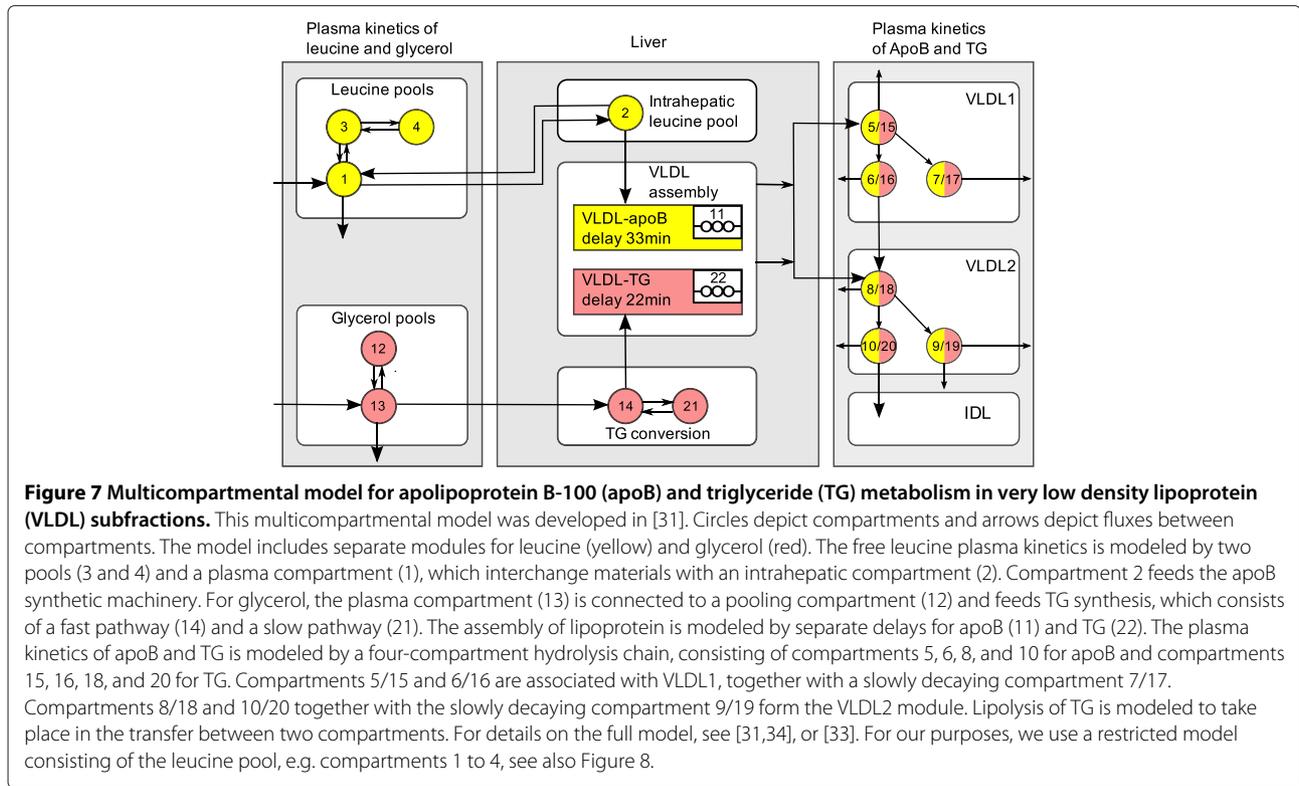
The patients get a bolus injection and therefore the input  $u_1(t)$  will be modelled as a delta distribution at time  $t = 0\text{h}$ ,

$$u_1(t) = u_{1,0}\delta(t).$$

Practically, this means that only the initial condition  $q_1$  is affected by it, and we can replace the initial condition for  $q_1$  by

$$q_1(0) = u_{1,0},$$

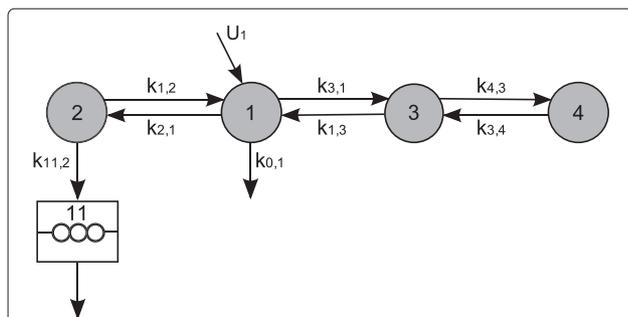
and set  $u_1(t) = 0$  in the differential equations.



The same differential equations, without noise terms, are assumed to be valid for the states  $Q_i$  and the input  $U_1$  of the tracee:

$$\frac{dQ(t)}{dt} = K(t)Q(t) + U(t)$$

where  $Q(t) = (Q_i(t))_{i=1,2,3,4}^T$ ,  $U(t) = (U_1(t), 0, 0, 0)^T$ . It is assumed that the tracee input  $U_1(t) = U_1$  is constant



**Figure 8 Schematic depiction of the restricted model (leucine pool) [33].** This scheme is a subscheme of Figure 7. Circles depict compartments. Arrows depict fluxes between compartments and are labelled with the corresponding fractional transfer coefficients. Compartment 1 is the plasma-leucine compartment where the leucine is injected. Compartment 2 is an intrahepatic compartment which is the source for apoB synthesis. Compartments 3 and 4 are body protein pools. The output is at compartment 1. Compartment 11 is a delay compartment used only as output from compartment 2.

but unknown. We will therefore estimate it together with the transfer parameters. Since for the tracee steady state is assumed (i.e.  $dQ_i(t)/dt = 0$ ), it is possible to solve those equations for  $Q_1(t)$ , and we get:

$$Q_1(t) = \frac{(k_{11,2} + k_{1,2})U_1}{k_{0,1}(k_{11,2} + k_{1,2}) + k_{11,2}k_{1,2}}$$

The output is a measurement proportional to the ratio between the tracer and the tracee, disturbed by log-normal noise:

$$y_1(t) = p_1 \frac{q_1(t)}{Q_1(t)} \xi_t, \quad \xi_t \sim \text{Log-}\mathcal{N}(0, \sigma_{y_1}^2) \text{ independently for each } t,$$

where we assume the value of the variance parameter to be  $\sigma_{y_1}^2 = 0.5^2$  (this is the variance of  $\log \xi_t$ ). The parameter  $p_1$  denotes the unknown proportion of plasma leucine that actually is in the plasma. The parameters  $p_1$  and  $U_1$  are not jointly identifiable, therefore we fix  $p_1 = 0.65$ . More details concerning the deterministic model (without noise terms) can be found in [31] and [33]. Note that the stochastic disturbances are not part of the original model, they are rather augmentations of the model used in this article.

#### The mixed effects model

The model as presented in the last paragraph contains only flux parameters  $k_{j,i}$  which are assumed to be the

same for every individual. Neither does it account for individual differences between several persons, nor does it account for possible changes of the flux parameters when the persons under consideration are affected by a disease or a treatment. To be able to treat these differences in an appropriate way, we introduce group and patient specific parameters in the model; namely the transfer coefficient  $k_{0,1}$  will be split into a group dependent and a patient dependent part. In this way, we introduce so-called mixed effects into the model. Mixed effects generally increase the difficulty in inference making. In the following test runs, we will use measurement data previously reported as individual data for a total of 34 subjects [31,32]. Among them, 15 patients belonged to the diabetes group, and 19 to the control group. From experiments, it can be observed that the degradation rate  $k_{0,1}$  of plasma leucine is significantly different for people with and without diabetes. We therefore assume that the expected value of  $k_{0,1}$  in each group differs and may be either  $k_{0,1}^d$  or  $k_{0,1}^c$  corresponding to the diabetes group and the control group, respectively. Additionally, we assume that we have patient-dependent random factors  $\zeta_p$  modelling the parametric uncertainties among individuals, such that finally

$$k_{0,1}^{(p)} = \begin{cases} \zeta_p k_{0,1}^d & \text{if the patient } p \text{ is in the diabetes group,} \\ \zeta_p k_{0,1}^c & \text{if the patient } p \text{ is in the control group} \end{cases}$$

where all  $\zeta_p$ 's are static and independently log-normally distributed:

$$\zeta_p = \exp(\eta_p) \quad \text{with } \eta_p \sim \mathcal{N}(0, \sigma_{\eta_p}^2) \\ \text{independently for each } p$$

for  $p = 1, \dots, 34$ . As a consequence, each of the states  $q_1, \dots, q_4$  has to be considered separately for each patient  $p$ . We indicate this by writing  $q_1^{(p)}, \dots, q_4^{(p)}$ ,  $p = 1, \dots, 34$ .

The aim of the estimation runs is, apart from estimating the remaining parameters, to show that the group dependent parameters  $k_{0,1}^d, k_{0,1}^c$  are indeed different. We want to apply Bayesian estimation to the parameters. For this reason, we principally have to treat them like state variables in the particle filter. Since the estimation of static variables is problematic with particle filter methods, it is standard to introduce small artificial stochastic dynamics to the parameters consisting of normal increments with decaying variances [35]. The same is true for the static noise parameters  $\eta_p$  which also are to be estimated. Our process  $X_t$  is then given as an augmented state vector

$$X_t = \left( q_{1:4}^{(1:34)}(t), k_{0,1}^c(t), k_{0,1}^d(t), k_{1,2}(t), k_{1,3}(t), k_{3,1}(t), \right. \\ \left. k_{4,3}(t), U_1(t), \eta_{1:34}(t) \right)^T.$$

The overall model is thus a non-linear mixed effects model with three levels of effects (parameters), namely global parameters, group dependent parameters ( $k_{0,1}^d, k_{0,1}^c$ ), and personal parameters ( $\zeta_p$ ). Nevertheless, since the core of

the model is linear, i.e. the states  $q_1, \dots, q_4$  conditioned on all parameters, a Rao-Blackwellization concerning the linear parts of the model is possible, and the Kalman filter applied to this linear partial model can be used in combination with particle filtering for the non-linear parts [36]. Since the model is used for demonstration purposes only, we did not use this technique although in principle it is possible.

#### Estimation runs

Estimation and simulation runs have been performed with data from all 34 patients (19 from control group and 15 from diabetes group). The computer experiments have been done as follows. We first estimate parameters with the MTU particle filter. Separately, we estimate parameters with the standard particle filter under the same conditions and with the same seed for the random number generator. The initial distribution of the particles is then the same in both cases. For both runs, we compute an estimate of the effective sample size (ESS) and the data likelihood over time. Both estimates allow a performance comparison of the MTU versus the standard particle filter. Secondly, the empirical medians of the final parameter distributions are afterwards used to perform simulation runs in both cases. Both versions of the particle filter, this time with parameters fixed to the estimated values, are used to perform these simulations. In these simulation runs, the data are used for the computation of the data likelihood conditioned on the estimated parameters. The resulting distributions of the simulated measurements can be compared to the true measurements, both visually and by observing the data likelihood.

We used 10,000 particles and a resampling threshold of 7,500. Stepsizes in the MTU filter are between  $10^{-7}$ h and  $10^{-3}$ h, adaptively computed based on the ESS estimate. In the standard filter, we use a fixed stepsize of  $10^{-3}$ h. The data contain measurements until time  $t = 8$ h, but we perform estimations and simulations only until time  $t = 1$ h, mainly to reduce computation time; anyhow, after  $t = 1$ h, the tracer concentrations are relatively low and do not change considerably, and therefore the measurements are not expected to improve the estimations significantly. In our implementation, we directly sample from the states  $X_t$  (i.e.  $\tilde{X}_{[t_0, \infty)} = X_{[t_0, \infty)}$  in law) using the Euler-Maruyama scheme for discretization [37].

As mentioned earlier, Bayesian parameter estimation with particle filters requires the introduction of artificial dynamic noise for the parameters. It is standard to use normal increments with decaying variances [35]. Since in our case all parameters with exception of the  $\eta_p$ 's are assumed to be positive, the application of a "log-normal" noise (based on a geometric Brownian motion) in place of the standard normal noise seems to be more appropriate for these parameters. In detail, it has been done as

follows. The priors and the distributions of the artificial noise are chosen to be log-normal for all parameters with exception of the individual parameters  $\eta_p$  which have normal priors and noise. The prior for the parameters  $k_{0,1}^c, k_{0,1}^d, k_{1,2}, k_{1,3}, k_{3,1}, k_{4,3}$  is  $\text{Log-}\mathcal{N}(0, 1^2)$ , the prior for  $U_1$  is  $\text{Log-}\mathcal{N}(\log(100), 1^2)$ . The prior for each  $\eta_p$  is  $\mathcal{N}(0, 0.5^2)$ . The parameter update (“artificial noise”) is generally performed according to

$$d\theta(t) = \theta(t)\sigma_\theta(t) dW_t^\theta$$

for  $\theta = k_{0,1}^c, k_{0,1}^d, k_{1,2}, k_{1,3}, k_{3,1}, k_{4,3}, U_1$ , and according to

$$d\theta(t) = \sigma_\theta(t) dW_t^\theta$$

for  $\theta = \eta_p$ . It is standard to decrease the variance of the artificial noise over time. In our case we have chosen the diffusion parameter  $\sigma_\theta$  of each artificial noise variable to be dependent on time via a quadratic function

$$\sigma_\theta(t) = a_\theta / (t - b_\theta)^2$$

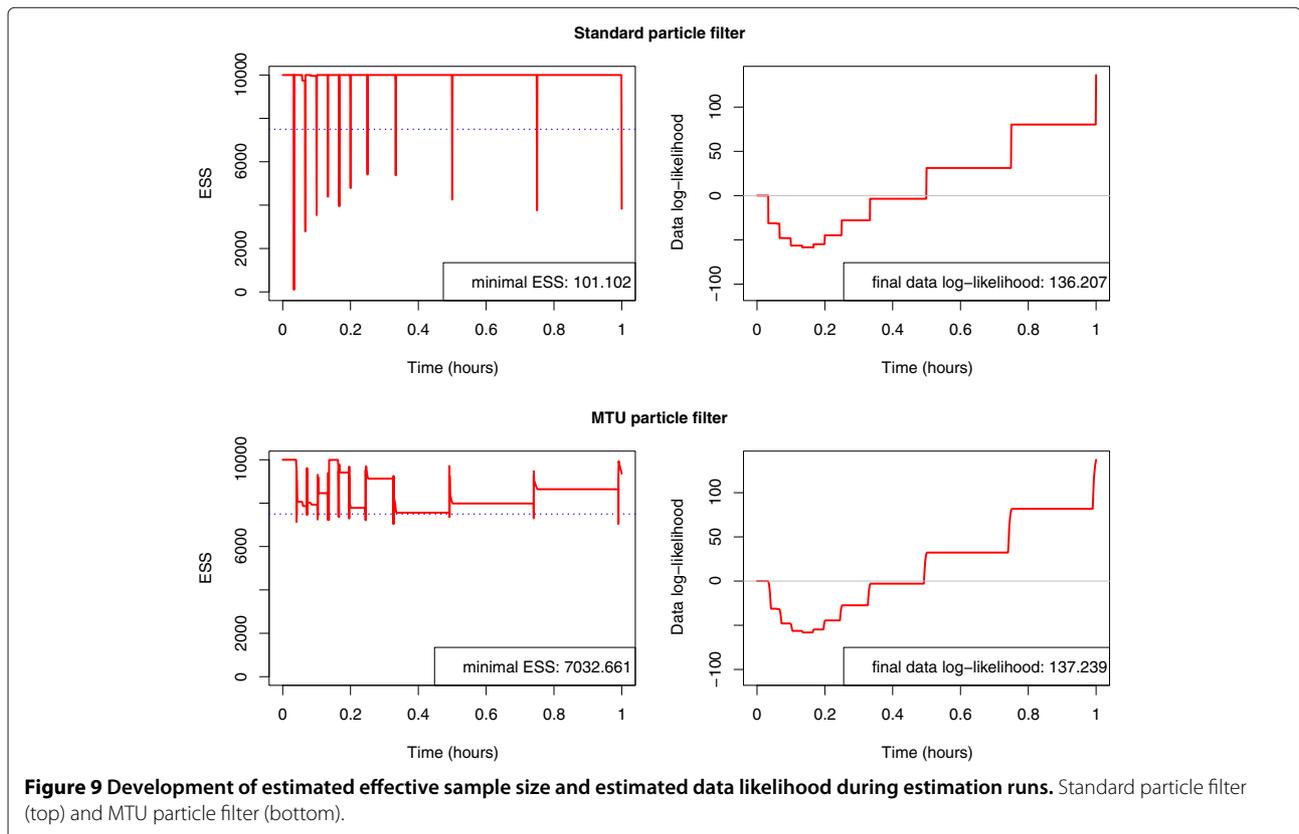
with parameter dependent coefficients  $a_\theta$  and  $b_\theta$ . Practically,  $a_\theta$  and  $b_\theta$  have been determined by fixing two interpolation points  $(t_0, \sigma_\theta(t_0))$  and  $(t_1, \sigma_\theta(t_1))$ . It holds:

$$b_\theta = (t_1 - t_0) / \left(1 - \sqrt{\sigma_\theta(t_0) / \sigma_\theta(t_1)}\right) \quad \text{and}$$

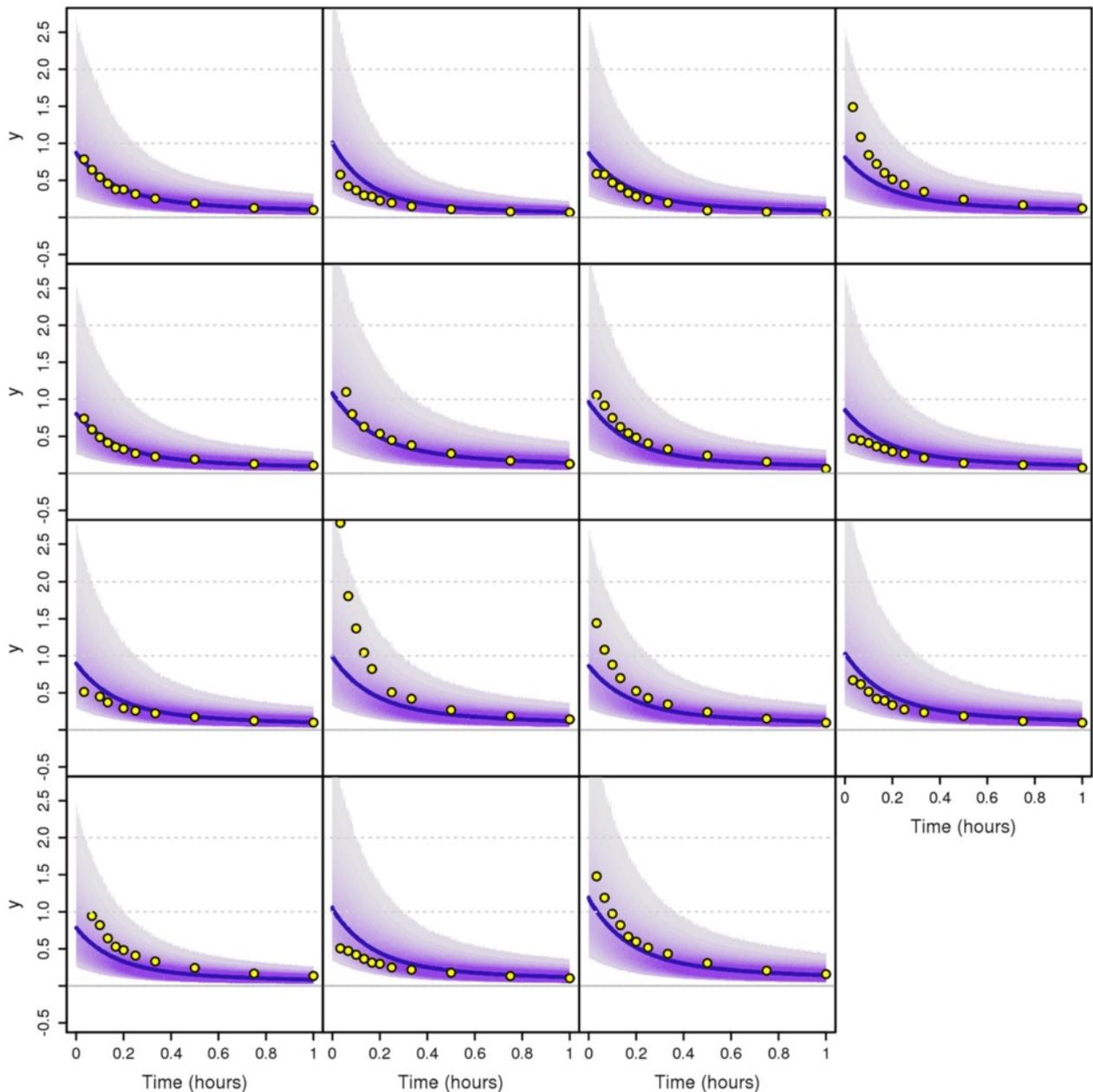
$$a_\theta = \sigma_\theta(t_0)(t_0 - b_\theta)^2.$$

We found best performance with the following choices: For all parameters, we have chosen  $t_0 = 0\text{h}, t_1 = 2\text{h}$  and  $\sigma_\theta(t_1) = \sigma_\theta(t_0)/10$  which means that the diffusion parameter has dropped to 10% of its initial value at time  $t = 2\text{h}$ . The initial values are  $\sigma_\theta(t_0) = 0.5\text{h}$  for all parameters with exception of the  $\eta_p$ 's which have higher initial diffusion  $\sigma_\theta(t_0) = 1\text{h}$ .

As mentioned earlier, we have performed two different estimation and simulation runs, one with the MTU particle filter with distributed measurement times, and for comparison one run with the standard particle filter. In the MTU particle filter, the distributions of the measurement times are truncated normal distributions with mean equal to the nominal value of the measurement time and with variance  $0.001^2$ . The normal distribution is truncated at the time point 0.01h left and right from the mean value. In Figure 9, the development of the estimated effective sample size and the estimated data likelihood is shown, both with respect to time  $t$ . The development of predictions of the measurements during a simulation run with the final estimated parameters is shown in Figures 10 and 11 for the run with the standard particle filter, and in Figures 12 and 13 for the run with the MTU particle filter. Finally, in Figures 14 and 15, box plots of the final estimated global and individual parameters are shown, respectively.



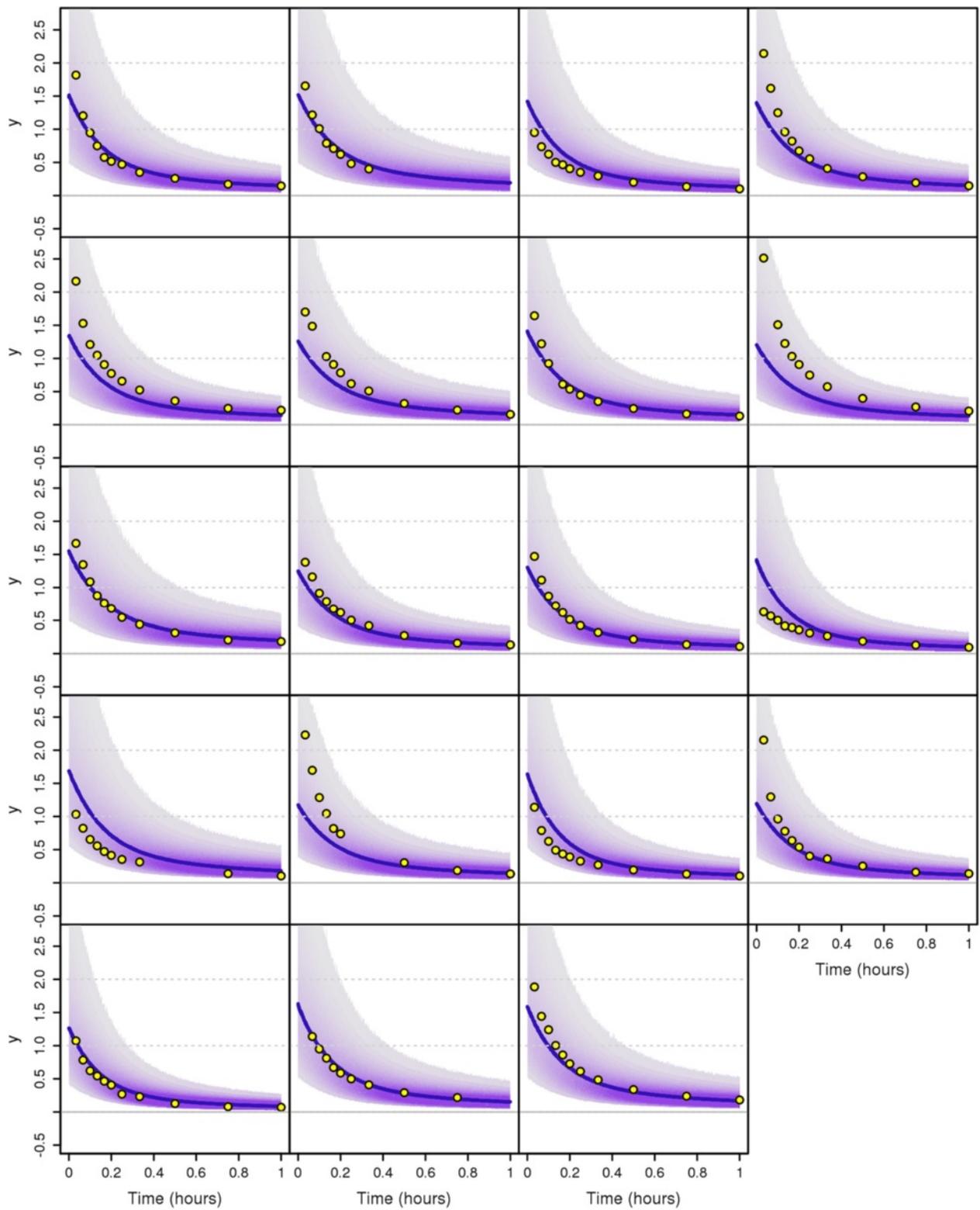
**Figure 9** Development of estimated effective sample size and estimated data likelihood during estimation runs. Standard particle filter (top) and MTU particle filter (bottom).



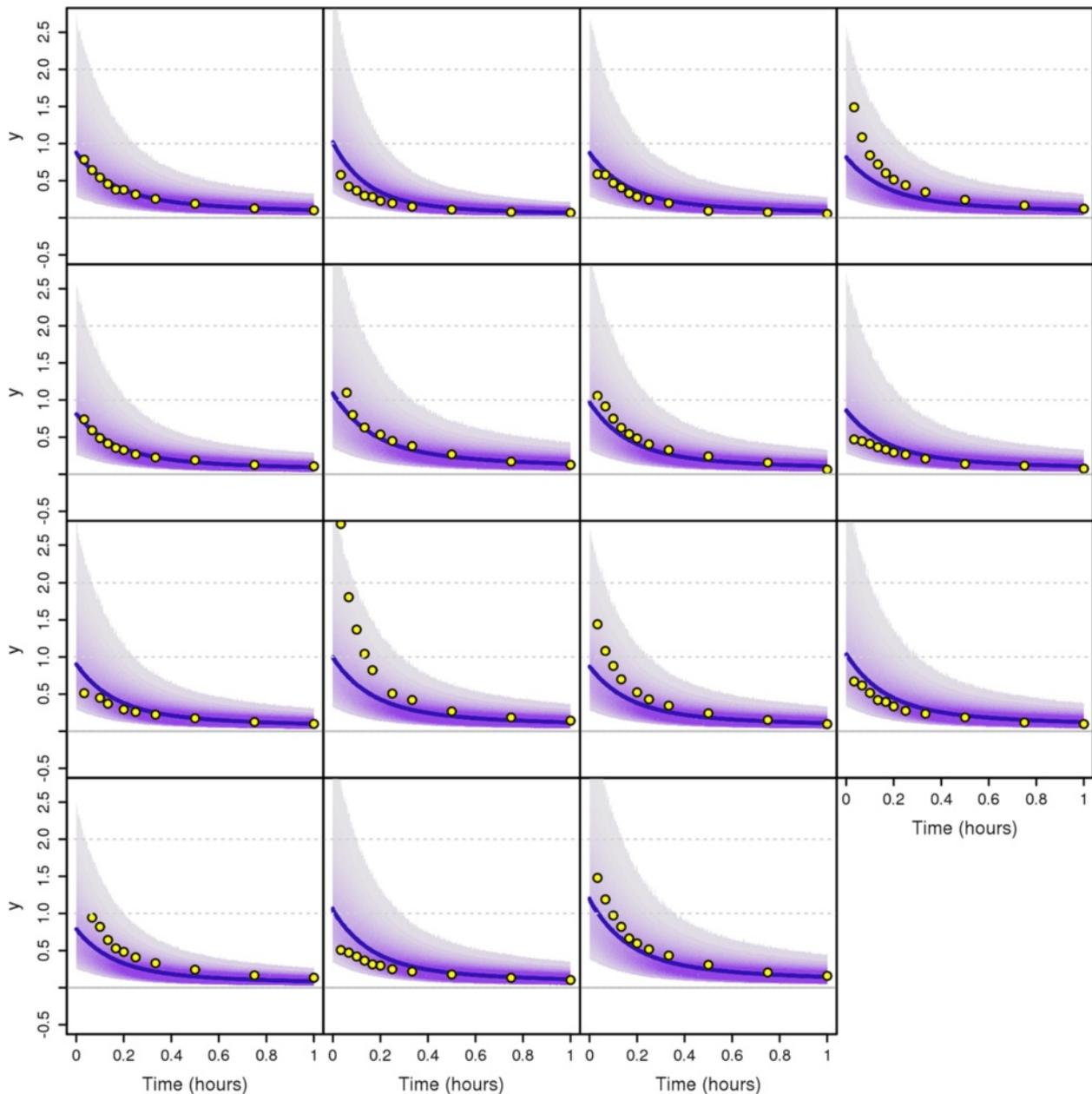
**Figure 10** Predicted measurement distribution over time (standard particle filter) during simulation run. Patients from diabetes group. Development of predicted measurement distributions over time during simulation run with parameters estimated by the standard particle filter. Circles: Measurements. Solid line: Median of simulated measurements. Violet shaded area: Measurement distributions based on empirical quantiles.

A comparison of the results of MTU-PF and standard particle filter shows that both algorithms exhibit very similar performance with respect to the quality of the estimated parameters, since in both cases the development of the data likelihood is very similar both for estimation and simulation. The estimated log-likelihood of the data at the final time is 137.239 for the MTU particle filter and 136.207 in the standard case, which is practically equal. The computation time of the MTU-PF is

only slightly higher than the one of the standard filter. Visual inspection of the simulation runs shows that the simulated measurements of the model with parameters estimated by both filters fit the data in a similar way. This impression is supported by the values of the estimated data likelihood. The MTU particle filter has a final log-likelihood value of 157.622, similar to the one of the standard filter with a value of 155.952. The difference is insignificant.



**Figure 11 Predicted measurement distribution over time (standard particle filter) during simulation run. Patients from control group.**  
Development of predicted measurement distributions over time during simulation run with parameters estimated by the standard particle filter. Circles: Measurements. Solid line: Median of simulated measurements. Violet shaded area: Measurement distributions based on empirical quantiles.

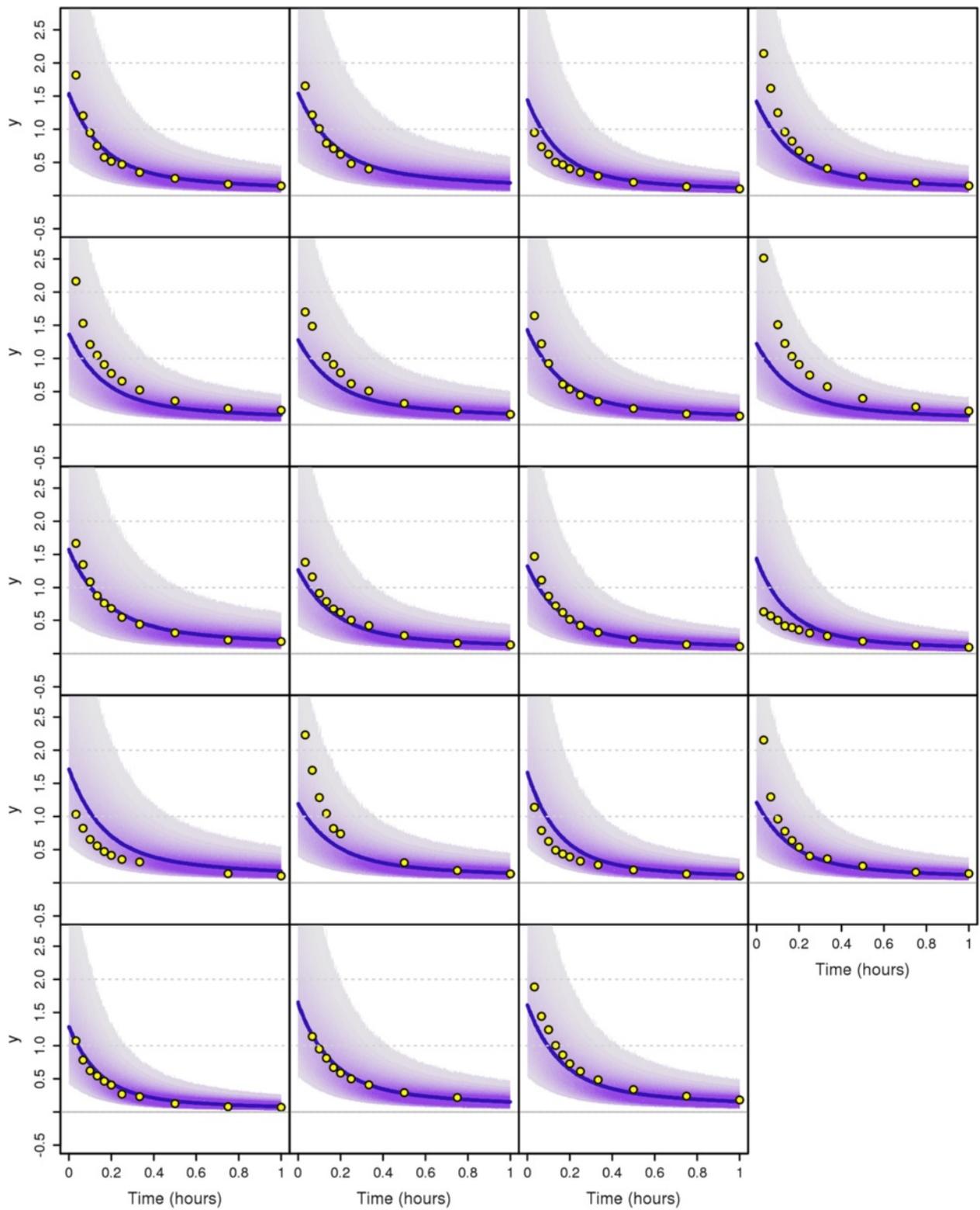


**Figure 12 Predicted measurement distribution over time (MTU particle filter) during simulation run. Patients from diabetes group.**

Development of predicted measurement distributions over time during simulation run with parameters estimated by the MTU particle filter. Circles: Measurements. Solid line: Median of simulated measurements. Violet shaded area: Measurement distributions based on empirical quantiles.

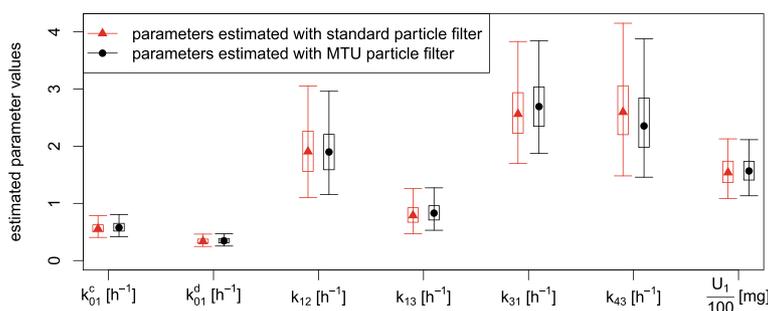
In contrast to the insignificant differences concerning the resulting likelihoods between the MTU-PF and the standard PF, the development of the ESS estimate in the estimation runs differs remarkably. With the MTU particle filter, the ESS estimate shows a high value at all times and does not drop below a value of 7032.661 during estimation. This is only slightly lower than the resampling threshold of 7500 (see Figure 9, upper part). The standard particle filter shows a much worse performance. As can be

observed from the lower part of Figure 9, the ESS drops several times to very low values, reaching a minimum of 101.102. The ESS measures in some sense the variance of the normalized particle weights (low ESS means high variance) and highly varying weights indicate that the particle cloud is in a bad condition, since in this case a few particles with very high weights dominate the majority of the particles with very low weights. The extreme case is  $ESS = 1$  where only one particle has a significant weight. After



**Figure 13 Predicted measurement distribution over time (MTU particle filter) during simulation run. Patients from control group.**

Development of predicted measurement distributions over time during simulation run with parameters estimated by the MTU particle filter. Circles: Measurements. Solid line: Median of simulated measurements. Violet shaded area: Measurement distributions based on empirical quantiles.



**Figure 14 Estimated global and group parameters.** Box plots of estimated distributions for the global parameters and the group dependent parameters. The medians are depicted with triangles (standard particle filter) or circles (MTU particle filter). The bottom and top of the box are the 0.25-quantile and the 0.75-quantile, i.e. 50% of the values lie within the box. The whiskers mark the 0.025-quantile and the 0.975-quantile, i.e. 95% of the values lie between the whiskers. The values for  $U_1$  have been scaled by a factor of 0.01 in order to fit into the plot.

a resampling step, only the information carried by this particle is present in the current particle cloud. Results obtained by estimation runs where such low ESS values have occurred cannot be trusted very much. If in contrast a resampling step is performed while the ESS is only slightly below the resampling threshold, which is the case in the MTU-PF algorithm, then many different particles will be chosen during resampling and a high percentage of the information contained in the particle cloud will be carried over to subsequent steps. In this sense, the MTU particle filter avoids the degeneration of the particle cloud by controlling the value of the ESS and holding it at a high value at every time. As a consequence, the estimation results should be considered to be more reliable than with the standard algorithm.

A look at the estimated values of the group parameters  $k_{0,1}^c$  and  $k_{0,1}^d$  (see Figure 14) shows that in both estimation runs (MTU-PF and standard PF), the rate  $k_{0,1}^d$  for diabetes patients is only about 60 percent of the rate  $k_{0,1}^c$  of the control group (standard particle filter:  $0.337h^{-1}$  vs  $0.557h^{-1}$ , MTU particle filter:  $0.346h^{-1}$  vs  $0.577h^{-1}$ ). The good performance especially of the MTU particle filter strengthens the confidence in the obtained result and leads to the conclusion that the secretion rate  $k_{0,1}$  is indeed lower for the

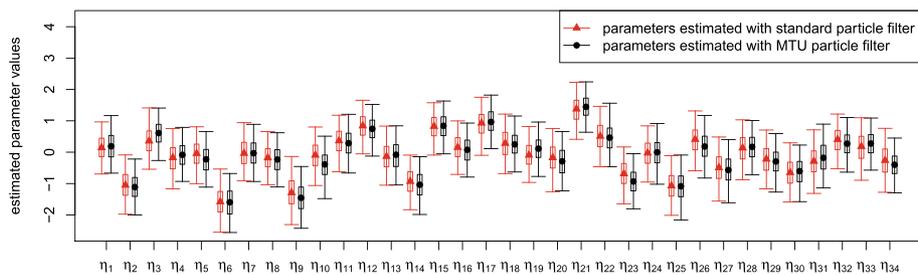
diabetes patients than for the people from the control group.

### Conclusions and future work

We proposed a new modification of the particle filter algorithm which works in continuous-time settings. It allows the direct inclusion of measurement time uncertainties in the underlying model. The modifications additionally allow the use of time-stepping strategies to improve the performance of the algorithm. The assumption of a random distribution of measurement times is natural in many applications.

The MTU-PF method is generally applicable. Even when measurement times may be assumed to be concentrated on single time points, our method can be used as a kind of regularization of the standard particle filter method if artificial distributions with highly concentrated masses around the measurement points are introduced.

We compared the performance of the MTU-PF to the standard PF and to an alternative Maximum Likelihood estimation method on a small artificial example. The results clearly show the advantage of the application of the MTU-PF in cases of uncertain measurement times.



**Figure 15 Estimated individual parameters.** Box plots of estimated distributions for the individual parameters. The medians are depicted with triangles (standard particle filter) or circles (MTU particle filter). The bottom and top of the box are the 0.25-quantile and the 0.75-quantile, i.e. 50% of the values lie within the box. The whiskers mark the 0.025-quantile and the 0.975-quantile, i.e. 95% of the values lie between the whiskers.

We believe that our MTU particle filter is especially suitable for biological/medical applications where — compared to technical applications — the variance of the measurement values is relatively high due to biological variation and because relatively few consecutive measurements are possible. We provided an illustrative application from a PK/PD study. A comparison of our MTU particle filter with the standard filter showed that in this case our method is able to avoid weight degeneracy measured in terms of the Effective Sample Size (ESS) estimate. Even though the estimations in this application with both standard and MTU particle filters are reasonably good, the fit to the measurements is still not perfect. Whether that is due to our choices of model and parameters, or due to the known weaknesses of the general SMC method (which also our modification necessarily suffers from), remains to be evaluated in greater detail. In subsequent studies, we plan to apply the new algorithm to the complete liver-plasma model using additional measurements to be able to draw conclusions of greater medical relevance.

Another topic may also prove interesting for future work. Our experiments showed that the results are highly dependent on the choice of the development of the diffusion coefficients of the artificial noise necessary for Bayesian parameter estimation. While there is a general agreement that these coefficients should decrease over time, there is currently a lack of automated methods for making appropriate choices of the diffusion coefficients, both for initial values and dynamic development. However, to provide a really practical method for parameter estimation in non-linear mixed effects models (or even in models which exhibit only global parameters), our approach could also be combined with methods better suited to the estimation of fixed parameters, a good candidate being the PMCMC methods proposed in [11]. This is future work.

In summary, we believe that the method presented in this article opens the door to even more efficient and reliable state sampling and parameter estimation methods based on the particle filter algorithm operating on continuous-time stochastic state space systems.

**Notation**

|   |  |
|---|--|
| $(\Omega, \mathcal{A}, P)$  | probability space  |
| $(\mathcal{X}_t, \mathcal{B}_{\mathcal{X}_t})$                    | arbitrary measurable space (for each $t \in [t_0, \infty)$ with $t_0 \in \mathbb{R}$ ) |
| $X_t : \Omega \rightarrow \mathcal{X}_t$                          | $\mathcal{A}$ - $\mathcal{B}_{\mathcal{X}_t}$ measurable random variable               |
| $X_{[t_0, \infty)} := (X_t)_{t \in [t_0, \infty)}$                | continuous-time Markov process with general state space $\mathcal{X}_{[t_0, \infty)}$  |
| $\mathcal{X}_{[t_0, \infty)} := \prod_{t_0 \leq s} \mathcal{X}_s$ | state space of $X_{[t_0, \infty)}$   |

|   |  |
|---|--|
| $\mathcal{L}_{X_t}$   | the pushforward measure of $P$ under $X_t$ , i.e. $\mathcal{L}_{X_t}(B) := P(X_t^{-1}(B))$ for all $B \in \mathcal{B}_{\mathcal{X}_t}$ |
| $\mathcal{L}_{X_{[t_0, \infty)}}$   | the pushforward measure of $P$ under $X_{[t_0, \infty)} := (X_s)_{s \in [t_0, \infty)}$ (with the corresponding product algebra)       |
| $\mathcal{X}_{[t_0, t]} := \prod_{t_0 \leq s \leq t} \mathcal{X}_s$   | the state space restricted to the interval $[t_0, t]$  |
| $\mathcal{L}_{X_{[t_0, t]}}$  | the corresponding pushforward measure  |
| $N$   | number of particles  |
| $(x_t^i)_{i=1, \dots, N}$   | state samples at time $t$  |
| $K_{s,t}(x_s, dx_t)$  | the Markov kernel of the process $X_{[t_0, \infty)}$ from time $s$ to time $t$   |
| $a(x, t)$   | drift vector   |
| $B(x, t)$   | diffusion matrix   |
| $\mathcal{W}_t$   | multidimensional standard Wiener process   |
| $(\mathcal{Y}_j, \mathcal{B}_{\mathcal{Y}_j})$  | measurable space   |
| $Y_{1:M}$   | observation random variable with values in measurable spaces $(\mathcal{Y}_j, \mathcal{B}_{\mathcal{Y}_j})$                            |
| $g_j(y_j   x_{t_j}, t_j)$   | conditional probability density with respect to a reference measure $\mu_{\mathcal{Y}_j}$  |
| $\mu_{\mathcal{Y}_j}$   | reference measure $\mu_{\mathcal{Y}_j}$ on $(\mathcal{Y}_j, \mathcal{B}_{\mathcal{Y}_j})$  |
| $t_j (j = 1, \dots, M)$   | observation times  |
| $T_j$   | random variables modelling the uncertainty about exact observation times   |
| $\lambda_{[t_0, \infty)}$   | Lebesgue measure on the interval $[t_0, \infty)$   |
| $\gamma_j(t_j)$   | probability density of $T_j$ with respect to $\lambda_{[t_0, \infty)}$   |
| $\tilde{X}_{t_0, M}, \mathcal{L}_{\tilde{X}_{t_j}}$ , and $\tilde{K}_{t_{j-1}, t_j}(x_{t_{j-1}}, dx_{t_j})$ | Markov chain, pushforward measure, and kernel for importance sampling  |
| $Q_{t_j   t_{j-1}}(x_{t_j}   x_{t_{j-1}})$  | Radon-Nikodym derivative $\frac{K_{t_{j-1}, t_j}(x_{t_{j-1}}, dx_{t_j})}{\tilde{K}_{t_{j-1}, t_j}(x_{t_{j-1}}, dx_{t_j})}$             |
| $Q_{t_0}(x_{t_0}) := \frac{d\mathcal{L}_{X_{t_0}}(x_{t_0})}{d\mathcal{L}_{\tilde{X}_{t_0}}(x_{t_0})}$       | Radon-Nikodym derivative at start time $t_0$   |
| $w_t^i$   | unnormalized weight of particle $i$ at time $t$  |
| $\tilde{w}_t^i := \frac{w_t^i}{\sum_{v=1}^N w_t^v}$   | normalized weight of particle $i$ at time $t$  |
| $s_\ell$  | $\ell$ -th resampling time   |

|   |   |
|---|---|
| $v_{s\ell}^i$   | (unnormalized) selection weight   |
| $p_\ell^i$  | normalized selection weight (probability that particle $i$ will be selected during resampling)      |
| $\nu_\ell : I \rightarrow I$                                    | selection function on the index set $I := \{1, \dots, N\}$ for the $\ell$ -th resampling step       |
| $Z_{t_k}(t_{1:M})$  | data likelihood   |
| $f_t$   | full density at time $t$  |
| $\hat{f}_t$   | filter density at time $t$ , i.e. only those observations $y_j$ are included for which $t_j \leq t$ |
| $\bar{g}_{j,t}(y_j   x_{t_j}, t_j)$                             | conditional probability density $g_j(y_j   x_{t_j}, t_j)$ if $t_j \leq t$ , 1 otherwise             |
| $W_{j,t} : \Omega \rightarrow \mathbb{R}_{\geq 0}$              | stochastic process given by $dW_{j,t}(\omega) = (g_j(y_j   X_t(\omega), t) - 1) \gamma_j(t) dt$     |
| $W_t : \Omega \rightarrow \mathbb{R}_{\geq 0}$                  | product process of the $W_{j,t}$ , where $j = 1, \dots, M$  |
| $\hat{E}_t[h(X_t)   Y_{1:M} = y_{1:M}]$                         | expectation of $h(X_t)$ given $Y_{1:M} = y_{1:M}$ with respect to the filtered state $X_t$          |
| $w_{j,t} : X_{[t_0,t]}(\Omega) \rightarrow \mathbb{R}_{\geq 0}$ | partial weight at time $t$ (w.r.t. the $j$ -th observation)   |
| $w_t : X_{[t_0,t]}(\Omega) \rightarrow \mathbb{R}_{\geq 0}$     | weight at time $t$  |
| $\bar{\gamma}_{j,t}$  | cumulative distribution function $\int_{t_0}^t \gamma_j(t_j) dt_j$                                  |
| $\bar{w}_{j,t}(x_{[t_0,t]})$                                    | $\int_{t_0}^t g_j(y_j   x_{t_j}, t_j) \gamma_j(t_j) dt_j$   |
| $t_0 = \tau_0 < \tau_1 < \dots < \tau_D$                        | time discretization   |
| $\bar{v}^i$   | cumulative product of the selection weights for particle $i$  |
| $\Delta\tau_d = \tau_d - \tau_{d-1}$                            | stepsize  |
| $\bar{Z}_{\tau_d}$  | correction factor for the data likelihood at time $\tau_d$  |
| $Q_i$   | mass of the tracee in compartment $i$   |
| $q_i$   | mass of the tracer in compartment $i$   |
| $U_i$   | input for the tracee in compartment $i$ (e.g. $U_1$ denotes the influx into compartment 1)          |
| $u_i$   | input for the tracer in compartment $i$   |
| $k_{j,i}$   | transfer coefficient of the tracers from compartment $i$ to compartment $j$                         |
| $\sigma_i$  | diffusion parameter   |
| $\xi_t$   | log-normal noise ( $\xi_t \sim \text{Log-}\mathcal{N}(0, \sigma_{\gamma_1}^2)$ )                    |

|                        |   |
|------------------------|---|
| $k_{0,1}^d, k_{0,1}^c$ | degradation rate of plasma leucine for people in the diabetes group and the control group, respectively   |
| $\zeta_p, \eta_p$      | patient-dependent random factors modelling the parametric uncertainties among individuals ( $\zeta_p = \exp(\eta_p)$ with $\eta_p \sim \mathcal{N}(0, \sigma_{\eta_p}^2)$ ) |

## Additional file

**Additional file 1: Data used for estimation and simulation.** The data consist of one record for each patient, each record consisting of 4 lines in the following format:

```
patient id [COMMA] group (control or diabetes) [COMMA] leucine initial
value [NEW LINE]
measurement 1 time [COMMA] measurement 2 time [COMMA]
... [COMMA] measurement  $n_p$  time [NEW LINE]
measurement 1 value [COMMA] measurement 2 value [COMMA]
... [COMMA] measurement  $n_p$  value [NEW LINE]
[NEW LINE]
```

where  $n_p$  is the number of measurements for patient  $p$ , time is given in hours, and each measurement value is the measured ratio of isotope labeled leucine.

## Competing interests

The authors declare that they have no competing interests.

## Authors' contributions

JH and AK contributed to the underlying ideas of the method and its elaboration and implemented the algorithm. Both authors contributed equally to the initial manuscript. MJ and MA commented and refined the manuscript. MA provided the compartmental model and the data. MT acquired the data. MJ contributed to the discussion and supervised the project. All authors read and approved the final manuscript.

## Acknowledgements

Thanks go to Michaela Höhne from the University of Kaiserslautern for her help in preparing the figures. Data was generated by Martin Adiels, Professor Marja-Riitta Taskinen, Helsinki University, and Professor Jan Borén, Göteborg University, and has been published elsewhere ([31,32]). We would like to thank the anonymous reviewers for their helpful suggestions and feedback. This work was partially supported by the Fraunhofer Internal Programs under Grant No. MAVO 819 620. The work was also supported by grants from the European Commission 7th Framework Programme (UNICELLSYS, grant No 201142), the Swedish Foundation for Strategic Research through the Gothenburg Mathematical Modelling Centre (GMMC), the Sahlgrenska Center for Cardiovascular and Metabolic Research (CMR), the Sigrid Juselius Foundation, and the Helsinki University Central Hospital Research Foundation, Helsinki, Finland.

## Author details

<sup>1</sup>Fraunhofer-Institut für Techno- und Wirtschaftsmathematik (ITWM; Fraunhofer Institute for Industrial Mathematics), Kaiserslautern, Germany. <sup>2</sup>Department of Medicine, University of Helsinki, Helsinki, Finland. <sup>3</sup>Department of Mathematical Sciences, University of Gothenburg, Gothenburg, Sweden. <sup>4</sup>Wallenberg laboratory, Sahlgrenska Center for Cardiovascular Research, Department of Medicine, University of Gothenburg, Gothenburg, Sweden. <sup>5</sup>Fraunhofer Chalmers Centre (FCC), Gothenburg, Sweden.

Received: 31 October 2011 Accepted: 7 January 2013  
 Published: 19 January 2013

## References

1. Overgaard RV, Jonsson N, Tornøe CW, Madsen H: **Non-linear mixed-effects models with stochastic differential equations: implementation of an estimation algorithm.** *Journal of Pharmacokinetics and Pharmacodynamics* 2005, **32**:85–107.
2. Moles CG, Mendes P, Banga JR: **Parameter estimation in biochemical pathways: a comparison of global optimization methods.** *Genome research* 2003, **13**(11):2467–2474.
3. Chou ICC, Voit EO: **Recent developments in parameter estimation and structure identification of biochemical and genomic systems.** *Mathematical biosciences* 2009, **219**(2):57–83.
4. Ashyraliyev M, Fomekong-Nanfack Y, Kaandorp JA, Blom JG: **Systems biology: parameter estimation for biochemical models.** *FEBS Journal* 2009, **276**(4):886–902.
5. Bohlin T: *Practical Grey-Box Process Identification: Theory and Applications.* Advances in Industrial Control, London: Springer; 2010.
6. Nielsen J, Madsen H, Young P: **Parameter Estimation in Stochastic Differential Equations: An Overview.** *Annual Reviews in Control* 2000, **24**:83–94.
7. Crisan D, Doucet A: **A survey of convergence results on particle filtering methods for practitioners.** *IEEE Transactions on Signal Processing* 2002, **50**(3):736–746.
8. Andrieu C, De Freitas N, Doucet A: **Sequential MCMC for Bayesian model selection.** In *Proceedings of the IEEE Signal Processing Workshop on Higher-Order Statistics, 1999.* Caesarea, Israel: IEEE; 1999:130–134.
9. Fearnhead P: **MCMC, sufficient statistics and particle filters.** *Journal Of Computational And Graphical Statistics* 2002, **11**(4):848–862.
10. Storvik G: **Particle filters for state-space models with the presence of unknown static parameters.** *IEEE Transactions on Signal Processing* 2002, **50**(2):281–289.
11. Andrieu C, Doucet A, Holenstein R: **Particle Markov chain Monte Carlo methods.** *Journal of the Royal Statistical Society - Series B: Statistical Methodology* 2010, **72**(3):269–342.
12. Gordon N, Salmond D, Smith A: **Novel approach to nonlinear/non-Gaussian Bayesian state estimation.** *IEE-Proceedings-F* 1993, **140**(2):107–113.
13. Cappé O, Godsill S, Moulines E: **An overview of existing methods and recent advances in Sequential Monte Carlo.** *Proceedings of the IEEE* 2007, **95**(5):899–924.
14. Doucet A, de Freitas N, Gordon N (Eds): *Sequential Monte Carlo Methods in Practice.* Statistics for Engineering and Information Science, New York: Springer Verlag; 2001.
15. Beal S, Sheiner L: *NONMEM User's Guides.* NONMEM Project Group, University of California, San Francisco; 1994.
16. Tornøe CW, Overgaard RV, Agersø H, Nielsen HA, Madsen H, Jonsson EN: **Stochastic differential equations in NONMEM: implementation, application, and comparison with ordinary differential equations.** *Pharmaceutical Research* 2005, **22**(8):1247–1258.
17. Berglund M, Sunnåker M, Adiels M, Jirstrand M, Wennberg B: **Investigations of a compartmental model for leucine kinetics using non-linear mixed effects models with ordinary and stochastic differential equations.** *Mathematical Medicine and Biology: A Journal of the IMA* 2011, **29**(4):361–384.
18. Donnet S, Samson A: **EM algorithm coupled with particle filter for maximum likelihood parameter estimation of stochastic differential mixed-effects models** [http://hal.archives-ouvertes.fr/hal-00519576]. [Preprint version 2 – 21 Jul 2011]
19. Donnet S, Samson A: **Parametric inference for mixed models defined by stochastic differential equations.** *ESAIM Probability and Statistics* 2008, **12**:196–218.
20. Racine-Poon A, Wakefield J: **Statistical methods for population pharmacokinetic modelling.** *Statistical Methods in Medical Research* 1998, **7**:63–84.
21. Sheiner L, Wakefield J: **Population modelling in drug development.** *Statistical Methods in Medical Research* 1999, **8**(3):183.
22. Andersen KE, Hojbjerg M: **A population-based Bayesian approach to the minimal model of glucose and insulin homeostasis.** *Statistics in Medicine* 2005, **24**(15):2381–2400.
23. Hu XL, Schön T, Ljung L: **A basic convergence result for particle filtering.** *IEEE Transactions on Signal Processing* 2008, **56**(4):1337–1348.
24. Douc R, Cappé O, Moulines E: **Comparison of Resampling Schemes for Particle Filtering.** In *Proceedings of the 4th International Symposium on Image and Signal Processing and Analysis, 2005.* ISPA, Zagreb, Croatia: IEEE; 2005:64–69.
25. Hol JD, Schön TB, Gustafsson F: **On resampling algorithms for particle filters.** In *Proceedings of Nonlinear Statistical Signal Processing Workshop (NSSPW).* Cambridge, UK: IEEE; 2006:79–82.
26. Pitt MK, Shephard N: **Filtering via simulation: auxiliary particle filter.** *Journal of the American Statistical Association* 1999, **94**:590–599.
27. Del Moral P, Doucet A, Jasra A: **Sequential Monte Carlo samplers.** *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 2006, **68**(3):411–436.
28. R Development Core Team: *R: A Language and Environment for Statistical Computing.* R Foundation for Statistical Computing, Vienna, Austria 2011. [http://www.R-project.org]. [ISBN 3-900051-07-0].
29. Demant T, Packard CJ, Demmelmair H, Stewart P, Bedynek A, Bedford D, Seidel D, Shepherd J: **Sensitive methods to study human apolipoprotein B metabolism using stable isotope-labeled amino acids.** *Am J Physiol* 1996, **270**(6 Pt 1):E1022–36.
30. Cobelli C, Saccomani MP, Tessari P, Biolo G, Luzi L, Matthews DE: **Compartmental model of leucine kinetics in humans.** *Am J Physiol* 1991, **261**(4 Pt 1):E539–50.
31. Adiels M, Packard C, Caslake MJ, Stewart P, Soro A, Westerbacka J, Wennberg B, Olofsson SO, Taskinen MR, Borén J: **A new combined multicompartmental model for apolipoprotein B-100 and triglyceride metabolism in VLDL subfractions.** *Journal of Lipid Research* 2005, **46**:58–67.
32. Adiels M, Borén J, Caslake MJ, Stewart P, Soro A, Westerbacka J, Wennberg B, Olofsson SO, Packard C, Taskinen MR: **Overproduction of VLDL1 Driven by Hyperglycemia Is a Dominant Feature of Diabetic Dyslipidemia.** *Arteriosclerosis, Thrombosis, and Vascular Biology* 2005, **25**(8):1697–1703.
33. Adiels M: **A compartmental model for kinetics of apolipoprotein B-100 and triglycerides in VLDL<sub>1</sub> and VLDL<sub>2</sub> in normolipidemic subjects.** *Licentiate thesis,* Chalmers University of Technology, Göteborg 2002.
34. Adiels M, Olofsson SO, Taskinen MR, Borén J: **Overproduction of very low-density lipoproteins is the hallmark of the dyslipidemia in the metabolic syndrome.** *Arteriosclerosis, thrombosis, and vascular biology* 2008, **28**(7):1225–1236.
35. Hürzeler M, Künsch HR: **Approximating and Maximizing the Likelihood for a General State-Space Model.** In *Sequential Monte Carlo Methods in Practice.* New York: Springer; 2001.
36. Doucet A, de Freitas N, Murphy KP, Russell SJ: **Rao-Blackwellised Particle Filtering for Dynamic Bayesian Networks.** In *UAI '00: Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence.* San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.; 2000:176–183.
37. Kloeden PE, Platen E: *Numerical solution of stochastic differential equations.* Berlin: Springer Verlag; 1999.

doi:10.1186/1752-0509-7-8

**Cite this article as:** Krengel et al.: A continuous-time adaptive particle filter for estimations under measurement time uncertainties with an application to a plasma-leucine mixed effects model. *BMC Systems Biology* 2013 **7**:8.