J Pharmacokinet Pharmacodyn (2015) 42:191–209
DOI 10.1007/s10928-015-9409-1

ORIGINAL PAPER

Using sensitivity equations for computing gradients of the FOCE and FOCEI approximations to the population likelihood

Joachim Almquist[1,2] · **Jacob Leander**[1,3] · **Mats Jirstrand**[1]

Received: 11 September 2014 / Accepted: 23 February 2015 / Published online: 24 March 2015

Abstract The first order conditional estimation (FOCE) method is still one of the parameter estimation workhorses for nonlinear mixed effects (NLME) modeling used in population pharmacokinetics and pharmacodynamics. However, because this method involves two nested levels of optimizations, with respect to the empirical Bayes estimates and the population parameters, FOCE may be numerically unstable and have long run times, issues which are most apparent for models requiring numerical integration of differential equations. We propose an alternative implementation of the FOCE method, and the related FOCEI, for parameter estimation in NLME models. Instead of obtaining the gradients needed for the two levels of quasi-Newton optimizations from the standard finite difference approximation, gradients are computed using so called sensitivity equations. The advantages of this approach were demonstrated using different versions of a pharmacokinetic model defined by nonlinear differential equations. We show that both the accuracy and precision of gradients can be improved extensively, which will increase the chances of a successfully converging parameter estimation. We also show that the proposed approach can lead to markedly reduced computational times. The accumulated effect of the novel gradient computations ranged from a 10-fold decrease in run times for the least complex model when comparing to forward finite differences, to a substantial 100-fold decrease for the most complex model when comparing to central finite differences. Considering the use of finite differences in for instance NONMEM and Phoenix NLME, our results suggests that significant improvements in the execution of FOCE are possible and that the approach of sensitivity equations should be carefully considered for both levels of optimization.

Keywords Nonlinear mixed effects modeling · First order conditional estimation (FOCE) · Sensitivity equations

✉ Joachim Almquist
joachim.almquist@fcc.chalmers.se

[1] Fraunhofer-Chalmers Centre, Chalmers Science Park, 41288 Gothenburg, Sweden

[2] Systems and Synthetic Biology, Department of Biology and Biological Engineering, Chalmers University of Technology, 41296 Gothenburg, Sweden

[3] Department of Mathematical Sciences, Chalmers University of Technology and University of Gothenburg, 41296 Gothenburg, Sweden

Introduction

Nonlinear mixed effects (NLME) models are suitable in situations where sparse time-series data is collected from a population of individuals exhibiting inter-individual variability [10]. This property has rendered NLME models popular in both pharmacokinetics and pharmacodynamics, and several public and commercial software packages have been developed for performing NLME modeling within these fields [13]. These modeling softwares include the well-known NONMEM [5], which was the first program to be developed and still is one of the most widely used, but also a number of other programs such as Phoenix NLME [21] and Monolix [15]. A core part of their functionality consist of various methods for addressing the problem of parameter estimation in NLME models, and several studies have been devoted to describing and comparing different aspects of these methods [4, 8, 9, 11, 12, 22].

The "mixed effects" in NLME refers to the fact that these models contain both fixed effect parameters, having the same value for all individuals, and random effect parameters, whose value differ from one individual to another and whose distribution in the population is determined by some statistical model. A common approach to the parameter estimation problem in NLME models is based on maximizing the so called population likelihood. The population likelihood is a function of the fixed effect parameters only, and it is obtained by marginalizing out the random effects from the joint distribution of data and random effects. However, the integral required for the marginalization lacks a closed-form solution for all realistic problems. Because of this, maximum likelihood parameter estimation for NLME models revolves around different numerical approximation methods for computing this integral. One of the main approaches for tackling the problem is a class of related methods based on the so called Laplacian approximation [25]. It includes the popular and widely used first order conditional estimation (FOCE) method, which is a special case of the closely related FOCE with interaction (FOCEI). With the FOCE and FOCEI methods, the approximation of the integral involves a Taylor expansion around the values of the random effect parameters that maximize the joint distribution. This means that one optimization problem per individual has to be solved for every evaluation of the approximated population likelihood. Since the aim is to maximize the (approximated) population likelihood, which constitutes the original optimization problem, conditional estimation methods such as FOCE produce a parameter estimation problem involving two nested layers of optimizations. For some NLME parameter estimation problems this results in long execution times, and in difficulties with numerical precision making the optimizations unstable and limiting the precision of estimates and the ability of obtaining confidence intervals. These issues are particularly pronounced for models that are formulated by systems of differential equations which are lacking analytical solutions [4, 7, 8].

The optimization problems resulting from the FOCE and FOCEI approximations, and other closely related approximations, are typically solved using gradient-based optimization methods such as the Broyden–Fletcher–Goldfarb–Shanno (BFGS) quasi-Newton method [20]. For problems where analytical expressions for the function and its gradient are not available, it is common that gradients are computed by finite difference approximations. We instead propose another approach for determining the gradient of the FOCE and FOCEI approximations of the population likelihood. Our approach is based on formally differentiating the likelihoods used at the two levels of optimization, and computing the required derivatives of the model state variables using so called sensitivity equations. The proposed approach for computing gradients is readily applicable for the inner level of the nested optimization problem. However, we also derive the necessary theory for computing gradients through the approach of sensitivity equations at the outer level optimization. This step is the more challenging, and requires that sensitivities up to second order of the state variables with respect to the parameters and random parameters are obtained. Being able to compute the gradient of the FOCE or FOCEI approximations of the population likelihood using the approach introduced in this paper is a great advantage as it circumvents the need for repeatedly having to solve the inner level optimization problem for obtaining the outer level gradients from a finite difference approximation.

This paper is organized in the following way. First, the mathematical theory is introduced. Here we recapitulate NLME models based on differential equations, including the formulation of the population likelihood and its approximations, as well as derive expressions for both the gradients of the individual joint log-likelihoods with respect to the random effect parameters, used for the inner level optimization problems, and the gradient of the approximate population likelihood with respect to the fixed effect parameters, used for the outer level optimization problem. Then, we apply the sensitivity approach for computing the gradients for different versions of a benchmark model. Compared to the finite difference approximation, the proposed approach leads to both higher precision and better accuracy of the gradient, as well as decreased computational times. Finally, the presented results are discussed and possible future extensions are outlined.

Theory

Various definitions and results from matrix calculus are used in the derivations of this section. These can be found in the "Appendix 1" section.

The nonlinear mixed effects model

Consider a population of N subjects and let the ith individual be described by the dynamical system

$$\begin{aligned} \frac{d\mathbf{x}_i(t)}{dt} &= \mathbf{f}\left(\mathbf{x}_i(t), t, \mathbf{Z}_i(t), \boldsymbol{\theta}, \boldsymbol{\eta}_i\right) \\ \mathbf{x}_i(t_0) &= \mathbf{x}_{0i}\left(\mathbf{Z}_i(t_0), \boldsymbol{\theta}, \boldsymbol{\eta}_i\right), \end{aligned} \tag{1}$$

where $\mathbf{x}_i(t)$ is a set of state variables, which for instance could be used to describe a drug concentration in one or more compartments, and where $\mathbf{Z}_i(t)$ is a set of possibly time dependent covariates, $\boldsymbol{\theta}$ a set of fixed effects parameters, and $\boldsymbol{\eta}_i$ a set of random effect parameters which are multivariate normally distributed with zero mean and covariance $\boldsymbol{\Omega}$. The covariance matrix $\boldsymbol{\Omega}$ is in general unknown and will therefore typically contain parameters

subject to estimation. These parameters will for convenience of notation be included in the fixed effect parameter vector $\boldsymbol{\theta}$. Fixed effects parameters will hence be used to refer to all parameters that are not random, not being limited for parameters appearing in the model differential equations. A model for the jth observation of the ith individual at time t_{j_i} is defined by

$$\mathbf{y}_{ij} = \mathbf{h}\left(\mathbf{x}_{ij}, t_{j_i}, \mathbf{Z}_i(t_{j_i}), \boldsymbol{\theta}, \boldsymbol{\eta}_i\right) + \mathbf{e}_{ij}, \tag{2}$$

where

$$\mathbf{e}_{ij} \in N\Big(\boldsymbol{\theta}, \mathbf{R}_{ij}\left(\mathbf{x}_{ij}, t_{j_i}, \mathbf{Z}_i(t_{j_i}), \boldsymbol{\theta}, \boldsymbol{\eta}_i\right)\Big), \tag{3}$$

and where the index notation ij is used as a short form for denoting the ith individual at the jth observation. Note that any fixed effect parameters of the observational model are included in $\boldsymbol{\theta}$. Furthermore, we let the expected value of the discrete-time observation model be denoted by

$$\hat{\mathbf{y}}_{ij} = \mathbf{E}\left[\mathbf{y}_{ij}\right]. \tag{4}$$

The population likelihood

Given a set of experimental observations, $\mathbf{d}_{ij}$, for the individuals $i = 1, \ldots, N$ at the time points t_{j_i}, where $j = 1, \ldots, n_i$, we define the residuals

$$\boldsymbol{\epsilon}_{ij} = \mathbf{d}_{ij} - \hat{\mathbf{y}}_{ij}, \tag{5}$$

and write the population likelihood

$$L(\boldsymbol{\theta}) = \prod_{i=1}^{N} \int p_1\left(\mathbf{d}_i|\boldsymbol{\theta}, \boldsymbol{\eta}_i\right) p_2\left(\boldsymbol{\eta}_i|\boldsymbol{\theta}\right) d\boldsymbol{\eta}_i, \tag{6}$$

where

$$p_1\left(\mathbf{d}_i|\boldsymbol{\theta}, \boldsymbol{\eta}_i\right) = \prod_{j=1}^{n_i} \frac{\exp\left(-\frac{1}{2}\boldsymbol{\epsilon}_{ij}^T \mathbf{R}_{ij}^{-1} \boldsymbol{\epsilon}_{ij}\right)}{\sqrt{\det\left(2\pi\mathbf{R}_{ij}\right)}} \tag{7}$$

and

$$p_2\left(\boldsymbol{\eta}_i|\boldsymbol{\theta}\right) = \frac{\exp\left(-\frac{1}{2}\boldsymbol{\eta}_i^T \boldsymbol{\Omega}^{-1} \boldsymbol{\eta}_i\right)}{\sqrt{\det\left(2\pi\boldsymbol{\Omega}\right)}}, \tag{8}$$

and where $\mathbf{d}_i$ is used to denote the collection of data from all time points for the ith individual.

The FOCE and FOCEI approximations

The marginalization with respect to $\boldsymbol{\eta}_i$ in Eq. 6 does not have a closed form solution. By writing Eq. 6 on the form

$$L(\boldsymbol{\theta}) = \prod_{i=1}^{N} \int \exp(l_i)\, d\boldsymbol{\eta}_i, \tag{9}$$

where the individual joint log-likelihoods are

$$\begin{aligned} l_i = &-\frac{1}{2}\sum_{j=1}^{n_i}\left(\boldsymbol{\epsilon}_{ij}^T \mathbf{R}_{ij}^{-1} \boldsymbol{\epsilon}_{ij} + \log\det\left(2\pi\mathbf{R}_{ij}\right)\right) \\ &-\frac{1}{2}\boldsymbol{\eta}_i^T \boldsymbol{\Omega}^{-1} \boldsymbol{\eta}_i - \frac{1}{2}\log\det\left(2\pi\boldsymbol{\Omega}\right), \end{aligned} \tag{10}$$

a closed form solution can be obtained by approximating the function l_i with a second order Taylor expansion with respect to $\boldsymbol{\eta}_i$. This is the well-known Laplacian approximation. Furthermore, we let the point around which the Taylor expansion is done to be conditioned on the $\boldsymbol{\eta}_i$ maximizing l_i, here denoted by $\boldsymbol{\eta}_i^*$; I.e., the expansion is done at the mode of the posterior distribution. Thus, the approximate population likelihood, L_L, becomes

$$L(\boldsymbol{\theta}) \approx L_L(\boldsymbol{\theta}) = \prod_{i=1}^{N}\left(\exp\left(l_i(\boldsymbol{\eta}_i^*)\right)\det\left[\frac{-\Delta l_i(\boldsymbol{\eta}_i^*)}{2\pi}\right]^{-\frac{1}{2}}\right). \tag{11}$$

Here, the Hessian $\Delta l_i(\boldsymbol{\eta}_i^*)$ is obtained by first differentiating l_i twice with respect to $\boldsymbol{\eta}_i$, and evaluating at $\boldsymbol{\eta}_i^*$. If we let η_{ik} denote the kth component of $\boldsymbol{\eta}_i$, we have

$$\begin{aligned} \frac{dl_i}{d\eta_{ik}} = -\frac{1}{2}\sum_{j=1}^{n_i}\Bigg(&2\boldsymbol{\epsilon}_{ij}^T \mathbf{R}_{ij}^{-1}\frac{d\boldsymbol{\epsilon}_{ij}}{d\eta_{ik}} - \boldsymbol{\epsilon}_{ij}^T \mathbf{R}_{ij}^{-1}\frac{d\mathbf{R}_{ij}}{d\eta_{ik}}\mathbf{R}_{ij}^{-1}\boldsymbol{\epsilon}_{ij} \\ &+ \operatorname{tr}\left[\mathbf{R}_{\mathrm{ij}}^{-1}\frac{d\mathbf{R}_{ij}}{d\eta_{ik}}\right]\Bigg) - \boldsymbol{\eta}_i^T \boldsymbol{\Omega}^{-1}\frac{d\boldsymbol{\eta}_i}{d\eta_{ik}}. \end{aligned} \tag{12}$$

Differentiating component-wise again, now with respect to the lth component of $\boldsymbol{\eta}_i$, we get the elements of the Hessian

$$\begin{aligned} \frac{d^2 l_i}{d\eta_{ik}d\eta_{il}} = -\frac{1}{2}\sum_{j=1}^{n_i}\Bigg(&2\frac{d\boldsymbol{\epsilon}_{ij}^T}{d\eta_{il}}\mathbf{R}_{ij}^{-1}\frac{d\boldsymbol{\epsilon}_{ij}}{d\eta_{ik}} - 2\boldsymbol{\epsilon}_{ij}^T \mathbf{R}_{ij}^{-1}\frac{d\mathbf{R}_{ij}}{d\eta_{il}}\mathbf{R}_{ij}^{-1}\frac{d\boldsymbol{\epsilon}_{ij}}{d\eta_{ik}} \\ &+ 2\boldsymbol{\epsilon}_{ij}^T \mathbf{R}_{ij}^{-1}\frac{d^2\boldsymbol{\epsilon}_{ij}}{d\eta_{ik}d\eta_{il}} - \boldsymbol{\epsilon}_{ij}^T \mathbf{R}_{ij}^{-1}\frac{d^2\mathbf{R}_{ij}}{d\eta_{ik}d\eta_{il}}\mathbf{R}_{ij}^{-1}\boldsymbol{\epsilon}_{ij} \\ &+ 2\boldsymbol{\epsilon}_{ij}^T \mathbf{R}_{ij}^{-1}\frac{d\mathbf{R}_{ij}}{d\eta_{ik}}\mathbf{R}_{ij}^{-1}\frac{d\mathbf{R}_{ij}}{d\eta_{il}}\mathbf{R}_{ij}^{-1}\boldsymbol{\epsilon}_{ij} \\ &- 2\boldsymbol{\epsilon}_{ij}^T \mathbf{R}_{ij}^{-1}\frac{d\mathbf{R}_{ij}}{d\eta_{ik}}\mathbf{R}_{ij}^{-1}\frac{d\boldsymbol{\epsilon}_{ij}}{d\eta_{il}} - \operatorname{tr}\left[\mathbf{R}_{ij}^{-1}\frac{d\mathbf{R}_{ij}}{d\eta_{il}}\mathbf{R}_{ij}^{-1}\frac{d\mathbf{R}_{ij}}{d\eta_{ik}}\right] \\ &+ \operatorname{tr}\left[\mathbf{R}_{ij}^{-1}\frac{d^2\mathbf{R}_{ij}}{d\eta_{ik}d\eta_{il}}\right]\Bigg) - \frac{d\boldsymbol{\eta}_i^T}{d\eta_{il}}\boldsymbol{\Omega}^{-1}\frac{d\boldsymbol{\eta}_i}{d\eta_{ik}}, \end{aligned} \tag{13}$$

where the last term is really just the klth element of $\boldsymbol{\Omega}^{-1}$, Ω_{kl}^{-1}. The expression for the elements of the Hessian may be approximated in different ways, with the main purpose of avoiding the need for computing the costly second order derivatives. We apply a first order approximation, where terms containing second order derivatives are ignored, and write the elements of the approximate Hessian, $\mathbf{H}_i$, as

$$\mathrm{H}_{ikl} = -\frac{1}{2}\sum_{j=1}^{n_i}\left(\mathbf{a}_l\,\mathbf{B}\,\mathbf{a}_k^T + \mathrm{tr}[-\mathbf{c}_l\,\mathbf{c}_k]\right) - \Omega_{kl}^{-1}, \tag{14}$$

where

$$\mathbf{a}_k = \left(\frac{d\boldsymbol{\epsilon}_{ij}^T}{d\eta_{ik}} - \boldsymbol{\epsilon}_{ij}^T\mathbf{R}_{ij}^{-1}\frac{d\mathbf{R}_{ij}}{d\eta_{ik}}\right), \tag{15}$$

$$\mathbf{B} = 2\mathbf{R}_{ij}^{-1}, \tag{16}$$

and

$$\mathbf{c}_k = \mathbf{R}_{ij}^{-1}\frac{d\mathbf{R}_{ij}}{d\eta_{ik}}. \tag{17}$$

This variant of the Laplacian approximation of the population likelihood is known as the first order conditional estimation with interaction (FOCEI) method. The closely related first order conditional estimation (FOCE) method is obtained by ignoring the dependence of the residual covariance matrix on the random effect parameters. The rationale for excluding the second order terms is that their expected values are zero for an appropriate model, as shown in the "Appendix 2" section. The Appendix also shows how the Hessian may be slightly further simplified, using similar arguments, to arrive at the variant of FOCE used in NONMEM. Those additional simplifications are however of relatively little importance from a computational point of view, since the components needed to evaluate these Hessian terms have to be provided for the remaining part of the Hessian anyway. We will therefore restrict the Hessian simplification by expectation to the second order terms only. Furthermore, we will from now on for convenience consider the logarithm of the FOCEI approximation to the population likelihood, L_F,

$$\log L(\boldsymbol{\theta}) \approx \log L_F(\boldsymbol{\theta}) = \sum_{i=1}^{N}\left(l_i(\boldsymbol{\eta}_i^*) - \frac{1}{2}\log\det\left[\frac{-\mathbf{H}_i(\boldsymbol{\eta}_i^*)}{2\pi}\right]\right). \tag{18}$$

Gradient of the individual joint log-likelihood with respect to the random effect parameters

We now turn to the computation of the gradient of the individual joint log-likelihoods, $l_i(\boldsymbol{\eta}_i)$, with respect to the random effect parameters, $\boldsymbol{\eta}_i$, using the approach of sensitivity equations. Consider the differentiation done in Eq. 12. Given values of $\boldsymbol{\theta}$ and $\boldsymbol{\eta}_i$, the quantities $\boldsymbol{\epsilon}_{ij}$, $\mathbf{R}_{ij}$, and $\boldsymbol{\Omega}$ can be obtained by solving the model equations. However, we additionally need to determine $d\boldsymbol{\epsilon}_{ij}/d\eta_{ik}$ and $d\mathbf{R}_{ij}/d\eta_{ik}$. Expanding the total derivative of these quantities we see that

$$\frac{d\boldsymbol{\epsilon}_{ij}}{d\eta_{ik}} = \frac{d\left(\mathbf{d}_{ij} - \hat{\mathbf{y}}_{ij}\right)}{d\eta_{ik}} = -\left(\frac{\partial\mathbf{h}}{\partial\eta_{ik}} + \frac{\partial\mathbf{h}}{\partial\mathbf{x}_{ij}}\frac{d\mathbf{x}_{ij}}{d\eta_{ik}}\right), \tag{19}$$

and

$$\frac{d\mathbf{R}_{ij}}{d\eta_{ik}} = \frac{\partial\mathbf{R}_{ij}}{\partial\eta_{ik}} + \frac{\partial\mathbf{R}_{ij}}{\partial\mathbf{x}_{ij}}\frac{d\mathbf{x}_{ij}}{d\eta_{ik}}. \tag{20}$$

The derivatives of $\mathbf{h}$ and $\mathbf{R}_{ij}$ are readily obtained since these expressions are given explicitly by the model formulation. In contrast, the derivative of the state variables, $\mathbf{x}_{ij}$, are not directly available but can be computed from the so called sensitivity equations. The sensitivity equations are a set of differential equations which are derived by differentiating the original system of differential equations (and the corresponding initial conditions) with respect to each random effect parameter η_{ik},

$$\begin{aligned}\frac{d}{dt}\left(\frac{d\mathbf{x}_i}{d\eta_{ik}}\right) &= \frac{\partial\mathbf{f}}{\partial\eta_{ik}} + \frac{\partial\mathbf{f}}{\partial\mathbf{x}_i}\left(\frac{d\mathbf{x}_i}{d\eta_{ik}}\right)\\ \left(\frac{d\mathbf{x}_i}{d\eta_{ik}}\right)(t_0) &= \frac{\partial\mathbf{x}_{0i}}{\partial\eta_{ik}}.\end{aligned} \tag{21}$$

The solution to the sensitivity equations can be used to evaluate the derivatives in Eqs. 19 and 20, which in turn are needed for the gradient of the individual joint log-likelihoods. Importantly, these derivatives are also used for computing the approximate Hessian, Eq. 14, appearing in the approximate population log-likelihood.

In the unusual event that one or more of the random effect parameters only appear in the observational model, all sensitivities of the state variables with respect to those parameters are trivially zero. Note also that the sensitivity equations for all but trivial models involve the original state variables, which means that the original system of differential equations has to be solved simultaneously. Thus, if there are q non-trivial sensitivities and n state variables, the total number of differential equations that has to be solved in order to be able to compute l_i and $dl_i/d\boldsymbol{\eta}_i$ for each individual is

$$n(1+q). \tag{22}$$

Gradient of the approximate population log-likelihood with respect to the fixed effect parameters

We now derive the expression for the gradient of the approximate population log-likelihood, $\log L_F(\boldsymbol{\theta})$, with respect to the parameter vector $\boldsymbol{\theta}$. Differentiating $\log L_F$ with respect to the mth element of $\boldsymbol{\theta}$ gives

$$\frac{\log L_F}{d\theta_m} = \sum_{i=1}^{N}\left(\frac{dl_i(\boldsymbol{\eta}_i^*)}{d\theta_m} - \frac{1}{2}\mathrm{tr}\left[\mathbf{H}_i^{-1}(\boldsymbol{\eta}_i^*)\frac{d\mathbf{H}_i(\boldsymbol{\eta}_i^*)}{d\theta_m}\right]\right). \tag{23}$$

Here it must be emphasized that all derivatives with respect to components of the parameter vector $\boldsymbol{\theta}$ are taken *after* replacing $\boldsymbol{\eta}_i$ with $\boldsymbol{\eta}_i^*$. This is critical since $\boldsymbol{\eta}_i^*$ is an implicit

function of theta, $\boldsymbol{\eta}_i^* = \boldsymbol{\eta}_i^*(\boldsymbol{\theta})$. In other words, we have to account for the fact that the $\boldsymbol{\eta}_i$ maximizing the individual joint log-likelihood changes as $\boldsymbol{\theta}$ changes.

To determine the total derivatives with respect to components of the parameter vector $\boldsymbol{\theta}$ we will be needing the following result. Consider a function $\mathbf{v}$ which may depend directly on the parameters $\boldsymbol{\theta}$ and $\boldsymbol{\eta}_i$, and on the auxiliary function $\mathbf{w}$ representing any indirect dependencies of these parameters,

$$\mathbf{v} = \mathbf{v}\big(\mathbf{w}(\boldsymbol{\theta}, \boldsymbol{\eta}_i), \boldsymbol{\theta}, \boldsymbol{\eta}_i\big). \tag{24}$$

We furthermore introduce the function $\mathbf{z}$ to denote the evaluation of $\mathbf{v}$ at $\boldsymbol{\eta}_i = \boldsymbol{\eta}_i^*(\boldsymbol{\theta})$,

$$\mathbf{z} = \mathbf{z}\big(\mathbf{w}(\boldsymbol{\theta}, \boldsymbol{\eta}_i^*(\boldsymbol{\theta})), \boldsymbol{\theta}, \boldsymbol{\eta}_i^*(\boldsymbol{\theta})\big) = \mathbf{v}|_{\boldsymbol{\eta}_i = \boldsymbol{\eta}_i^*(\boldsymbol{\theta})}. \tag{25}$$

Separating the complete dependence of $\mathbf{z}$ on $\boldsymbol{\theta}$ into partial dependencies we get that

$$\begin{aligned}
\frac{d}{d\boldsymbol{\theta}}\left(\mathbf{v}|_{\boldsymbol{\eta}_i=\boldsymbol{\eta}_i^*(\boldsymbol{\theta})}\right) &= \frac{d\mathbf{z}}{d\boldsymbol{\theta}} \\
&= \frac{\partial \mathbf{z}}{\partial \mathbf{w}}\frac{d\mathbf{w}}{d\boldsymbol{\theta}} + \frac{\partial \mathbf{z}}{\partial \boldsymbol{\theta}} + \frac{\partial \mathbf{z}}{\partial \boldsymbol{\eta}_i^*}\frac{d\boldsymbol{\eta}_i^*}{d\boldsymbol{\theta}} \\
&= \frac{\partial \mathbf{z}}{\partial \mathbf{w}}\frac{\partial \mathbf{w}}{\partial \boldsymbol{\theta}} + \frac{\partial \mathbf{z}}{\partial \mathbf{w}}\frac{\partial \mathbf{w}}{\partial \boldsymbol{\eta}_i^*}\frac{d\boldsymbol{\eta}_i^*}{d\boldsymbol{\theta}} + \frac{\partial \mathbf{z}}{\partial \boldsymbol{\theta}} + \frac{\partial \mathbf{z}}{\partial \boldsymbol{\eta}_i^*}\frac{d\boldsymbol{\eta}_i^*}{d\boldsymbol{\theta}} \\
&= \frac{\partial \mathbf{z}}{\partial \mathbf{w}}\frac{\partial \mathbf{w}}{\partial \boldsymbol{\theta}} + \frac{\partial \mathbf{z}}{\partial \boldsymbol{\theta}} + \frac{d\mathbf{z}}{d\boldsymbol{\eta}_i^*}\frac{d\boldsymbol{\eta}_i^*}{d\boldsymbol{\theta}} \\
&= \frac{\partial}{\partial \mathbf{w}}\left(\mathbf{v}|_{\boldsymbol{\eta}_i=\boldsymbol{\eta}_i^*(\boldsymbol{\theta})}\right)\frac{\partial \mathbf{w}}{\partial \boldsymbol{\theta}} + \frac{\partial}{\partial \boldsymbol{\theta}}\left(\mathbf{v}|_{\boldsymbol{\eta}_i=\boldsymbol{\eta}_i^*(\boldsymbol{\theta})}\right) \\
&\quad + \frac{d}{d\boldsymbol{\eta}_i^*}\left(\mathbf{v}|_{\boldsymbol{\eta}_i=\boldsymbol{\eta}_i^*(\boldsymbol{\theta})}\right)\frac{d\boldsymbol{\eta}_i^*}{d\boldsymbol{\theta}} \\
&= \left(\frac{\partial \mathbf{v}}{\partial \mathbf{w}}\frac{\partial \mathbf{w}}{\partial \boldsymbol{\theta}}\right)\Bigg|_{\boldsymbol{\eta}_i=\boldsymbol{\eta}_i^*(\boldsymbol{\theta})} + \left(\frac{\partial \mathbf{v}}{\partial \boldsymbol{\theta}}\right)\Bigg|_{\boldsymbol{\eta}_i=\boldsymbol{\eta}_i^*(\boldsymbol{\theta})} \\
&\quad + \left(\frac{d\mathbf{v}}{d\boldsymbol{\eta}_i}\right)\Bigg|_{\boldsymbol{\eta}_i=\boldsymbol{\eta}_i^*(\boldsymbol{\theta})}\frac{d\boldsymbol{\eta}_i^*}{d\boldsymbol{\theta}} \\
&= \frac{d\mathbf{v}}{d\boldsymbol{\theta}}\Bigg|_{\boldsymbol{\eta}_i=\boldsymbol{\eta}_i^*(\boldsymbol{\theta})} + \frac{d\mathbf{v}}{d\boldsymbol{\eta}_i}\Bigg|_{\boldsymbol{\eta}_i=\boldsymbol{\eta}_i^*(\boldsymbol{\theta})}\frac{d\boldsymbol{\eta}_i^*}{d\boldsymbol{\theta}}.
\end{aligned} \tag{26}$$

Thus, the total derivative with respect to $\boldsymbol{\theta}$ after insertion of $\boldsymbol{\eta}_i^*$ is equal to the sum of total derivatives with respect to $\boldsymbol{\theta}$ and $\boldsymbol{\eta}_i$ before insertion of $\boldsymbol{\eta}_i^*$, where the second derivative is multiplied with the sensitivity of the random effect optimum with respect to the parameters $\boldsymbol{\theta}$. It is straightforward to see that this result holds also when differentiating functions that only exhibit a subset of the possible direct and indirect dependencies of Eq. 24, for instance functions with just an indirect dependence on the two kind of parameters.

Applying the results from Eq. 26 to the first term within the summation of Eq. 23, we have that

$$\frac{dl_i(\boldsymbol{\eta}_i^*)}{d\theta_m} = \frac{dl_i(\boldsymbol{\eta}_i)}{d\theta_m}\Bigg|_{\boldsymbol{\eta}_i=\boldsymbol{\eta}_i^*(\boldsymbol{\theta})} + \frac{dl_i(\boldsymbol{\eta}_i)}{d\boldsymbol{\eta}_i}\Bigg|_{\boldsymbol{\eta}_i=\boldsymbol{\eta}_i^*(\boldsymbol{\theta})}\frac{d\boldsymbol{\eta}_i^*}{d\theta_m}. \tag{27}$$

However, since $dl_i/d\boldsymbol{\eta}_i$ evaluated at $\boldsymbol{\eta}_i^*$ is zero by definition, the second term of the right hand side of Eq. 27 disappears and

$$\begin{aligned}
\frac{dl_i(\boldsymbol{\eta}_i^*)}{d\theta_m} &= \frac{dl_i(\boldsymbol{\eta}_i)}{d\theta_m}\Bigg|_{\boldsymbol{\eta}_i=\boldsymbol{\eta}_i^*(\boldsymbol{\theta})} \\
&= \Bigg[-\frac{1}{2}\sum_{j=1}^{n_i}\left(2\boldsymbol{\epsilon}_{ij}^T\mathbf{R}_{ij}^{-1}\frac{d\boldsymbol{\epsilon}_{ij}}{d\theta_m} - \boldsymbol{\epsilon}_{ij}^T\mathbf{R}_{ij}^{-1}\frac{d\mathbf{R}_{ij}}{d\theta_m}\mathbf{R}_{ij}^{-1}\boldsymbol{\epsilon}_{ij}\right. \\
&\quad \left. + \operatorname{tr}\left[\mathbf{R}_{\mathrm{ij}}^{-1}\frac{d\mathbf{R}_{ij}}{d\theta_m}\right]\right) + \frac{1}{2}\boldsymbol{\eta}_i^T\,\boldsymbol{\Omega}^{-1}\frac{d\boldsymbol{\Omega}}{d\theta_m}\boldsymbol{\Omega}^{-1}\boldsymbol{\eta}_i \\
&\quad - \frac{1}{2}\operatorname{tr}\left[\boldsymbol{\Omega}^{-1}\frac{d\boldsymbol{\Omega}}{d\theta_m}\right]\Bigg]_{\boldsymbol{\eta}_i=\boldsymbol{\eta}_i^*(\boldsymbol{\theta})}.
\end{aligned} \tag{28}$$

Using asterisks to denote that $\boldsymbol{\eta}_i$ has been replaced with $\boldsymbol{\eta}_i^*$, we also get the following for the derivative of the second term within the summation of Eq. 23,

$$\begin{aligned}
\frac{d\mathrm{H}_{ikl}(\boldsymbol{\eta}_i^*)}{d\theta_m} = -\frac{1}{2}\sum_{j=1}^{n_i}\Bigg(&\frac{d\mathbf{a}_l^*}{d\theta_m}\mathbf{B}^*\,\mathbf{a}_k^{*T} + \mathbf{a}_l^*\frac{d\mathbf{B}^*}{d\theta_m}\mathbf{a}_k^{*T} + \mathbf{a}_l^*\,\mathbf{B}^*\frac{d\mathbf{a}_k^{*T}}{d\theta_m} \\
&+ \operatorname{tr}\left[-\frac{d\mathbf{c}_l^*}{d\theta_m}\mathbf{c}_k^* - \mathbf{c}_l^*\frac{d\mathbf{c}_k^*}{d\theta_m}\right]\Bigg) - \frac{d\Omega_{kl}^{-1}}{d\theta_m},
\end{aligned} \tag{29}$$

where

$$\begin{aligned}
\frac{d\mathbf{a}_k^*}{d\theta_m} = &\frac{d}{d\theta_m}\left(\frac{d\boldsymbol{\epsilon}_{ij}^T}{d\eta_{ik}}\right)^* - \frac{\boldsymbol{\epsilon}_{ij}^{*T}}{d\theta_m}\mathbf{R}_{ij}^{*-1}\left(\frac{d\mathbf{R}_{ij}}{d\eta_{ik}}\right)^* \\
&+ \boldsymbol{\epsilon}_{ij}^{*T}\mathbf{R}_{ij}^{*-1}\frac{d\mathbf{R}_{ij}^*}{d\theta_m}\mathbf{R}_{ij}^{*-1}\left(\frac{d\mathbf{R}_{ij}}{d\eta_{ik}}\right)^* \\
&- \boldsymbol{\epsilon}_{ij}^{*T}\mathbf{R}_{ij}^{*-1}\frac{d}{d\theta_m}\left(\frac{d\mathbf{R}_{ij}}{d\eta_{ik}}\right)^*,
\end{aligned} \tag{30}$$

$$\frac{d\mathbf{B}^*}{d\theta_m} = -2\mathbf{R}_{ij}^{*-1}\frac{d\mathbf{R}_{ij}^*}{d\theta_m}\mathbf{R}_{ij}^{*-1}, \tag{31}$$

and

$$\frac{d\mathbf{c}_k^*}{d\theta_m} = -\mathbf{R}_{ij}^{*-1}\frac{d\mathbf{R}_{ij}^*}{d\theta_m}\mathbf{R}_{ij}^{*-1}\left(\frac{d\mathbf{R}_{ij}}{d\eta_{ik}}\right)^* + \mathbf{R}_{ij}^{*-1}\frac{d}{d\theta_m}\left(\frac{d\mathbf{R}_{ij}}{d\eta_{ik}}\right)^*. \tag{32}$$

We now continue to expand the terms in Eqs. 28–32 containing derivatives with respect to θ_m. The terms $d\boldsymbol{\Omega}/d\theta_m$ and $d\Omega_{kl}^{-1}/d\theta_m$ are obtainable by straightforward differentiation. Noting that the terms $\boldsymbol{\epsilon}_{ij}^*$, $(d\boldsymbol{\epsilon}_{ij}/d\eta_{ik})^*$, $\mathbf{R}_{ij}^*$, and $(d\mathbf{R}_{ij}/d\eta_{ik})^*$, have indirect and/or direct dependence on $\boldsymbol{\theta}$

and $\boldsymbol{\eta}_i^*$, we apply the results from Eq. 26 and expand the remaining derivatives. First,

$$\frac{d\boldsymbol{\epsilon}_{ij}^*}{d\theta_m} = \left.\frac{d\boldsymbol{\epsilon}_{ij}}{d\theta_m}\right|_{\boldsymbol{\eta}_i=\boldsymbol{\eta}_i^*(\boldsymbol{\theta})} + \left.\frac{d\boldsymbol{\epsilon}_{ij}}{d\boldsymbol{\eta}_i}\right|_{\boldsymbol{\eta}_i=\boldsymbol{\eta}_i^*(\boldsymbol{\theta})}\frac{d\boldsymbol{\eta}_i^*}{d\theta_m}. \tag{33}$$

Here, $d\boldsymbol{\epsilon}_{ij}/d\boldsymbol{\eta}_i$ was determined previously in Eq. 19, and the derivative in the first term is given by

$$\frac{d\boldsymbol{\epsilon}_{ij}}{d\theta_m} = \frac{d(\mathbf{d}_{ij} - \hat{\mathbf{y}}_{ij})}{d\theta_m} = -\left(\frac{\partial \mathbf{h}}{\partial \theta_m} + \frac{\partial \mathbf{h}}{\partial \mathbf{x}_{ij}}\frac{d\mathbf{x}_{ij}}{d\theta_m}\right). \tag{34}$$

The sensitivity of the random effect optimum with respect to the fixed effect parameters, $d\boldsymbol{\eta}_i^*/d\boldsymbol{\theta}$, must also be determined, which we will return to later. Then,

$$\frac{d\mathbf{R}_{ij}^*}{d\theta_m} = \left.\frac{d\mathbf{R}_{ij}}{d\theta_m}\right|_{\boldsymbol{\eta}_i=\boldsymbol{\eta}_i^*(\boldsymbol{\theta})} + \left.\frac{d\mathbf{R}_{ij}}{d\boldsymbol{\eta}_i}\right|_{\boldsymbol{\eta}_i=\boldsymbol{\eta}_i^*(\boldsymbol{\theta})}\frac{d\boldsymbol{\eta}_i^*}{d\theta_m}, \tag{35}$$

where $d\mathbf{R}_{ij}/d\boldsymbol{\eta}_i$ was determined in Eq. 20, and

$$\frac{d\mathbf{R}_{ij}}{d\theta_m} = \frac{\partial \mathbf{R}_{ij}}{\partial \theta_m} + \frac{\partial \mathbf{R}_{ij}}{\partial \mathbf{x}_{ij}}\frac{d\mathbf{x}_{ij}}{d\theta_m}. \tag{36}$$

Next,

$$\begin{aligned}
&\frac{d}{d\theta_m}\left(\left.\frac{d\boldsymbol{\epsilon}_{ij}}{d\eta_{ik}}\right|_{\boldsymbol{\eta}_i=\boldsymbol{\eta}_i^*(\boldsymbol{\theta})}\right)\\
&= \left(\frac{d}{d\theta_m}\left(\frac{d\boldsymbol{\epsilon}_{ij}}{d\eta_{ik}}\right)\right)\Bigg|_{\boldsymbol{\eta}_i=\boldsymbol{\eta}_i^*(\boldsymbol{\theta})} + \left(\frac{d}{d\boldsymbol{\eta}}\left(\frac{d\boldsymbol{\epsilon}_{ij}}{d\eta_{ik}}\right)\right)\Bigg|_{\boldsymbol{\eta}_i=\boldsymbol{\eta}_i^*(\boldsymbol{\theta})}\frac{d\boldsymbol{\eta}_i^*}{d\theta_m}\\
&= \left(\frac{d}{d\theta_m}\left(\frac{d\boldsymbol{\epsilon}_{ij}}{d\eta_{ik}}\right)\right)\Bigg|_{\boldsymbol{\eta}_i=\boldsymbol{\eta}_i^*(\boldsymbol{\theta})} + \sum_l \left(\frac{d}{d\eta_{il}}\left(\frac{d\boldsymbol{\epsilon}_{ij}}{d\eta_{ik}}\right)\right)\Bigg|_{\boldsymbol{\eta}_i=\boldsymbol{\eta}_i^*(\boldsymbol{\theta})}\frac{d\eta_{il}^*}{d\theta_m}\\
&= -\left(\frac{\partial^2 \mathbf{h}}{\partial \eta_{ik}\partial \theta_m} + \frac{\partial^2 \mathbf{h}}{\partial \eta_{ik}\partial \mathbf{x}_{ij}}\frac{d\mathbf{x}_{ij}}{d\theta_m} + \left(\frac{\partial^2 \mathbf{h}}{\partial \mathbf{x}_{ij}\partial \theta_m} + \frac{\partial^2 \mathbf{h}}{\partial \mathbf{x}_{ij}^2}\frac{d\mathbf{x}_{ij}}{d\theta_m}\right)\frac{d\mathbf{x}_{ij}}{d\eta_{ik}}\right.\\
&\quad \left. + \frac{\partial \mathbf{h}}{\partial \mathbf{x}_{ij}}\frac{d^2\mathbf{x}_{ij}}{d\eta_{ik}d\theta_m}\right)\Bigg|_{\boldsymbol{\eta}_i=\boldsymbol{\eta}_i^*(\boldsymbol{\theta})} - \sum_l \left(\frac{\partial^2 \mathbf{h}}{\partial \eta_{ik}\partial \eta_{il}} + \frac{\partial^2 \mathbf{h}}{\partial \eta_{ik}\partial \mathbf{x}_{ij}}\frac{d\mathbf{x}_{ij}}{d\eta_{il}}\right.\\
&\quad \left. + \left(\frac{\partial^2 \mathbf{h}}{\partial \mathbf{x}_{ij}\partial \eta_{il}} + \frac{\partial^2 \mathbf{h}}{\partial \mathbf{x}_{ij}^2}\frac{d\mathbf{x}_{ij}}{d\eta_{il}}\right)\frac{d\mathbf{x}_{ij}}{d\eta_{ik}} + \frac{\partial \mathbf{h}}{\partial \mathbf{x}_{ij}}\frac{d^2\mathbf{x}_{ij}}{d\eta_{ik}d\eta_{il}}\right)\Bigg|_{\boldsymbol{\eta}_i=\boldsymbol{\eta}_i^*(\boldsymbol{\theta})}\frac{d\eta_{il}^*}{d\theta_m},
\end{aligned} \tag{37}$$

where we after the third equality have used the results from Eq. 19. The derivative of $(d\mathbf{R}_{ij}/d\eta_{ik})^*$ with respect to θ_m is done in a highly similar way and is left to the reader as an exercise.

In the above expressions, derivatives of $\mathbf{h}$ and $\mathbf{R}_{ij}$ are obtained by direct differentiation. The derivatives of the state variables are determined by the previously derived sensitivity equation in Eq. 21 and by the additional sensitivity equations

$$\begin{aligned}
&\frac{d}{dt}\left(\frac{d\mathbf{x}_i}{d\theta_m}\right) = \frac{\partial \mathbf{f}}{\partial \theta_m} + \frac{\partial \mathbf{f}}{\partial \mathbf{x}_i}\left(\frac{d\mathbf{x}_i}{d\theta_m}\right)\\
&\left(\frac{d\mathbf{x}_i}{d\theta_m}\right)(t_0) = \frac{\partial \mathbf{x}_{0i}}{\partial \theta_m},
\end{aligned} \tag{38}$$

$$\begin{aligned}
\frac{d}{dt}\left(\frac{d^2\mathbf{x}_i}{d\eta_{ik}d\theta_m}\right) &= \frac{\partial^2 \mathbf{f}}{\partial \eta_{ik}\partial \theta_m} + \frac{\partial^2 \mathbf{f}}{\partial \eta_{ik}\partial \mathbf{x}_i}\frac{d\mathbf{x}_i}{d\theta_m}\\
&\quad + \left(\frac{\partial^2 \mathbf{f}}{\partial \mathbf{x}_i\partial \theta_m} + \frac{\partial^2 \mathbf{f}}{\partial^2 \mathbf{x}_i}\frac{d\mathbf{x}_i}{d\theta_m}\right)\left(\frac{d\mathbf{x}_i}{d\eta_{ik}}\right)\\
&\quad + \frac{\partial \mathbf{f}}{\partial \mathbf{x}_i}\left(\frac{d^2\mathbf{x}_i}{d\eta_{ik}d\theta_m}\right)\\
\left(\frac{d^2\mathbf{x}_i}{d\eta_{ik}d\theta_m}\right)(t_0) &= \frac{\partial^2 \mathbf{x}_{0i}}{\partial \eta_{ik}\partial \theta_m},
\end{aligned} \tag{39}$$

and

$$\begin{aligned}
\frac{d}{dt}\left(\frac{d^2\mathbf{x}_i}{d\eta_{ik}d\eta_{il}}\right) &= \frac{\partial^2 \mathbf{f}}{\partial \eta_{ik}\partial \eta_{il}} + \frac{\partial^2 \mathbf{f}}{\partial \eta_{ik}\partial \mathbf{x}_i}\frac{d\mathbf{x}_i}{d\eta_{il}}\\
&\quad + \left(\frac{\partial^2 \mathbf{f}}{\partial \mathbf{x}_i\partial \eta_{il}} + \frac{\partial^2 \mathbf{f}}{\partial^2 \mathbf{x}_i}\frac{d\mathbf{x}_i}{d\eta_{il}}\right)\left(\frac{d\mathbf{x}_i}{d\eta_{ik}}\right)\\
&\quad + \frac{\partial \mathbf{f}}{\partial \mathbf{x}_i}\left(\frac{d^2\mathbf{x}_i}{d\eta_{ik}d\eta_{il}}\right)\\
\left(\frac{d^2\mathbf{x}_i}{d\eta_{ik}d\eta_{il}}\right)(t_0) &= \frac{\partial^2 \mathbf{x}_{0i}}{\partial \eta_{ik}\partial \eta_{il}}.
\end{aligned} \tag{40}$$

As noted previously, all sensitivity equations must be solved simultaneously with the original differential equations for all but trivial models. However, since one or more parameters in the vector $\boldsymbol{\theta}$ may not appear in the differential equation part of the model (such as parameters appearing only in $\boldsymbol{\Omega}$), there may be sensitivities which are trivially zero. If there are p non-trivial sensitivities among the parameters in $\boldsymbol{\theta}$, q non-trivial sensitivities among the parameters in $\boldsymbol{\eta}$, and n state variables, the total number of differential equations that has to be solved in order to be able to compute $\log L_F$ and $d\log L_F/d\boldsymbol{\theta}$ for each individual is

$$n(1+q)(1+p+q/2). \tag{41}$$

Finally, we need to determine $d\boldsymbol{\eta}_i^*/d\boldsymbol{\theta}$. At the the optimum of each individual joint log-likelihood we have that

$$\frac{dl_i}{d\boldsymbol{\eta}_i} = \boldsymbol{0}, \tag{42}$$

or put differently,

$$\left.\frac{dl_i}{d\boldsymbol{\eta}_i}\right|_{\boldsymbol{\eta}_i=\boldsymbol{\eta}_i^*(\boldsymbol{\theta})} = \boldsymbol{0}. \tag{43}$$

This equality holds for any $\boldsymbol{\theta}$, and thus

$$\frac{d}{d\boldsymbol{\theta}}\left(\left.\frac{dl_i}{d\boldsymbol{\eta}_i}\right|_{\boldsymbol{\eta}_i=\boldsymbol{\eta}_i^*(\boldsymbol{\theta})}\right)=\mathbf{0}. \tag{44}$$

Recognizing that $dl_i/d\boldsymbol{\eta}_i$ fulfills the requirements of applying the results from Eq. 26, we can write this as

$$\frac{d}{d\boldsymbol{\theta}}\left(\left.\frac{dl_i}{d\boldsymbol{\eta}_i}\right|_{\boldsymbol{\eta}_i=\boldsymbol{\eta}_i^*(\boldsymbol{\theta})}\right)=\left.\frac{d^2l_i}{d\boldsymbol{\eta}_i d\boldsymbol{\theta}}\right|_{\boldsymbol{\eta}_i=\boldsymbol{\eta}_i^*(\boldsymbol{\theta})}+\left.\frac{d^2l_i}{d\boldsymbol{\eta}_i^2}\right|_{\boldsymbol{\eta}_i=\boldsymbol{\eta}_i^*(\boldsymbol{\theta})}\frac{d\boldsymbol{\eta}_i^*}{d\boldsymbol{\theta}}=\mathbf{0}. \tag{45}$$

By rearranging terms and inverting the matrix, we finally get that

$$\frac{d\boldsymbol{\eta}_i^*}{d\boldsymbol{\theta}}=-\left(\left.\frac{d^2l_i}{d\boldsymbol{\eta}_i^2}\right|_{\boldsymbol{\eta}_i=\boldsymbol{\eta}_i^*(\boldsymbol{\theta})}\right)^{-1}\left.\frac{d^2l_i}{d\boldsymbol{\eta}_i d\boldsymbol{\theta}}\right|_{\boldsymbol{\eta}_i=\boldsymbol{\eta}_i^*(\boldsymbol{\theta})}. \tag{46}$$

The second order derivatives of the individual joint log-likelihoods with respect to the random effect parameters were previously derived in Eq. 13. In contrast to the first order approximation of the Hessian used in the approximate population log-likelihood, the second order derivatives of $\boldsymbol{\epsilon}_{ij}$ and $\mathbf{R}_{ij}$ are kept. These are obtained by differentiating Eqs. 19 and 20 once more with respect to $\boldsymbol{\eta}_i$ (not shown). This in turn requires the second order sensitivity equations of the state variables with respect to $\boldsymbol{\eta}_i$, which were previously provided in Eq. 40. In addition to second order derivatives of the individual joint log-likelihoods with respect to the random effect parameters, Eq. 46 also requires the second order mixed derivatives, which are given by

$$\begin{aligned}\frac{d^2l_i}{d\eta_{ik}d\theta_m}=-\frac{1}{2}\sum_{j=1}^{n_i}\Bigg(&2\frac{d\boldsymbol{\epsilon}_{ij}^T}{d\theta_m}\mathbf{R}_{ij}^{-1}\frac{d\boldsymbol{\epsilon}_{ij}}{d\eta_{ik}}-2\boldsymbol{\epsilon}_{ij}^T\mathbf{R}_{ij}^{-1}\frac{d\mathbf{R}_{ij}}{d\theta_m}\mathbf{R}_{ij}^{-1}\frac{d\boldsymbol{\epsilon}_{ij}}{d\eta_{ik}}\\&+2\boldsymbol{\epsilon}_{ij}^T\mathbf{R}_{ij}^{-1}\frac{d^2\boldsymbol{\epsilon}_{ij}}{d\eta_{ik}d\theta_m}-\boldsymbol{\epsilon}_{ij}^T\mathbf{R}_{ij}^{-1}\frac{d^2\mathbf{R}_{ij}}{d\eta_{ik}d\theta_m}\mathbf{R}_{ij}^{-1}\boldsymbol{\epsilon}_{ij}\\&+2\boldsymbol{\epsilon}_{ij}^T\mathbf{R}_{ij}^{-1}\frac{d\mathbf{R}_{ij}}{d\eta_{ik}}\mathbf{R}_{ij}^{-1}\frac{d\mathbf{R}_{ij}}{d\theta_m}\mathbf{R}_{ij}^{-1}\boldsymbol{\epsilon}_{ij}\\&-2\boldsymbol{\epsilon}_{ij}^T\mathbf{R}_{ij}^{-1}\frac{d\mathbf{R}_{ij}}{d\eta_{ik}}\mathbf{R}_{ij}^{-1}\frac{d\boldsymbol{\epsilon}_{ij}}{d\theta_m}\\&+\operatorname{tr}\left[\mathbf{R}_{ij}^{-1}\frac{d\mathbf{R}_{ij}}{d\theta_m}\mathbf{R}_{ij}^{-1}\frac{d\mathbf{R}_{ij}}{d\eta_{ik}}+\mathbf{R}_{ij}^{-1}\frac{d^2\mathbf{R}_{ij}}{d\eta_{ik}d\theta_m}\right]\Bigg)\\&-\boldsymbol{\eta}_i^T\boldsymbol{\Omega}^{-1}\frac{d\boldsymbol{\Omega}}{d\theta_m}\boldsymbol{\Omega}^{-1}\frac{d\boldsymbol{\eta}_i}{d\eta_{ik}}.\end{aligned} \tag{47}$$

Here, all terms have previously been introduced except $d^2\boldsymbol{\epsilon}_{ij}/d\eta_{ik}d\theta_m$ and $d^2\mathbf{R}_{ij}/d\eta_{ik}d\theta_m$, which are provided within the derivation of Eq. 37 and through a corresponding derivation involving $\mathbf{R}_{ij}$.

Better starting values for optimization of random effect parameters

Computing the approximate population log-likelihood and its gradient with respect to the parameters $\boldsymbol{\theta}$ requires the determination of $\boldsymbol{\eta}_i^*$ for every individual. The first time $\log L_F$ and its gradient are evaluated it is reasonable to initiate the inner level optimizations for $\boldsymbol{\eta}_i^*$ with $\boldsymbol{\eta}_i=\mathbf{0}$. However, in the subsequent steps of the optimization with respect to $\boldsymbol{\theta}$, better starting values for $\boldsymbol{\eta}_i$ can be provided. One way of choosing the starting values $\boldsymbol{\eta}_i^0$ for the optimization of $\boldsymbol{\eta}_i$ is to set them equal to the optimized value from the last step of the outer optimization. If we for simplicity of notation from now on suppress the index of $\boldsymbol{\eta}_i$ denoting the individual, i, and instead let the the index s denote the step of the outer optimization with respect to $\boldsymbol{\theta}$, this can be expressed as $\boldsymbol{\eta}_{s+1}^0=\boldsymbol{\eta}_s^*$. This will be particularly helpful as the optimization converges and the steps in $\boldsymbol{\theta}$ become smaller. Using $\boldsymbol{\eta}^*$ from the evaluation of $\log L_F$ as starting value is also a good strategy when computing the gradient of $\log L_F$ by a finite difference approximation.

If the sensitivity approach is used for computing the gradient of $\log L_F$, even better starting values of $\boldsymbol{\eta}$ can be provided. This is accomplished by exploiting the fact that the sensitivity $d\boldsymbol{\eta}^*/d\boldsymbol{\theta}$ happens to be part of the gradient calculation. By making a first order Taylor expansion of the implicit function $\boldsymbol{\eta}^*(\boldsymbol{\theta})$, we propose the following update of the starting values of the random effect parameters

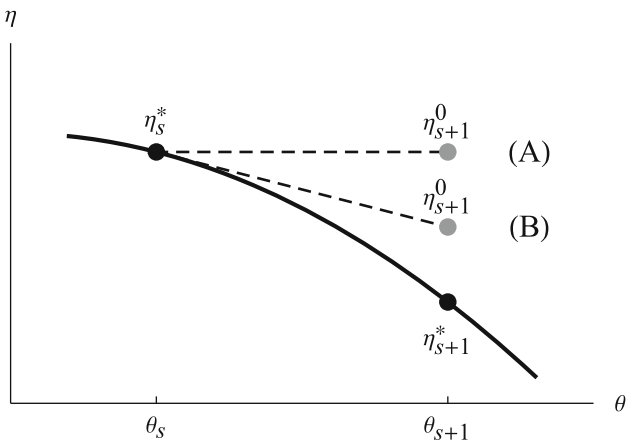

Fig. 1 Starting values for finding optimal random parameter values. The hypothetical relationship between a parameter θ and the optimal value of a random effect parameter η^* is depicted by the *solid curve*, and the optimal values of η for two consecutive θ of the optimization, θ_s and θ_{s+1}, are shown as *black points*. The two approaches for selecting starting values η_{s+1}^0 are shown as *dashed lines* and *gray points*, with the label (*A*) for using the previous value and (*B*) for using the gradient based update

$$\boldsymbol{\eta}_{s+1}^{0} = \boldsymbol{\eta}_{s}^{*} + \frac{d\boldsymbol{\eta}_{s}^{*}}{d\boldsymbol{\theta}}(\boldsymbol{\theta}_{s+1} - \boldsymbol{\theta}_{s}). \quad (48)$$

The two approaches for choosing $\boldsymbol{\eta}_{s+1}^{0}$ are illustrated in Fig. 1.

Results

Based on the theory presented in the previous section, we propose an alternative implementation of the FOCE and FOCEI methods for parameter estimation of NLME models based on differential equations. The steps of this novel approach are outlined in Algorithm 1. The crucial points are the computation of gradients using sensitivity equations, for both the inner and outer problem, and the way that starting values for the inner problem are determined.

Algorithm 1 Parameter estimation algorithm

$s := 0$, $\boldsymbol{\theta}_s := \boldsymbol{\theta}_{starting}$ ▷ Initialize algorithm
for all individuals **do**
  $u := 0$, $\boldsymbol{\eta}_s^u := 0$
end for
repeat ▷ Solve the outer problem
  for all individuals **do**
    $u := 0$
    repeat ▷ Solve the inner problem
      Solve for $\mathbf{x}$ and the sensitivities $d\mathbf{x}/d\boldsymbol{\eta}$
      Compute l and $dl/d\boldsymbol{\eta}$
      Update $\boldsymbol{\eta}_s^{u+1}$ according to BFGS
      $u := u + 1$
    until $\boldsymbol{\eta}_s^*$ is obtained
  end for
  for all individuals **do**
    Set $\boldsymbol{\eta} := \boldsymbol{\eta}_s^*$
    Solve for $\mathbf{x}$ and the sensitivities $d\mathbf{x}/d\boldsymbol{\eta}$, $d\mathbf{x}/d\boldsymbol{\theta}$, $d^2\mathbf{x}/d\boldsymbol{\eta}^2$, and $d^2\mathbf{x}/d\boldsymbol{\eta}d\boldsymbol{\theta}$
  end for
  Compute $\log L_F$ and $d\log L_F/d\boldsymbol{\theta}$
  Update $\boldsymbol{\theta}_{s+1}$ according to BFGS
  for all individuals **do** ▷ Set starting values for inner problem
    $\boldsymbol{\eta}_{s+1}^{0} = \boldsymbol{\eta}_s^* + \frac{d\boldsymbol{\eta}_s^*}{d\boldsymbol{\theta}}(\boldsymbol{\theta}_{s+1} - \boldsymbol{\theta}_s)$
  end for
  $s := s + 1$
until convergence of $\boldsymbol{\theta}$

The algorithm was evaluated using a two-compartment model with a capacity-limited elimination. This is a moderately complex pharmacokinetic model that requires the numerical solution of differential equations. All details regarding the model, including model equations, parameters used for simulating data, the starting values for the parameter estimation, and the parameter estimates, can be found in the "Appendix 3" section. A short summary of the model is shown in Table 1. Briefly, four versions of the model (M1-M4) were used. In model M1, some parameters were fixed to the true values, hence excluded from the estimation. Three random effect parameters were introduced but their covariance matrix was limited to a diagonal matrix. Observations were modeled using a normally distributed additive error. All parameters were estimated in model M2, including the full covariance matrix for the random effect parameters. In model M3, an additional random effect parameter was introduced and the full covariance matrix was extended accordingly. The observational model was also altered to include measurements from both compartments, and the error in the measurements from the first compartments was modeled with both an additive and proportional term. Model M4 is the same as M3 but for this model the parameter estimation was performed with FOCEI instead of FOCE.

Improving gradient precision and accuracy

We compared our proposed method of computing the gradient of the approximate population log-likelihood, $\log L_F$, with respect to $\boldsymbol{\theta}$ to the more straightforward approach of finite difference approximation. Two versions of the finite difference approximations were considered, a forward difference and a central difference. To investigate the precision and accuracy of these approximations, we first determined the estimate of $\boldsymbol{\theta}$ for model M1. We then computed all 6 elements of the gradient at this point in parameter space using different values of the relative step size, 10^{-h}. The details of the comparison are explained in the methods section. In addition, we computed the gradient using the approach based on sensitivity equations. A comparison of the two approaches is shown in Fig. 2,

Table 1 Overview of benchmark models showing the method used, the numbers of different types of parameters, and the total number of ordinary differential equations (ODEs) per individual for the inner and outer problem (including the number of sensitivity equations according to Eqs. 22 and 41)

Model	M1	M2	M3	M4
Method	FOCE	FOCE	FOCE	FOCEI
Total number of fixed effect parameters ($\boldsymbol{\theta}$)	6	12	18	18
Parameters in the ODE model	3	5	5	5
Parameters in the observational model	0	1	3	3
Parameters in the random effect covariance matrix	3	6	10	10
Number of random effect parameters ($\boldsymbol{\eta}$)	3	3	4	4
ODEs per individual, inner problem	8	8	10	10
ODEs per individual, outer problem	44	60	80	80

where each row shows one element of the gradient at two levels of magnification.

The left column of Fig. 2 shows a pattern that appears to be consistent for all parameters; for large h, i.e. small step sizes, the result of the finite difference approach is dominated by numerical noise for both forward and central differences. Thus, for this particular model, and for this particular point in parameter space, the finite difference approximations have low precision as h increases beyond 3. For small h, i.e. large step sizes, there is a trend of severely decreased accuracy for the forward differences. Looking at the values of the gradient from the approach of sensitivity equations, it is clear that for h around 2 and smaller, forward differences produces values of elements of the gradient that are up to two orders of magnitude larger, and with a wrong sign in four of six cases. The behavior of the central difference approximation for small and intermediate h is best viewed in the right column, where the scales of the axis have been chosen differently. For the three first elements of the gradient, namely the derivatives of $\log L_F$ with respect to V_{max}, V_1, and K_m, the central difference approximation appears to be accurate but, on the scale of the size of the gradient computed according to the sensitivity equation approach, the limits in precision are visible. For the derivatives with respect to the the parameters of $\boldsymbol{\Omega}$, ω_{11}, ω_{22}, and ω_{33}, there are obvious issues with both accuracy and precision of the approximation, producing derivatives that are both of wrong size and sign. The fact that the approximation starts to deviate systematically for h less than 2 indicates that in these parameter directions, and on this scale, an expansion of the approximate log-likelihood function has a significant contribution of third order terms and higher, causing a bias in the approximation of the gradient using central differences.

The approach of determining the gradient using sensitivity equations is also subject to numerical errors. By repeated evaluation of the gradient using randomized values for the starting values of the inner optimization problem, we determined the relative standard error. For all 6 parameter directions of the gradients, the relative standard errors were between 0.1 and 1 %. Thus, these numerical errors are so small that they would not even be visible on the scales of Fig. 2.

Improving computational time

We investigated the improved computational times resulting from replacing finite difference approximations of the gradients in the inner and outer problem with gradients computed using sensitivity equations, and from using better starting values for the inner problems. The contribution from each of these three steps, as well as their accumulative effect, are shown in Fig. 3.

For the first step of improvement, using gradients based on sensitivity equations for the inner problem, computational times for models M1 and M2 (with 3 random effect parameters) decreased to almost a third compared to the approximation using forward differences, and to a fifth compared to central differences. The ratio of these two relative decreases is reasonable considering that the forward difference approximation requires 4 function evaluations and the central difference requires 7 evaluations. Model M3 and M4 contain one additional random effect parameter and the gains in speed were slightly larger compared to both variants of the finite difference approximation.

Replacing the finite difference approximation of the gradient in the outer problem with the approach based on sensitivity equations results in further improvement of computational times. As the number of parameters in the outer optimization problem increase from 6 to 18 for the models M1 to M3, the reduction in computational times improves from 29 to 14 % when compared to forward differences, and from 16 to 7 % compared to central differences. Although model M4 is identical to M3, the reduction in computational times are slightly less for this model. This is because M4 uses FOCEI for estimating parameters, which compared to FOCE requires more time

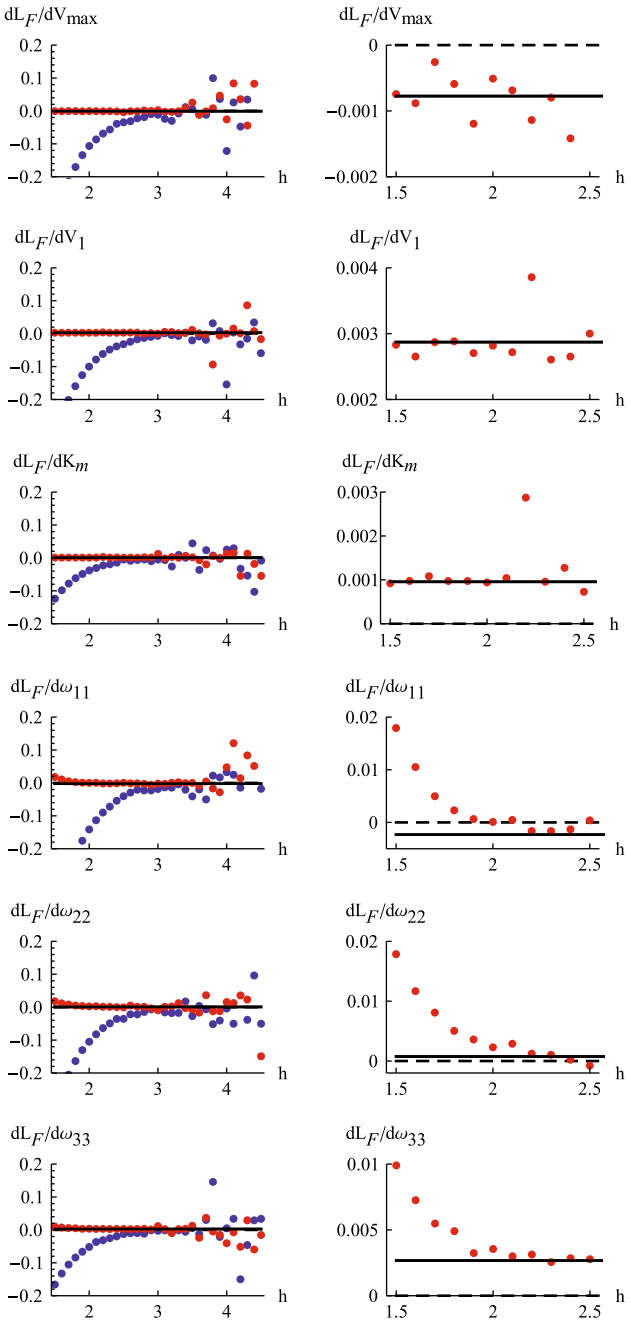

Fig. 2 Precision and accuracy of the approximate population log-likelihood gradient. Each *row* displays one element of the gradient, and the *left* and *right* columns show two different levels of magnification, respectively. Evaluations of the derivatives of $\log L_F$ using forward and central differences with different relative step sizes are shown as *blue* and *red dots*, respectively. A single evaluation of the derivatives using the approach based on sensitivity equations is indicated by a *black line*, and the value zero is shown as a *dashed line* for comparison

for putting together the more complex gradient expressions once the sensitivity equations have been solved. Again it is reasonable to expect a nearly doubled factor of decrease when comparing central and forward finite differences since the former need almost twice as many function evaluations.

The final step of improvement is only applicable when gradients for both the inner and outer problem are computed using the approach based on sensitivity equations. Thus, the distinction between forward and central differences is no longer of importance. The decrease in computational times were around 70 % for models M1 to M3, and somewhat less for model M4, which again benefits less due to its larger overhead of having to compute all interaction terms.

The accumulated effect of all the steps range from a decrease in computational times to 7 % for the least complex model when comparing to forward differences, to the substantial decrease to 1 % for the most complex model when comparing to central differences.

Discussion

This article has demonstrated a novel approach to the computation of gradients needed for the FOCE and FOCEI approximation of the population likelihood encountered in NLME modeling. We have derived the analytic expressions for the gradients of both the individual and population log-likelihoods as well as the so called sensitivity equations, whose solution is a necessity for evaluating the gradient expressions.

Using sensitivity equations to compute the gradient for the inner problem is quite straightforward. As we understand it, approaches along these lines are in fact used for the inner problem, at least to some extent, in softwares such as NONMEM and Phoenix NLME. For the approximate population log-likelihood on the other hand, the sensitivity approach to gradient computation is complicated by the fact that this function depends on the nested optimization of the individual joint log-likelihoods. In this work we have, to the best of our knowledge, for the first time demonstrated how sensitivity equations can be used for computing the gradient of the FOCE and FOCEI approximations to the population log-likelihood. A key step to obtain this gradient involves the derivative of the optimal random effect parameters with respect to the fixed effect parameters. It was shown that this derivative could be determined given second order sensitivity equations.

Abandoning the finite difference approximation of gradients in favor of the approach of sensitivity equations were shown to have two advantages; gradients could be computed with a higher precision and computational times were substantially reduced. Though, implementation of the presented method is more challenging compared to finite difference FOCE/FOCEI, and the limitations of the Laplacian approximation are still present.

Increased precision and accuracy of gradients

The optimization of the approximate population log-likelihood $\log L_F$ with respect to $\boldsymbol{\theta}$ would typically be performed with a quasi-Newton method. A straightforward approach to obtaining the gradient needed for such methods is to compute it from a finite difference approximation. However, the finite difference approach may result in issues with both precision and accuracy of the gradient. We demonstrated this for the computation of the gradient in the outer problem, evaluated close to the optimum of $\log L_F$. Although the use of central differences with an appropriate step length could avoid the worst problems, precision and accuracy were still inferior compared to the approach based on sensitivity equations. The potential limitations of combining NLME models based on differential equations with likelihood optimization using gradients computed by finite differences have previously been recognized [3]. The issues with the finite difference approximation depend both on numerical limitations and on the approximation itself. First of all, evaluation of $\log L_F$ can only be done to a certain precision. This is especially evident for models based on differential equations, whose solution involves adaptive schemes for numerical integration. In addition to the numerical precision of functions like log, which is high, the precision of $\log L_F$ depends on the precision of the solutions to the differential equations, and the precision of computing derivatives with respect to $\boldsymbol{\eta}$. The precision of $\log L_F$ also has a strong dependence on the precision of $\boldsymbol{\eta}^*$, which in turn again depends on the solutions of differential equations and, if the inner level optimization problem is performed using a gradient-based method, depends on computing derivatives of the individual joint log-likelihoods with respect to $\boldsymbol{\eta}$. Secondly, taking finite differences of $\log L_F$ will amplify numerical errors, resulting in increasingly poor precision of the gradient as the step size is decreased. On the other hand, taking too long steps will decrease the accuracy of the approximation due to the increasing impact of higher order terms in an expansion of $\log L_F$ (forward differences is only exact up to first order terms, and central differences is only exact up to second order terms). Even if it for a given model in some cases would be possible to customize the step length for the finite difference approximation (which typically would be different in each separate parameter direction) using an analysis like the one performed here, it would be infeasible in practice since such an investigation may take longer time than solving the parameter estimation problem itself. Adding further to the problem, the choice of a suitable step size will most certain be different depending on the point in parameter space, thus constantly requiring a reevaluation of the step size.

There are several advantages of being able to compute gradients with an improved precision and accuracy (i) Parameter estimates can be computed with higher precision, or alternatively, the same precision can be obtained but with shorter run times since we may afford to reduce the precision of the inner problem while still maintaining a similar precision in the outer problem [11]. (ii) Premature termination and convergence problems of the parameter estimation algorithm can be avoided or at least reduced [8, 24]. (iii) May enable the calculation of standard errors of the parameter estimates in cases where this was not possible due to the numerical issues of the finite difference approach [7]. However, we want to point out that for many points in the parameter space the limited precision and accuracy of the finite difference approach may not be crucial for the progression of the optimization as long as the approximation of the gradient results in a true ascent direction of the function being maximized.

Decreased computational times

The relative decrease in computational times were investigated for the successive application of three specific steps toward improvement, namely (i) Gradients based on sensitivity equations in the inner problem, (ii) Gradients based on sensitivity equations in the outer problem, and (iii) Better starting values for the inner problem. In all cases of applying the two first steps, we found that the decrease in computational times were substantially larger when comparing to central differences instead of forward differences. This was anticipated since central differences requires almost twice as many function evaluations as forward differences. Moreover, for both the inner and outer levels of optimization, the gains in computational times tended to be larger for models with higher number of parameters. For instance, the run time improvements of providing gradients from sensitivity equations in the outer problem were more than doubled for model M3 with 18 parameters compared to model M1 with 6 parameters. It was also observed that the improvement factor in the outer optimization was slightly lower for FOCEI compared to FOCE. Although the number of ODEs to be solved in both the inner and outer problem is the same, this was expected considering that the FOCEI method is based on more extensive expressions for both the likelihood and its gradient.

There are two main reasons why the approaches based on sensitivity equations should be faster. First of all, the right hand side of the sensitivity equations has lots of common subexpressions both with other sensitivity equations and with the original system of differential equations. Thus, the cost of evaluating the right hand side for the combined system of the original differential equations and the sensitivity equations can be surprisingly small. Furthermore, since the sensitivity equations are linear in the sensitivity state variables, there is typically little extra

effort needed in the adaptive time stepping of the differential equations solver for accommodating these additional equations. For the inner problem this means that it is faster to solve the combined system, yielding in total $n(1+q)$ differential equations, rather than having to solve the n original differential equations $1+q$ times, which would have been the case using forward finite differences. Secondly, the use of sensitivity equations in the outer level optimization avoids the repeated need of having to solve the inner problems for perturbed values of the outer parameters. The exact improvement made at this step depends on several factors of which perhaps the most important one is the desired precision (and hence the number of iterations required) of the inner optimizations needed for every parameter perturbation of a finite difference approximation (had this alternative been used instead).

We furthermore note that the computation of gradients based on sensitivity equations is highly amenable to parallelization, something which may be exploited to speed up computations considerably. The potential gains of doing this are expected to be similar to those of parallelizing the computation of the population log-likelihood itself [11].

In addition to the reduced computational times coming from the two steps of improved gradient computations, a third level of speed up was obtained by choosing more informed starting values for the inner problem. Although this improvement was not as substantial as the others, the gains from this step may be quite dependent on the starting values of the outer optimization problem. As the outer level optimization converges, the steps in $\boldsymbol{\theta}$ become successively smaller, which in turn means that the linear approximation of $\boldsymbol{\eta}^*(\boldsymbol{\theta})$ becomes better. Thus, the overall improvement in computational time will depend on how much of the optimization that was spent in these "later stages" of convergence. This means that it is likely that the relative improvement will be larger if the optimization had been started closer to the optimum.

Setting the results of Fig. 3 in relation to commercial softwares for NLME parameter estimation, we would like to comment on a mixed analytical/finite difference approach to the differentiation of the FOCE likelihood with respect to the parameters of the random effect covariance matrix $\boldsymbol{\Omega}$, which is used as default by NONMEM (when the SLOW option is not selected). Since these parameters do not normally directly influence neither the residuals, nor the residual covariance matrix, their part of the likelihood gradient is less complicated compared to other parameters. As shown by the theory in this paper, their part of the gradient may be computed using only second order $\boldsymbol{\eta}$ sensitivities (Eq. 40), not requiring first order $\boldsymbol{\theta}$ or second order mixed sensitivities (Eqs. 38 and 39, respectively). Although NONMEM FOCE does not use second order $\boldsymbol{\eta}$ sensitivities, it still utilizes this technique by performing a central finite difference evaluation on the first order $\boldsymbol{\eta}$ sensitivities. While this is slower than performing completely analytical second derivatives, along with some erosion of precision, it is certainly faster than the SLOW FOCE method, which must perform the inner problem re-optimizations at each outer level perturbation of the $\boldsymbol{\Omega}$-parameters. The derivatives of the likelihood with respect to the remaining parameters are still obtained from finite differences.

The degree of improvement of speed for the S-S approach compared to an approach that is mixing finite differences and analytical methods at the outer level, i.e, an S-F/S approach, may therefore be less substantial than what can be achieved for going from S-F to S-S. Under the realistic assumption that all perturbed evaluations of $\log L_F$ are equally costly, and further assuming that the $\boldsymbol{\Omega}$-part of the gradient can be obtained at a computationally insignificant cost (ignoring the relatively few extra evaluations needed for the central finite difference of the first order $\boldsymbol{\eta}$ sensitivities), the reference time of 100 % for going from forward differences S-F to S-S in Fig. 3 would change to $((1+P_\theta-P_\Omega)/(1+P_\theta))100$ % if instead going from S-F/S to S-S, where P_θ is the total number of parameters and P_Ω is the number of $\boldsymbol{\Omega}$-parameters. The reference time for going from central differences S-F to S-S would for S-F/S to S-S similarly change to $((1+2P_\theta-2P_\Omega)/(1+2P_\theta))100$ %. For model M1 this would mean that the improvements to 29 and 16, for forward and central differences, respectively, should be compared to the S-F/S references of 57 and 54, rather than to 100, and for model M3 the improvements to 14 and 7 should be compared to 47 and 46. In general, one would expect the advantage of the S-S approach to decrease as the fraction of $\boldsymbol{\Omega}$-parameters with respect to the total number of parameters increases, e.g., for problems with many random effect parameters when estimating the full random effect covariance matrix. It must however be emphasized that this is a mixed analytical/finite difference approach, and may as such have lower precision and accuracy compared to the S-S approach. Moreover, the remaining part of the gradient will still be completely derived from finite differences, and is expected to have the same comparable quality to the S-S approach as demonstrated in the results section.

Extending the line of thought, one could also consider a hybrid between the above S-F/S approach and the S-S approach, where the derivatives of $\log L_F$ with respect to the $\boldsymbol{\Omega}$-parameters are computed according to the exact approach presented in this work but where the derivatives for the remaining parameters of the outer level problem are obtained from a finite difference approach. This would indeed require the second order sensitivity

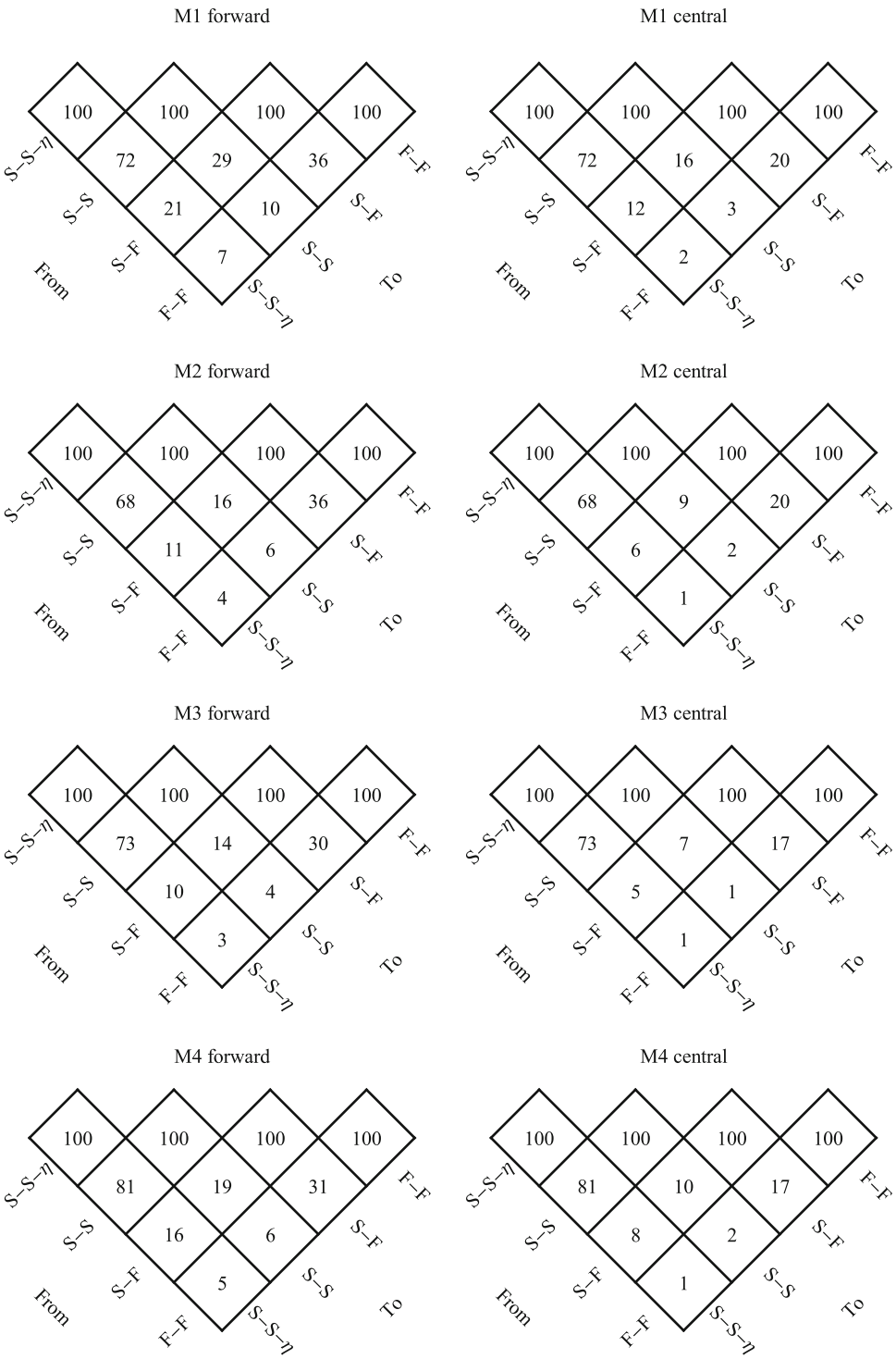

Fig. 3 Comparison of relative estimation times. The relative computation times expressed in percentage are shown for going from one scheme for obtaining gradients to another. Results are shown for the model variants M1-M4, using either a forward or central implementation of the finite difference approach. F-F denotes the use of finite differences for both the inner and outer problem, S-F the use of gradients based on sensitivity equations for the inner problem, S-S the use of gradients based on sensitivity equations for both inner and outer problems, and S-S-η denotes the additional implementation of the better starting values for the inner problem

equations with respect to $\boldsymbol{\eta}$, but not the first order $\boldsymbol{\theta}$ or the mixed second order sensitivity equations. The accuracy and precision would still be lower for the part of the gradient obtained from finite differences but the elements corresponding to the parameters of $\boldsymbol{\Omega}$ would be of the same quality as the S-S approach, i.e., without approximations.

Challenges and limitations

Moving from a convenient proof-of-concept environment such as Mathematica, in which the proposed method currently is implemented, to a more stand-alone environment of a commercial software may present various challenges. One of the most obvious challenges is the integration of

functionality for performing symbolic differentiation. This is essential since the sensitivity equations, i.e., the differential equations in Eqs. 21, 38, 39, and 40, are model specific and have to be derived for every new model, in order to apply the results of this paper. It also applies to the derivatives of $\mathbf{h}$, $\mathbf{R}_{ij}$, and $\boldsymbol{\Omega}$, which too are model specific. Since differential equation models may be quite complex, and because second order derivatives are needed, it is not realistic to perform these derivations manually, and a tool that can perform symbolic differentiation will be required. To this end, one may consider to look at free symbolic packages such a SymPy [23]. The use of tools for symbolic analysis may furthermore be crucial to exploit the existence of common subexpressions, e.g., in the right hand sides of the sensitivity equations.

An alternative approach, which does not require symbolic differentiation, would be to use so called automatic differentiation (AD) [19]. The idea of AD is that every mathematical function that can be written as a computer program can be differentiated by applying the chain rule of differentiation, leading to the differentiation of every elementary operation of that computer program. Even though AD in principle could be applied directly to the approximate population likelihood, whose gradient we wish to compute, this would in practice be infeasible as this function is based on the execution of both optimization routines and adaptive numerical integration of differential equations. If used, AD would therefore not be applied to the population likelihood, but to the right hand sides of the model differential equations, and to the other model objects requiring differentiation. The parameter estimation would thus still proceed according to the steps laid out in Algorithm 1, but with symbolic differentiation replaced with AD. Following such an approach, the precision and accuracy of the gradients are not expected to differ, but it would have to be investigated how AD performs in terms of computational times. With a so called reverse mode AD it may actually be possible to improve run times even further compared to the current results.

Even if tools for differentiation can be provided for a stand-alone implementation, estimation methods which involve the direct differentiation of model state variables, etc., may experience limitations when considering other types of mathematical formalisms, such as models based on stochastic differential equations or hidden Markov models, since the required derivatives may be challenging to obtain. The method of computing gradients based on finite differences, on the other hand, do not care about the details of how a model is evaluated and has no limitations in this sense.

Finally, it should also be mentioned that although the approach for gradient computations presented here may improve the performance of FOCE and FOCEI, the fundamental limitations of the Laplacian approximation as such still remains. Being only an approximation to the population likelihood, this class of methods do not guarantee the desirable statistical properties of a true maximum likelihood estimate. In this respect the new generation of estimation methods which are based on Monte Carlo expectation maximization methods, such as stochastic approximation expectation maximization and importance sampling, are superior to the classical ones since the parameter estimates and their confidence intervals, etc., are not biased by likelihood approximations. However, FOCE and FOCEI will likely be important complementary methods for a long time still, and improving their efficiency is therefore nonetheless relevant.

Possible extensions

The approach of computing gradients using sensitivity equations presented here could be modified for other variants of the population likelihood based on the Laplacian approximation. For instance, with some alterations it could be applied to the first order (FO) approximation of the population likelihood. Since the FO method does not rely on conditioning with respect to the optimal random effect parameters, the use of an approach based on sensitivity equations would be less complicated but at the same time also less rewarding. Gradients based on the approach of sensitivity equations could with some adjustments also be derived for the Laplace method. This would however require third order sensitivity equations but may be worthwhile since the potential gains should be at least as substantial as for FOCE and FOCEI. Because the theory presented in this article is derived for the FOCEI approximation, it accounts for the dependence of residual errors on the random effect parameters. This means that the gradient expressions stated here are suitable for prediction error-type NLME models, including models based on stochastic differential equations (see for instance [6, 14, 18]), since these typically display an interaction between residuals and random effects. The first step towards this end has in fact already been taken through the successful application of sensitivity equations for computing gradients in stochastic differential equation models on the single-subject level [16]. Furthermore, gradient computations based on sensitivity equations may be useful for the problem of optimal experimental design [1, 17].

Conclusions

The presented approach of computing gradients for both the individual- and population-level log-likelihoods of the FOCE and FOCEI approximations leads to more robust gradients and decreased computational times. We therefore

suggest that future implementations of these conditional estimation methods should include the approach based on sensitivity equations for computing the gradients. We eagerly await the further development of the proposed approach from the prototyped version used in the present study to its implementation in publicly or commercially available software packages.

Methods

The NLME parameter estimation algorithm investigated in this study was implemented in Mathematica 9. An executable version of the code, and the data sets used within this study, may be received from the authors upon request.

Comparison of performance

The performance of a computer program for parameter estimation in NLME models depends on several factors, such as the particular NLME model, the experimental data, how the estimation problem is formulated and possibly approximated, the choice and settings of the optimization method (including sub-methods such as line-searches, etc.), starting values of parameters, the differential equation solver used, the design of convergence criteria, etc. This paper is investigating the advantages of providing gradients by means of sensitivity equations for the FOCE or FOCEI approximation of the population likelihood. However, this paper is not claiming to address all the other factors that will impact on the parameter estimation. Comparing measures such as absolute run-times of our implementation with commercial software like NONMEM may therefore be misleading with respect to the advantages of gradient calculations. To avoid this the comparison is designed to look only at the improvements made by abandoning the finite difference approximation in our own implementation.

Comparison of precision and accuracy

The comparison of precision and accuracy was performed in the following way. At the optimal values of $\boldsymbol{\theta}$ (found from the comparison of computational times), the elements of the gradient of the approximate log-likelihood function were approximated with finite differences, using a relative step size, according either to a forward difference

$$\frac{\log L_F\left(\theta_m(1+10^{-h})\right)-\log L_F(\theta_m)}{\theta_m 10^{-h}}, \tag{49}$$

or a central difference,

$$\frac{\log L_F\left(\theta_m(1+10^{-h})\right)-\log L_F\left(\theta_m(1-10^{-h})\right)}{2\theta_m 10^{-h}}. \tag{50}$$

For these function evaluations, the inner problem was solved to a precision of 4 digits (using the gradients from the approach of sensitivity equations). Furthermore, for forward differences the value of $\log L_F$ was recalculated for every h using randomized starting values for the inner problems. This was done to avoid correlations between differences with different step size that may otherwise have resulted from a single realization of the numerical error of $\log L_F$.

The approach of determining gradients using sensitivity equations does not involve any approximations, and is therefor expected to be correct on average. Its precision was assessed by computing the gradient 500 times using randomized starting values for the inner problems. For these gradient evaluations, the inner problem was solved to a precision of 4 digits.

Comparison of computational times

The comparison of computational times was done in the following way. Both the inner and outer problem were solved using gradients based on sensitivity equations, as outlined in the theory section. The inner problem was solved to a precision of 4 digits, and the outer to a precision of 3 digits. The comparison to finite differences was done by simultaneously clocking the time of computing gradients by a finite difference approximation but proceeding with the optimizations according to values of the gradient from the sensitivity approach. The reason for doing this is that the number of iterations, and the properties of every iteration (such as stiffness of the model equation with that certain set of parameters), for solving both the inner and outer problem might be affected by the choice of method for computing the gradients. Even small numerical differences in the results of the two methods may cause the paths taken in the parameter space to diverge substantially over the course of the optimizations, potentially making the comparison unfair. In this way we isolate the comparison to the actual computational times for the different methods of obtaining the gradients. Since the methods based on sensitivity equations were shown to have a higher precision in the evaluation of gradients, there may be additional gains in computational times to be made from traversing the parameter space based on more exact gradients. However, quantifying this type of contribution may require averaging over a large number of models and parameter starting values and was not considered. Thus, our implementation of the comparison focuses on the direct improvements in computational times and will therefore be a conservative measure of the gains in speed.

To make a fair implementation of timing the finite differences approach the following starting values of the random effect parameters for the inner problem were used.

When evaluating the approximate population log-likelihood at the unperturbed parameter values of the outer problem, the starting values for the parameters of the inner problem were set to the optimum from the previous outer evaluation, i.e., according to approach A in Fig. 1. For evaluating the approximate population log-likelihood at the perturbed parameter values of the outer problem, the starting values for the parameters of the inner problem were set to the optimum obtained for the unperturbed outer problem parameters. The relative size of each perturbation of the parameters in $\boldsymbol{\theta}$ was 10^{-2}.

Compared to the finite difference approaches, using sensitivity equations had an overhead of evaluating the quite substantial mathematical expressions for the gradients once the differential equations are integrated, something which was carefully included in the comparison of computational times.

Optimization algorithm

Both the inner and outer optimization problems were solved using the BFGS method [20].

Derivation of sensitivity equations

Given an NLME differential equation model, the corresponding sensitivity equations were derived by symbolic differentiation in Mathematica.

Acknowledgments We thank the two anonymous reviewers for their constructive comments which significantly contributed to improving the quality of this publication. This project has in part been supported by the Swedish Foundation for Strategic Research, which is gratefully acknowledged.

Appendix 1: Matrix calculus

The default representation of a vector is a column vector,

$$\mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{pmatrix}. \tag{51}$$

The derivatives of vectors and matrices by scalars are defined as element-wise derivatives, according to

$$\frac{d\mathbf{y}}{dx} = \begin{pmatrix} \frac{dy_1}{dx} \\ \frac{dy_2}{dx} \\ \vdots \\ \frac{dy_m}{dx} \end{pmatrix}, \tag{52}$$

and

$$\frac{d\mathbf{A}}{dx} = \begin{pmatrix} \frac{da_{11}}{dx} & \frac{da_{12}}{dx} & \cdots & \frac{da_{1n}}{dx} \\ \frac{da_{21}}{dx} & \frac{da_{22}}{dx} & \cdots & \frac{da_{2n}}{dx} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{da_{m1}}{dx} & \frac{da_{m2}}{dx} & \cdots & \frac{da_{mn}}{dx} \end{pmatrix}, \tag{53}$$

respectively. The derivative of scalar by vector is given by

$$\frac{dy}{d\mathbf{x}} = \begin{pmatrix} \frac{dy}{dx_1} & \frac{dy}{dx_2} & \cdots & \frac{dy}{dx_m} \end{pmatrix}, \tag{54}$$

the derivative of vector by vector is given by

$$\frac{d\mathbf{y}}{d\mathbf{x}} = \begin{pmatrix} \frac{dy_1}{dx_1} & \frac{dy_1}{dx_2} & \cdots & \frac{dy_1}{dx_n} \\ \frac{dy_2}{dx_1} & \frac{dy_2}{dx_2} & \cdots & \frac{dy_2}{dx_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{dy_m}{dx_1} & \frac{dy_m}{dx_2} & \cdots & \frac{dy_m}{dx_n} \end{pmatrix}, \tag{55}$$

and the derivative of row-vector by vector is given by

$$\frac{d\mathbf{y}^T}{d\mathbf{x}} = \begin{pmatrix} \frac{dy_1}{dx_1} & \frac{dy_2}{dx_1} & \cdots & \frac{dy_m}{dx_1} \\ \frac{dy_1}{dx_2} & \frac{dy_2}{dx_2} & \cdots & \frac{dy_m}{dx_2} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{dy_1}{dx_m} & \frac{dy_2}{dx_m} & \cdots & \frac{dy_m}{dx_n} \end{pmatrix}. \tag{56}$$

The derivative of a quadratic form is obtained in the following way. Let $y = \mathbf{b}^T\mathbf{A}\mathbf{b}$, where $\mathbf{A}$ is a square matrix and $\mathbf{b}$ a suitable vector. If $\mathbf{A}$ is symmetric then

$$\begin{aligned} \frac{dy}{dx} &= \frac{d\mathbf{b}^T}{dx}\mathbf{A}\mathbf{b} + \mathbf{b}^T\frac{d\mathbf{A}}{dx}\mathbf{b} + \mathbf{b}^T\mathbf{A}\frac{d\mathbf{b}}{dx} \\ &= \mathbf{b}^T\mathbf{A}^T\frac{d\mathbf{b}}{dx} + \mathbf{b}^T\frac{d\mathbf{A}}{dx}\mathbf{b} + \mathbf{b}^T\mathbf{A}\frac{d\mathbf{b}}{dx} \\ &= 2\mathbf{b}^T\mathbf{A}\frac{d\mathbf{b}}{dx} + \mathbf{b}^T\frac{d\mathbf{A}}{dx}\mathbf{b}. \end{aligned} \tag{57}$$

The derivative of an inverse matrix is found by noting that

$$\frac{d\mathbf{A}^{-1}}{dx} = \frac{d\left(\mathbf{A}^{-1}\mathbf{A}\mathbf{A}^{-1}\right)}{dx} = \frac{d\mathbf{A}^{-1}}{dx}\mathbf{A}\mathbf{A}^{-1} + \mathbf{A}^{-1}\frac{d\mathbf{A}}{dx}\mathbf{A}^{-1} + \mathbf{A}\mathbf{A}^{-1}\frac{d\mathbf{A}^{-1}}{dx}, \tag{58}$$

and thus that

$$\frac{d\mathbf{A}^{-1}}{dx} = -\mathbf{A}^{-1}\frac{d\mathbf{A}}{dx}\mathbf{A}^{-1}. \tag{59}$$

The derivative of the logarithm of the determinant of a covariance matrix is given by the following expression. If $\mathbf{A}$ is a real-valued, symmetric, positive-definite matrix, then

$$\frac{d}{dx}\log|\mathbf{A}| = \mathrm{tr}\left[\mathbf{A}^{-1}\frac{d\mathbf{A}}{dx}\right]. \tag{60}$$

This can be seen by first writing $\mathbf{A}$ as $\mathbf{A} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^{-1}$, where $\mathbf{\Lambda}$ is a diagonal matrix. Now, the left-hand side of Eq. 60 becomes

$$\frac{d}{dx}\log|\mathbf{A}| = \frac{d}{dx}\log\left(|\mathbf{Q}|\cdot|\mathbf{\Lambda}|\cdot|\mathbf{Q}^{-1}|\right) = \frac{d}{dx}\log|\mathbf{\Lambda}| = \frac{d}{dx}\sum_i \log\mathbf{\Lambda}_{ii} = \sum_i \frac{1}{\mathbf{\Lambda}_{ii}}\frac{d\mathbf{\Lambda}_{ii}}{dx} = \mathrm{tr}\left[\mathbf{\Lambda}^{-1}\frac{d\mathbf{\Lambda}}{dx}\right], \tag{61}$$

which is equal to the right-hand side of Eq. 60 since

$$\begin{aligned}\mathrm{tr}\left[\mathbf{A}^{-1}\frac{d\mathbf{A}}{dx}\right] &= \mathrm{tr}\left[\mathbf{Q}\mathbf{\Lambda}^{-1}\mathbf{Q}^{-1}\frac{d\mathbf{Q}}{dx}\mathbf{\Lambda}\mathbf{Q}^{-1}\right] \\ &\quad + \mathrm{tr}\left[\mathbf{Q}\mathbf{\Lambda}^{-1}\frac{d\mathbf{\Lambda}}{dx}\mathbf{Q}^{-1}\right] - \mathrm{tr}\left[\frac{d\mathbf{Q}}{dx}\mathbf{Q}^{-1}\right] \\ &= \mathrm{tr}\left[\frac{d\mathbf{Q}}{dx}\mathbf{Q}^{-1}\right] + \mathrm{tr}\left[\mathbf{\Lambda}^{-1}\frac{d\mathbf{\Lambda}}{dx}\right] \\ &\quad - \mathrm{tr}\left[\frac{d\mathbf{Q}}{dx}\mathbf{Q}^{-1}\right] = \mathrm{tr}\left[\mathbf{\Lambda}^{-1}\frac{d\mathbf{\Lambda}}{dx}\right].\end{aligned} \tag{62}$$

Appendix 2: Hessian approximation

For an appropriate model, it holds that

$$\mathbf{E}[\boldsymbol{\epsilon}_{ij}] = \boldsymbol{0}, \tag{63}$$

and

$$\mathbf{E}\left[\boldsymbol{\epsilon}_{ij}\boldsymbol{\epsilon}_{ij}^T\right] = \mathbf{R}_{ij}, \tag{64}$$

where the expected values are taken with respect to data, which here are considered to be random variables whose values have not yet been realized. Based on these equations, the Hessian in Eq. 13 can be simplified to various degrees by approximating its different terms with their expected values. A minimal simplification for eliminating the second order derivative terms is achieved by noting that

$$\mathbf{E}\left[2\boldsymbol{\epsilon}_{ij}^T\mathbf{R}_{ij}^{-1}\frac{d^2\boldsymbol{\epsilon}_{ij}}{d\eta_{ik}d\eta_{il}}\right] = \mathbf{E}\left[2\boldsymbol{\epsilon}_{ij}^T\right]\mathbf{R}_{ij}^{-1}\frac{d^2\boldsymbol{\epsilon}_{ij}}{d\eta_{ik}d\eta_{il}} = 0, \tag{65}$$

and

$$\begin{aligned}&\mathbf{E}\left[-\boldsymbol{\epsilon}_{ij}^T\mathbf{R}_{ij}^{-1}\frac{d^2\mathbf{R}_{ij}}{d\eta_{ik}d\eta_{il}}\mathbf{R}_{ij}^{-1}\boldsymbol{\epsilon}_{ij} + \mathrm{tr}\left[\mathbf{R}_{ij}^{-1}\frac{d^2\mathbf{R}_{ij}}{d\eta_{ik}d\eta_{il}}\right]\right] \\ &= \mathbf{E}\left[-\mathrm{tr}\left[\boldsymbol{\epsilon}_{ij}^T\mathbf{R}_{ij}^{-1}\frac{d^2\mathbf{R}_{ij}}{d\eta_{ik}d\eta_{il}}\mathbf{R}_{ij}^{-1}\boldsymbol{\epsilon}_{ij}\right]\right] + \mathrm{tr}\left[\mathbf{R}_{ij}^{-1}\frac{d^2\mathbf{R}_{ij}}{d\eta_{ik}d\eta_{il}}\right] \\ &= \mathbf{E}\left[-\mathrm{tr}\left[\mathbf{R}_{ij}^{-1}\frac{d^2\mathbf{R}_{ij}}{d\eta_{ik}d\eta_{il}}\mathbf{R}_{ij}^{-1}\boldsymbol{\epsilon}_{ij}\boldsymbol{\epsilon}_{ij}^T\right]\right] + \mathrm{tr}\left[\mathbf{R}_{ij}^{-1}\frac{d^2\mathbf{R}_{ij}}{d\eta_{ik}d\eta_{il}}\right] \\ &= -\mathrm{tr}\left[\mathbf{R}_{ij}^{-1}\frac{d^2\mathbf{R}_{ij}}{d\eta_{ik}d\eta_{il}}\right] + \mathrm{tr}\left[\mathbf{R}_{ij}^{-1}\frac{d^2\mathbf{R}_{ij}}{d\eta_{ik}d\eta_{il}}\right] = 0,\end{aligned} \tag{66}$$

where we are making use of the fact that the trace of a scalar is just the scalar, the order of the expectation and trace operators can be shifted, and the cyclic property of the trace operator. This simplification is used in the present study.

Further simplifications of Eq. 13 may be performed by noting that the expectation of additional terms vanishes,

$$\mathbf{E}\left[-2\boldsymbol{\epsilon}_{ij}^T\mathbf{R}_{ij}^{-1}\frac{d\mathbf{R}_{ij}}{d\eta_{il}}\mathbf{R}_{ij}^{-1}\frac{d\boldsymbol{\epsilon}_{ij}}{d\eta_{ik}}\right] = 0, \tag{67}$$

$$\mathbf{E}\left[-2\boldsymbol{\epsilon}_{ij}^T\mathbf{R}_{ij}^{-1}\frac{d\mathbf{R}_{ij}}{d\eta_{ik}}\mathbf{R}_{ij}^{-1}\frac{d\boldsymbol{\epsilon}_{ij}}{d\eta_{il}}\right] = 0, \tag{68}$$

and by taking the expected value and collecting terms,

$$\begin{aligned}&\mathbf{E}\left[2\boldsymbol{\epsilon}_{ij}^T\mathbf{R}_{ij}^{-1}\frac{d\mathbf{R}_{ij}}{d\eta_{ik}}\mathbf{R}_{ij}^{-1}\frac{d\mathbf{R}_{ij}}{d\eta_{il}}\mathbf{R}_{ij}^{-1}\boldsymbol{\epsilon}_{ij} - \mathrm{tr}\left[\mathbf{R}_{ij}^{-1}\frac{d\mathbf{R}_{ij}}{d\eta_{il}}\mathbf{R}_{ij}^{-1}\frac{d\mathbf{R}_{ij}}{d\eta_{ik}}\right]\right] \\ &= \mathrm{tr}\left[\mathbf{R}_{ij}^{-1}\frac{d\mathbf{R}_{ij}}{d\eta_{il}}\mathbf{R}_{ij}^{-1}\frac{d\mathbf{R}_{ij}}{d\eta_{ik}}\right].\end{aligned} \tag{69}$$

Taken together, all simplifications yield the following Hessian

$$\tilde{\mathrm{H}}_{ikl} = -\sum_{j=1}^{n_i}\left(\frac{d\boldsymbol{\epsilon}_{ij}^T}{d\eta_{il}}\mathbf{R}_{ij}^{-1}\frac{d\boldsymbol{\epsilon}_{ij}}{d\eta_{ik}} + \frac{1}{2}\mathrm{tr}\left[\mathbf{R}_{ij}^{-1}\frac{d\mathbf{R}_{ij}}{d\eta_{il}}\mathbf{R}_{ij}^{-1}\frac{d\mathbf{R}_{ij}}{d\eta_{ik}}\right]\right) - \Omega_{kl}^{-1}, \tag{70}$$

which is the variant used in NONMEM [2].

Appendix 3: Benchmark models and data

The equations for the two-compartment pharmacokinetic model are

Table 2 Parameter values used for simulating data (D), starting values for estimation (S), and parameter estimates (E) for the different models

Parameter	D	S, M1	S, M2	S, M3/M4	E, M1	E, M2	E, M3	E, M4
V_{max}	0.5	0.2	0.2	0.2	0.424	0.419	0.473	0.473
K_m	4	3	3	3	3.91	2.53	4.37	4.37
Cl_d	0.01	–	0.01	0.01	–	0.00976	0.00813	0.00813
V_1	0.3	0.1	0.1	0.1	0.288	0.285	0.321	0.321
V_2	0.1	–	0.1	0.1	–	0.0956	0.0959	0.0959
r_{a1}	$\sqrt{0.5} \approx 0.707$	–	$\sqrt{0.1} \approx 0.316$	$\sqrt{0.1} \approx 0.316$	–	0.414	0.644	0.644
r_{p1}	0^*	$-^*$	$-^*$	$\sqrt{0.1} \approx 0.316$	$-^*$	$-^*$	0.00165	0.00163
r_{a2}	$\sqrt{0.5}^* \approx 0.707$	$-^*$	$-^*$	$\sqrt{0.1} \approx 0.316$	$-^*$	$-^*$	0.730	0.730
ω_{11}	$\sqrt{0.5} \approx 0.707$	1	1	1	0.616	0.553	0.559	0.560
ω_{12}	0	–	0	0	–	−0.0518	−0.123	−0.123
ω_{13}	0	–	0	0	–	0.439	−0.138	−0.138
ω^*_{14}	0^*	$-^*$	$-^*$	0	$-^*$	$-^*$	0.0273	0.0275
ω_{22}	$\sqrt{0.5} \approx 0.707$	1	1	1	0.772	0.575	0.533	0.533
ω_{23}	0	–	0	0	–	−0.485	0.0174	0.0174
ω^*_{24}	0^*	$-^*$	$-^*$	0	$-^*$	$-^*$	−0.0230	−0.0230
ω_{33}	$\sqrt{0.5} \approx 0.707$	1	1	1	0.994	1.39	0.776	0.776
ω^*_{34}	0^*	$-^*$	$-^*$	0	$-^*$	$-^*$	−0.409	−0.409
ω^*_{44}	$\sqrt{0.5}^* \approx 0.707$	$-^*$	$-^*$	1	$-^*$	$-^*$	0.870	0.870

Parameters which were not estimated are indicated with a dash. The * indicate that a parameter is only used in models M3 and M4

$$V_1 \frac{d\, c_1(t)}{dt} = u(t) + Cl_d\,(c_2(t) - c_1(t)) - \frac{V_{max}\, c_1(t)}{K_m\, c_1(t)}$$
$$V_2 \frac{d\, c_2(t)}{dt} = Cl_d\,(c_1(t) - c_2(t))$$
$$c_1(0) = c_2(0) = 0, \tag{71}$$

where $u(t)$ is an input function, which was used to model a constant infusion with the rate 0.67 per minute during the first 30 minutes followed by another 30 minutes of washout. For models M1 and M2, the scalar-valued observation model was defined by $\mathbf{y}_t = c_1(t) + \mathbf{e}_t$, where $\mathbf{e}_t \in N(0, \mathbf{R}_t)$ and

$$\mathbf{R}_t = \left(r_{a1}^2 \right). \tag{72}$$

For models M3 and M4, the vector-valued observation model was defined by $\mathbf{y}_t = (c_1(t), c_2(t)) + \mathbf{e}_t$, where

$$\mathbf{R}_t = \begin{pmatrix} (r_{a1} + r_{p1} c_1(t))^2 & \\ & r_{a2}^2 \end{pmatrix}. \tag{73}$$

In models M1 and M2, the three parameters V_{max}, K_m, and V_1, were defined to be log-normally distributed on the population level. This was accomplished by multiplying them with $\exp(\eta_1)$, $\exp(\eta_2)$, and $\exp(\eta_3)$, respectively, where $\boldsymbol{\eta} = (\eta_1, \eta_2, \eta_3)$ is normally distributed with zero mean. In the first variant of this model, M1, the covariance matrix for the random effect parameters is defined by the diagonal matrix

$$\Omega = \begin{pmatrix} \omega_{11}^2 & & \\ & \omega_{22}^2 & \\ & & \omega_{33}^2 \end{pmatrix}, \tag{74}$$

and in the second variant, M2, the full matrix is estimated using the parameterization

$$\Omega = \begin{pmatrix} \omega_{11}^2 + \omega_{12}^2 + \omega_{13}^2 & \omega_{12}\omega_{22} + \omega_{13}\omega_{23} & \omega_{13}\omega_{33} \\ \omega_{12}\omega_{22} + \omega_{13}\omega_{23} & \omega_{22}^2 + \omega_{23}^2 & \omega_{23}\omega_{33} \\ \omega_{13}\omega_{33} & \omega_{23}\omega_{33} & \omega_{33}^2 \end{pmatrix} \tag{75}$$

to ensure positive definiteness. In models M3 and M4, an additional random effect parameter was in the same way introduced for the parameter Cl_d. A similarly defined full matrix for 4 random effect parameters was used for models M3 and M4.

The parameter values used for simulating data are shown in Table 2, together with information of which parameters are being estimated in the four model variants, and what the starting values of the estimation were. One data set consisting of 10 simulated individuals was used for models M1 and M2. Here, the values of c_1 were collected at the time points $t = 10, 15, 20, \ldots, 60$. For models M3 and M4, another data set consisting of 20 simulated

individuals was used, where the values of c_1 and c_2 were collected at the time points $t = 10, 15, 20, \ldots, 60$.

References

1. Atkinsona AC, Bogacka B (2002) Compound and other optimum designs for systems of nonlinear differential equations arising in chemical kinetics. Chemom Intell Lab Sys 61:17–33
2. Bauer R (2010) NONMEM7 Technical Guide
3. Bauer R (2014) NONMEM User's Guides Introduction to NONMEM 7.3.0. Hanover, MD
4. Bauer RJ, Guzy S, Ng C (2007) A survey of population analysis methods and software for complex pharmacokinetic and pharmacodynamic models with examples. AAPS J 9(1):E60–E83. doi:10.1208/aapsj0901007
5. Beal S, Sheiner L, Boeckmann A, Bauer R (2009) NONMEM User's Guides. (1989–2009). Icon Development Solutions, Ellicott City, MD
6. Berglund M, Sunnåker M, Adiels M, Jirstrand M, Wennberg B (2012) Investigations of a compartmental model for leucine kinetics using non-linear mixed effects models with ordinary and stochastic differential equations. Math Med Biol 29(4):361–384. doi:10.1093/imammb/dqr021
7. Bertrand J, Laffont CM, Mentré F, Chenel M, Comets E (2011) Development of a complex parent-metabolite joint population pharmacokinetic model. AAPS J 13(3):390–404. doi:10.1208/s12248-011-9282-9
8. Chan PLS, Jacqmin P, Lavielle M, McFadyen L, Weatherley B (2011) The use of the SAEM algorithm in MONOLIX software for estimation of population pharmacokinetic-pharmacodynamic-viral dynamics parameters of maraviroc in asymptomatic HIV subjects. J Pharmacokinet Pharmacodyn 38(1):41–61. doi:10.1007/s10928-010-9175-z
9. Dartois C, Lemenuel-Diot A, Laveille C, Tranchand B, Tod M, Girard P (2007) Evaluation of uncertainty parameters estimated by different population PK software and methods. J Pharmacokinet Pharmacodyn 34(3):289–311. doi:10.1007/s10928-006-9046-9
10. Davidian M, Giltinan DM (2003) Nonlinear models for repeated measurement data: an overview and update. J Agric Biol Environ Stat 8:387–419
11. Gibiansky L, Gibiansky E, Bauer R (2012) Comparison of Nonmem 7.2 estimation methods and parallel processing efficiency on a target-mediated drug disposition model. J Pharmacokinet Pharmacodyn 39(1):17–35. doi:10.1007/s10928-011-9228-y
12. Johansson ÅM, Ueckert S, Plan EL, Hooker AC, Karlsson MO (2014) Evaluation of bias, precision, robustness and runtime for estimation methods in NONMEM 7. J Pharmacokinet Pharmacodyn 41(3):223–238. doi:10.1007/s10928-014-9359-z
13. Kiang TKL, Sherwin CMT, Spigarelli MG, Ensom MHH (2012) Fundamentals of population pharmacokinetic modelling: modelling and software. Clin Pharmacokinet 51(8):515–525. doi:10.2165/11634080-000000000-00000
14. Kristensen NR, Madsen H, Ingwersen SH (2005) Using stochastic differential equations for PK/PD model development. J Pharmacokinet Pharmacodyn 32(1):109–141. doi:10.1007/s10928-005-2105-9
15. Lavielle M (2014) Monolix: a software for the analysis of nonlinear mixed effects models. The Monolix Group
16. Leander J, Lundh T, Jirstrand M (2014) Stochastic differential equations as a tool to regularize the parameter estimation problem for continuous time dynamical systems given discrete time measurements. Math Biosci 251:54–62. doi:10.1016/j.mbs.2014.03.001
17. Mentré F, Mallet A, Baccar D (1997) Optimal design in random-effects regression models. Biometrika 84:429–442
18. Møller JB, Overgaard RV, Madsen H, Hansen T, Pedersen O, Ingwersen SH (2010) Predictive performance for population models using stochastic differential equations applied on data from an oral glucose tolerance test. J Pharmacokinet Pharmacodyn 37(1):85–98. doi:10.1007/s10928-009-9145-5
19. Neidinger RD (2010) Introduction to automatic differentiation and MATLAB object-oriented programming. SIAM Rev 52:545–563
20. Nocedal J, Wright SJ (1999) Numerical optimization. Springer, Berlin
21. Pharsight, Cary, NC: Phoenix NLME
22. Plan EL, Maloney A, Mentré F, Karlsson MO, Bertrand J (2012) Performance comparison of various maximum likelihood nonlinear mixed-effects estimation methods for dose-response models. AAPS J 14(3):420–432. doi:10.1208/s12248-012-9349-2
23. SymPy Development Team: SymPy: Python library for symbolic mathematics (2014). http://www.sympy.org
24. Tapani S, Almquist J, Leander J, Ahlström C, Peletier LA, Jirstrand M, Gabrielsson J (2014) Joint feedback analysis modeling of nonesterified fatty acids in obese Zucker rats and normal Sprague–Dawley rats after different routes of administration of nicotinic acid. J Pharm Sci 103(8):2571–2584. doi:10.1002/jps.24077
25. Wang Y (2007) Derivation of various NONMEM estimation methods. J Pharmacokinet Pharmacodyn 34(5):575–593. doi:10.1007/s10928-007-9060-6