Research Article

Preconditioning of Nonlinear Mixed Effects Models for Stabilisation of Variance-Covariance Matrix Computations

Yasunori Aoki,^{1,2,3} Rikard Nordgren,¹ and Andrew C. Hooker¹

Received 29 July 2015; accepted 4 January 2016; published online 8 February 2016

Abstract. As the importance of pharmacometric analysis increases, more and more complex mathematical models are introduced and computational error resulting from computational instability starts to become a bottleneck in the analysis. We propose a preconditioning method for non-linear mixed effects models used in pharmacometric analyses to stabilise the computation of the variance-covariance matrix. Roughly speaking, the method reparameterises the model with a linear combination of the original model parameters so that the Hessian matrix of the likelihood of the reparameterised model becomes close to an identity matrix. This approach will reduce the influence of computational error, for example rounding error, to the final computational result. We present numerical experiments demonstrating that the stabilisation of the computation using the proposed method can recover failed variance-covariance matrix computations, and reveal non-identifiability of the model parameters.

KEY WORDS: computational stability; identifiability; non-linear mixed effects model; parameter estimation uncertainty; preconditioning.

INTRODUCTION

Non-linear mixed effects models have been shown to be an effective tool for the analysis of clinical trial data. As a result, pharmacometric analysis based on non-linear mixed effects models, also known as the population approach, has become an essential step in drug development. The pharmacometric analysis is usually based on the assumption that the parameter and uncertainty estimates of the non-linear model are estimated correctly by a numerical method; however, these estimates will unavoidably be influenced by the numerical stability of the computational algorithm when the numerical method is based on finite precision computations.

In this paper, we propose a preconditioning method for non-linear mixed effects models to increase the computational stability of the variance-covariance matrix. Preconditioning is a widely used technique to increase the computational stability for numerically solving large sparse systems of linear equations (1,2). Inspired by this technique, we reparameterise the model with a linear combination of the model parameters so that the Hessian of the $-2\ln(\text{likelihood})$ (R-matrix) of the reparameterised model becomes close to an identity matrix. This approach should reduce the influence of computational issues and reduce the chance of the R-matrix

being non-positive definite and also give an indication of when the R-matrix is fundamentally singular.

To test this preconditioning method, we have conducted numerical experiments using published non-linear mixed effects models (and data) from applications in pharmacometrics. Our hypothesis is that preconditioning will reduce the failed variance-covariance matrix computations when the R-matrix is fundamentally positive definite, and correctly indicate a fundamentally singular R-matrix in case of the existence of non-identifiable model parameters. In addition, an automated preconditioning routine is made available as a part of the software package Perl-speaks-NONMEM (PsN) version 4.4 and above (3).

BACKGROUND

Computational instability related to the R-matrix can influence pharmacometric analyses in two ways. First, given that the positive definiteness of the R-matrix is a necessary condition for the estimated parameters to be at the maximum likelihood (*cf.* Appendix A.3), if we are at the maximum likelihood, an R-matrix may appear to be non-positive definite. For example when using a parameter estimation software for a non-linear mixed effects model such as NONMEM, a user may encounter a message like “R-matrix Algorithmically Singular and Non-Positive Semi Definite” and the computation of the variance-covariance matrix will fail (when the model is, in fact, identifiable and the parameters are at the maximum likelihood). In this case, preconditioning will help reduce the misjudgement of the nature of the R-matrix by making the R-matrix of the reparameterised model as close to an identity matrix as possible. Second, if the R-matrix is in theory singular (*e.g.* if a model parameter is not estimable from the data),

¹ Department of Pharmaceutical Biosciences, Uppsala University, Uppsala, Sweden.

² Department of Mathematics, Uppsala University, Uppsala, Sweden.

³ To whom correspondence should be addressed. (e-mail: yaoki@uwaterloo.ca;)

however, due to the computational error in handling a singular matrix, this matrix can appear to be positive definite. In this case, computational instability can lead to the misleading conclusion that the model parameter is estimable, as standard errors may be calculated using the R-matrix. In this situation, preconditioning can help by demonstrating the singularity of the matrix.

Variance-Covariance Matrix

The variance-covariance matrix (M) of a non-linear mixed effects model is used to asymptotically quantify the parameter estimation uncertainty and correlations. The square root of the i th diagonal element of M (i.e. $\sqrt{m_{ii}}$) is an estimate of the standard error (SE) of the i th parameter, while the relative standard error (RSE) can be computed as $\sqrt{m_{ii}}/|\theta_i|$ where θ_i is the value of the i th parameter. Correlations between the i th and j th parameters are computed as $m_{ji}/\sqrt{m_{ii}*m_{jj}}$. M can be estimated as follows (4,5):

$$M := R^{-1}SR^{-1} \quad (1)$$

where matrix R is the Hessian of the $-2\ln(\text{likelihood})$ with respect to the model parameters and matrix S denotes the sum of the cross products of the gradient vectors of the $-2\ln(\text{individual} - \text{likelihood})$. Hence, to obtain the variance-covariance matrix M , we need to have the inverse of R . In addition, if R is not positive semi-definite, then the estimated parameters are not at a minimum of the $-2\ln(\text{likelihood})$ (see [Appendix A.3](#) for a detailed discussion).

Preconditioning aims to avoid the R-matrix being computationally non-positive definite by estimating the parameter s and calculating the R-matrix in a linearly reparameterised parameter space. It uses the R-matrix from a previous parameter estimation and reparameterises the model such that the R-matrix of the preconditioned model is close to an identity matrix. Note that an R-matrix should be, generally, easily obtainable after parameter estimation; it is the inverse of R that may be difficult if it is close to singular. If the R-matrix of the preconditioned model is close to an identity matrix, the chance of the matrix appearing to be non-positive semi-definite due to rounding error will be reduced. In addition, linear transformation of a singular matrix cannot be non-singular; hence, if the R-matrix is fundamentally singular, the R-matrix of the preconditioned model will remain singular, giving a stronger indication that a model parameter is non-identifiable.

METHOD

In this section, we describe the technical details of how we construct the preconditioned model and justify why this method of preconditioning should increase the computational stability.

Preconditioning the Model

The goal of preconditioning is to linearly reparameterise the model so that the R-matrix of the preconditioned model is as close to an identity matrix as possible. In other words, preconditioning will scale and rotate the parameter space so that the

curvature of the likelihood surface becomes the same in all directions.

Let f be a map from the parameter vector θ to $-2\ln(\text{likelihood})$. A number of approximation methods for the likelihood for a non-linear mixed effects model can be found in (6), which can be used to obtain this map. We assume f to be a twice continuously differentiable real function. The Hessian matrix R of $f(\theta)$ is defined as follows:

$$r_{ij} = \frac{\partial^2 f(\theta)}{\partial \theta_i \partial \theta_j}, \quad (2)$$

where r_{ij} is an element in matrix R at the i th row and the j th column. Due to an assumption on the smoothness of f , matrix R is a symmetric matrix.

We write the linear reparameterisation of the model parameter vector θ to be

$$\theta = P \hat{\theta}, \quad (3)$$

where P is a preconditioning matrix and $\hat{\theta}$ is a new parameter vector in a preconditioned model. We choose P to be an invertible matrix so that the estimated parameters and variance-covariance matrix of the preconditioned model can be transformed back to the original parameterisation of the model. We can then write each element of the Hessian matrix of the preconditioned model, $\hat{R}$, as follows:

$$\hat{r}_{ij} = \frac{\partial^2 f(P \hat{\theta})}{\partial \hat{\theta}_i \partial \hat{\theta}_j}. \quad (4)$$

Our goal is to find a preconditioning matrix P so that $\hat{R}$ becomes close to an identity matrix. To do this, we first rewrite $\hat{R}$ in terms of P and R .

$$\hat{r}_{ij} = \frac{\partial^2 f}{\partial \hat{\theta}_i \partial \hat{\theta}_j}, \quad (5)$$

$$= \sum_k \frac{\partial}{\partial \theta_k} \frac{\partial f}{\partial \hat{\theta}_j} \frac{d\theta_k}{d\hat{\theta}_i}, \quad (6)$$

$$= \sum_k \frac{\partial}{\partial \theta_k} \left(\sum_l \frac{\partial f}{\partial \theta_l} \frac{\partial \theta_l}{\partial \hat{\theta}_j} \right) \frac{d\theta_k}{d\hat{\theta}_i}, \quad (7)$$

$$= \sum_k \frac{\partial}{\partial \theta_k} (\nabla f \cdot \mathbf{p}_{\cdot j}) \frac{d\theta_k}{d\hat{\theta}_i}, \quad (8)$$

$$= \left(\sum_k P_{k,i} \begin{bmatrix} \frac{\partial^2}{\partial \theta_k \theta_1} \\ \frac{\partial^2}{\partial \theta_k \theta_2} \\ \vdots \\ \frac{\partial^2}{\partial \theta_k \theta_n} \end{bmatrix} f \right) \mathbf{p}_{\cdot j}, \quad (9)$$

$$= (Rp_{\cdot,i}) \cdot p_{\cdot,j}, \quad (10)$$

$$= (Rp_{\cdot,i})^T p_{\cdot,j}, \quad (11)$$

$$= p_{\cdot,i}^T R p_{\cdot,j}; \quad (12)$$

hence, we have

$$\hat{R} = P^T R P. \quad (13)$$

Given that R is a symmetric matrix, we can obtain an eigendecomposition of R so that

$$R = V \Lambda V^T, \quad (14)$$

where V is a matrix constructed by the normalised eigenvectors, *i.e.* $[v_1, v_2, \dots, v_n]$, and Λ is a diagonal matrix with the corresponding eigenvalues, *i.e.* $\text{diag}(\lambda_1, \lambda_2, \dots, \lambda_n)$. Note that V is a unitary matrix, that is to say $V^T V = I$. We now choose the preconditioning matrix as follows:

$$P = V \Lambda^{-1/2}, \quad (15)$$

where $\Lambda^{-1/2}$ is a diagonal matrix with the reciprocal of the square root of the absolute value of the eigenvalues, *i.e.* $\text{diag}(1/\sqrt{|\lambda_1|}, 1/\sqrt{|\lambda_2|}, \dots, 1/\sqrt{|\lambda_n|})$. Substituting Eq. (15) into Eq. (13) reveals that

$$\hat{R} = (V \Lambda^{-1/2})^T R V \Lambda^{-1/2}, \quad (16)$$

$$= \Lambda^{-1/2} V^T R V \Lambda^{-1/2}, \quad (17)$$

$$= \Lambda^{-1/2} V^T (V \Lambda V^T) V \Lambda^{-1/2} \quad (\text{by the eigendecomposition of } R) \quad (18)$$

$$= \Lambda^{-1/2} \Lambda \Lambda^{-1/2} \quad (\text{since } V \text{ is a unitary matrix}) \quad (19)$$

$$= I. \quad (20)$$

In other words, if the model is preconditioned by the Hessian R of the model using the preconditioning matrix defined in Eq. (15), then the Hessian $\hat{R}$ of the preconditioned model will be an identity matrix.

Transformation of the Estimated Parameters and Variance-Covariance Matrix of the Preconditioned Model Back to the Original Model Parameterisation

Once maximum likelihood estimation of the preconditioned model has been performed, one can transform the parameters to the original parameterisation using Eq. (3). Similarly, the variance-covariance matrix M of the original model parameterisation can be calculated from the variance-covariance matrix of the preconditioned model $\hat{M}$ as follows:

$$M = P \hat{M} P^T. \quad (21)$$

Effect of the Inaccuracy of the R-matrix Used for Preconditioning

In reality, we do not have the exact R-matrix of the model (*e.g.* by the rounding error, the Hessian not being evaluated at the exact maximum of the likelihood, or numerical error of the model evaluation) so we now consider preconditioning using a perturbed Hessian (inaccurate R-matrix). We denote the perturbed Hessian as $\tilde{R}$, and assuming it is symmetric, we can rewrite it with eigendecomposition as follows

$$\tilde{R} = (V + \Delta V)(\Lambda + \Delta\Lambda)(V + \Delta V)^T, \quad (22)$$

where $(\Lambda + \Delta\Lambda)$ is a diagonal matrix with the eigenvalues of $\tilde{R}$, $(V + \Delta V)$ is a matrix containing the eigenvectors of $\tilde{R}$ and V and Λ are as defined in Eq. (14). We now have the preconditioning matrix as follows:

$$\tilde{P} = (V + \Delta V) | \Lambda + \Delta\Lambda |^{-1/2}. \quad (23)$$

The difference between the identity matrix and the R-matrix of the preconditioned model with the preconditioning matrix $\tilde{P}$ can be bounded as follows:

$$\|I - \tilde{P}^T \tilde{R} \tilde{P}\|_F \leq \left(\sum_{i=1}^n \frac{1}{|\lambda_i + \Delta\lambda_i|} \right) \left(\sqrt{\sum_{i=1}^n \Delta\lambda_i^2} + 2\|\Delta V\|_F \sqrt{\sum_{i=1}^n \lambda_i^2} \right). \quad (24)$$

Roughly speaking, this shows that if we use the matrix $\tilde{R}$ that is close to the true Hessian R of the original model (*i.e.* ΔV and $\Delta\Lambda$ are small), then the Hessian of the preconditioned model will be close to the identity matrix if $\lambda_i \neq 0$. Also, we can observe that choosing matrix $\tilde{R}$ with eigenvalues of similar order of magnitude as the true Hessian of the original model is desirable to make the Hessian of the preconditioned model close to an identity matrix.

Iterated Preconditioning

If the R-matrix from the original model can be significantly different from the true R-matrix, preconditioning just once may not resolve the computational instability. We can construct the preconditioning matrix based on a previously run preconditioned model by approximating the R-matrix of the original model as follows:

$$R = (P^T)^{-1} \bar{R} (P)^{-1}, \quad (25)$$

where R is the estimation of the Hessian of the original model, P is the preconditioning matrix used to construct the preconditioned model and $\bar{R}$ is the Hessian of the preconditioned model. We can use this estimated R-matrix of the original model, construct the preconditioning matrix as in Eq. (15) and iteratively precondition.

SOFTWARE IMPLEMENTATION IN PsN

The preconditioning technique proposed in this paper is implemented in the software PsN (versions $\geq 4.4.0$) (7). For detailed instructions on the use of this Perl script, we refer the users to the documentation enclosed in the PsN distribution. As PsN is a collection of Perl scripts to augment the functionality of NONMEM (8), in this section, we assume basic knowledge on how the parameters of a non-linear mixed effects model can be estimated using NONMEM.

Overall Workflow

This section describes the overall workflow when the following PsN command is used:

```
precond < model_name > .mod
```

1. Read in the model file <model_name>.mod (we refer to it as the “original model”).
2. Estimate the model parameters and variance-covariance matrix of the original model using NONMEM and obtain the R-matrix (note again that the R-matrix is relatively problem free to calculate; the inverse is the step that may give problems). The estimated parameters and their standard errors (if the variance-covariance matrix computation is successful) are saved as a text file. If the variance-covariance matrix computation of the original model is successful, then the script terminates and will not proceed to the following steps.
3. Create preconditioning matrix P as defined in Eq. (15).
4. Fixed effect parameters are reparameterised as

$\theta = P\hat{\theta}$ where θ is the vector of fixed effect parameters of the original model and $\hat{\theta}$ is the vector of the fixed effect parameters of the preconditioned model. For example, considering an original model with two fixed effect parameters, this reparameterisation can be found in the preconditioned model as follows:

```
$PK
IF (NEWIND == 0) THEN
  THE_1 = p11 * THETA(1) + p12 * THETA(2)
  THE_2 = p21 * THETA(1) + p22 * THETA(2)
END IF
```

In this case, the vectors θ and $\hat{\theta}$ are

```
 $\theta = [\text{THE\_1}, \text{THE\_2}]^T$ 
 $\hat{\theta} = [\text{THETA}(1), \text{THETA}(2)]^T$ 
```

For models with many THETAs, it is necessary to increase the bounds of the number of variables and constants used by NONMEM. These are set using \$SIZES for DIMTMP, DIMCNS and DIMNEW. To save computational time, THE_x are only calculated as often as the THETAs are updated (using the “IF (NEWIND == 0) THEN ... END IF” structure).

5. THETA(x) in the original model file are replaced with THE_x in all relevant code blocks (currently pk, pred, error, des, aes, aesinitial and infn). For example, $CL = \text{THETA}(1)$ in the original model will be replaced with $CL = \text{THE_1}$ in the preconditioned model.
6. All bounds for thetas in the preconditioned model are removed and the initial estimate of the parameters are updated by $\hat{\theta}_{init}$ where

$$\hat{\theta}_{init} = P^{-1}\theta_{init}.$$

The preconditioned model can be found in m1/<modelname>_repara.mod in the precondition run directory.

7. Estimate the parameters $\hat{\theta}$ and variance-covariance matrix $\hat{M}$ of the preconditioned model using NONMEM.
8. If $\hat{\theta}$ was obtained, then the estimated parameter vector of the original model θ is calculated using Eq. (3) and saved as a text file.
9. If $\hat{M}$ was obtained, then the variance-covariance matrix of the original model M is calculated using Eq. (21) and saved as a text file. Standard errors of θ are computed from M and saved to the file created in the previous step.

Options

Here, we describe two options to the precondition tool in PsN that are used for the numerical experiments presented in “NUMERICAL EXPERIMENTS”. We refer the readers to the documentation of PsN for the complete list of options.

always: If we specify the **-always** option, the script will proceed to step 3 of the workflow regardless of the success of the variance-covariance step. This option can be used to verify that the estimated parameters are consistent between the original model and the ones obtained through preconditioning. If the estimated parameters are not in agreement while the likelihoods are identical, most likely, the parameters are not identifiable from the data (see “Experiment 2: Reveal Non-identifiability” for an example). Also, if the computational results of the variance-covariance matrix will be used for further analysis, we strongly recommend using this option as the variance-covariance matrix obtained through preconditioning is more accurate (less influenced by the rounding error and less dependent on the computational environment) than the original model.

pre=precond_dir1: With this option we can conduct multiple, iterative, preconditioning steps as presented in “Iterated Preconditioning”. Step 2 of the workflow will be skipped and in step 3, the preconditioning matrix is created as in Eq. (25) with the R-matrix obtained from the previous preconditioning step. Multiple preconditioning can be repeated indefinitely (*i.e.* by using the options **-pre=precond_dir1 -pre=precond_dir2 -pre=precond_dir3 ...**)

as long as the R-matrix from the previous preconditioned model was obtained.

Limitations

The current implementation of preconditioning in PsN only preconditions the fixed effects portion of the model. If the user wishes to precondition both the fixed effects and random effects portions of the model, then a reparameterisation of the model can be done as presented in [Appendix C.1.1](#). Also, the current implementation neglects constraints on the parameter search space (*i.e.* boundaries of the fixed effect parameters set in the \$THETA record). If the sign of the parameter is crucial in order for it to be estimated appropriately, negativity or positivity of the parameter can be imposed through simple reparameterisation as presented in [Appendix C.1.2](#). The current implementation cannot precondition mixture models (*i.e.* cannot reparameterise a model with a \$MIX record).

NUMERICAL EXPERIMENTS

In order to illustrate that preconditioning can be a useful tool for pharmacometric analyses, we have conducted numerical experiments using NONMEM version 7.3 and PsN version 4.4.0 (version 4.5.6 for SIR analysis) on a Linux Cluster (2.4GHz Intel Xeon E5645, Scientific Linux release 6.5, GCC 4.4.7, Perl 5.10.1). We have used three published models as described in Appendices C.2.1–C.2.4 and one realistic simulation study described in [Appendix C.2.5](#).

Experiment 1: Recover Failed Variance-Covariance Matrix Computation

Computational instability in the computation of the variance-covariance matrix can make the R-matrix appear to be non-positive semi-definite.

In theory, if the R-matrix is not positive semi-definite and the gradient of the likelihood is zero, then the estimated parameter is located at a saddle point of the likelihood surface; hence, it is not at the maximum of the likelihood. In practice, instability in the computation of the variance-covariance matrix can make the R-matrix appear to be non-positive semi-definite. In this case, NONMEM will terminate the variance-covariance matrix computation. Preconditioning can rectify this situation and allow the variance-covariance matrix to be computed.

This computational instability of the R-matrix can be caused by computational inaccuracy of the parameter estimation, the model evaluation (especially when the model evaluation includes numerical approximation, *e.g.* ODEs and matrix exponential) and finite difference step scheme for computing the second derivative.

Through this numerical experiment, we show that preconditioning can improve the accuracy of the computation so that an R-matrix evaluated at the (local) maximum of the

likelihood will be positive semi-definite (*i.e.* if the R-matrix is theoretically positive semi-definite); hence, the variance-covariance matrix can be obtained.

For models 1, 2 and 3, we have first estimated the model parameters from data and then, using the estimated parameters and the model, simulated 100 data sets with the same data structure as the original data and reestimated the parameters and variance-covariance matrix for each simulated data set. This process is automated using the SSE script of PsN. For model 4, as it is a time-to-event model and to avoid creation of a simulation data set, we have used a case-sampling bootstrap method to create 100 different data sets and reestimated the parameters and variance-covariance matrix for each bootstrapped data set. For model 4, we used the bootstrap script of PsN to conduct this numerical experiment. To maintain the reproducibility of the results, we have fixed the random seed. We observe that the computation of the variance-covariance matrix for these simulations and reestimations failed 36, 31, 20 and 13% of the time for models 1, 2, 3 and 4, respectively.

For the data sets for which the computation of the variance-covariance matrix was not successful, we have iteratively repeated preconditioning until the computation was successful. As can be seen in [Table I](#), most of the variance-covariance matrices of the models can be computed when the computation was stabilised by preconditioning (a variance-covariance matrix computation failure rate of 1, 1, 6 and 0% for models 1, 2, 3 and 4, respectively).

For model 1, we have repeated the same numerical experiment using the Laplace approximation of the likelihood. We observe that the computation of the variance-covariance matrix failed 72% of the time and that after preconditioning the variance-covariance matrix calculation failure rate decreased to 40%.

Experiment 2: Reveal Non-identifiability

Through this numerical experiment, we show that numerical instability can potentially hide non-identifiability of the model parameters and preconditioning can help discover the non-identifiability of the model parameters. We show this using model 5 (see [Appendix C.2.5](#) for the details of the model) and first conduct a standard analysis on the parameter estimation uncertainty quantification and then conduct an analysis using preconditioning. The standard analysis suggests that the parameters can be estimated accurately; however, the preconditioning contradicts this by demonstrating the existence of two different sets of model parameters with the same maximised likelihood.

Standard Analysis

As proven in [Appendix C.2.5](#), model 5 is not structurally identifiable; hence, the Hessian matrix R is, in theory, a singular matrix (*i.e.* the inverse of this matrix does not exist). Thus, the variance-covariance matrix as defined in [Eq. \(1\)](#) does not exist. Despite this, the R-matrix is incorrectly calculated to be non-singular and the

variance-covariance matrix was obtained using NONMEM in our computational environment. The minimisation was determined to be “successful” according to NONMEM. As an alternative to the variance-covariance matrix, a case-resampling bootstrap method or a sampling importance resampling (SIR) method (9,10) could be used to quantify the parameter estimation uncertainty. We have conducted both methods on the original model using PsN, obtained the parameter estimation uncertainty and tabulated the standard errors in Table II. All of the variance-covariance matrix, the bootstrap method and SIR method suggest reasonable relative standard errors, so one would conclude that the parameters can be estimated accurately from this data set (despite the model being not structurally identifiable). Note that we have used the (in this case incorrectly computed) variance-covariance matrix to create the proposal distribution (default setting of SIR implemented in PsN 4.5.6). The bootstrap method relies on the parameter estimation where the parameters are locally searched near the user-specified initial estimate to fit the bootstrap data. Thus, as can be seen in Table II, if the bootstrap method is run on the non-estimable model, the estimated bootstrap parameters remain close to the initial estimate.

Analysis Using Preconditioning

After preconditioning, the variance-covariance matrix was still obtainable; however, as can be seen in Table III, the large SEs for the parameters that are not identifiable which clearly indicate these parameters cannot be estimated. The eigenvalues of the R-matrix of the original model and after preconditioning are tabulated in Table IV.

Table I. Number of Successful Variance-Covariance Matrix Computations

	Model 1	Model 1 (Laplace)	Model 2	Model 3	Model 4
Original model without user-defined boundaries	64	28	69	80	87
Pre = 1	30	11	17	5	10
Pre = 2	4	10	4	5	1
Pre = 3	1	4	8	3	
Pre = 4		3	0		1
Pre = 5		2	1	1	1
Pre = 6		2			
Pre > 6		17	1		
Estimation failed S-matrix unobtainable	1	23		4	
var-cov matrix saved by precondition	35	32	30	14	13

Out of 100 variance-covariance matrix computations from simulated data sets for each model for models 1, 2 and 3 and bootstrap data sets for model 4. Pre = 1, 2, 3, 4, 5 and 6 indicate the preconditioning was applied one, two, three, four, five and six times, respectively. Pre > 6 indicates that the preconditioning was not successful after preconditioning six times

Table II. Estimated Parameters and Relative Standard Error (RSE) of Model 5

	V1	Q	V2	CL	PRV
Estimated parameters	4.19	15.3	9.83	2.83	0.104
RSE by var-cov matrix (%)	51.7	49.5	47.0	43.4	2.68
RSE by bootstrap (%)	32.0	31.1	30.8	33.0	2.36
RSE by SIR (%)	43.8	42.6	39.7	36.9	3.03

Bootstrap: default setting (minimisation was successful 95% of the time). SIR: samples = 1000, 1000, 1000, 2000, 2000; resamples = 200, 400, 500, 1000, 1000. We have confirmed the SIR iteration has converged after five iterations using a diagnostic plot of the distribution of dOFV

PRV proportional residual variability

An almost zero eigenvalue and an increase in the condition number after preconditioning suggest the R-matrix to be fundamentally singular; hence, the parameters are not locally practically identifiable (non-estimable) from the data. In fact, as can be seen in Table III, the estimated parameters from the original model and through the preconditioned model are significantly different despite almost identical likelihoods. Thus, the parameters of this model with the given data cannot be estimated uniquely in the maximum likelihood sense. In order to cross-check this result, we have used the parameters obtained through the preconditioned model that appears in Table III and used them as the initial estimate in the original model and reestimated the parameters. The reestimated parameters were identical to what appears in Table III even with the original model after moving the initial estimates. Hence, the parameters we have found through preconditioning were not an artefact of preconditioning but due to the fact that the model parameters were not estimable from the given data. In addition, this result is consistent with our analytic proof in Appendix C.2.5 showing that only the ratios between the parameters V1, Q, V2 and CL are identifiable.

Additionally, we have conducted SIR analysis using the variance-covariance matrix obtained from preconditioning to construct the proposal distribution. As can be seen in Table III, SIR clearly shows some of the model parameters

Table III. Estimated Parameters and Relative Standard Errors (RSEs) After Preconditioning of Model 5

	V1	Q	V2	CL	PRV
Estimated parameters with preconditioning	25.3	92.7	59.5	17.1	0.104
SE by var-cov matrix after preconditioning	103	385	243	69.6	0.00279
Centre estimate of parameters by SIR	4.19	15.3	9.83	2.83	0.104
SE by SIR using precondition var-cov matrix	189	702	445	127	0.00340

SIR: samples = 1000, 1000, 1000, 2000, 2000; resamples = 200, 400, 500, 1000, 1000. We have confirmed the SIR iteration has converged after five iterations using a diagnostic plot of the distribution of dOFV

PRV proportional residual variability

Table IV. Eigenvalues of the Hessian of the $-2\ln(\text{likelihood})$ with Respect to Fixed Effect Parameters of Model 5 With or Without Preconditioning

	Original model without user-defined boundaries	After preconditioning
Eigenvalues	102,000	102,000
	8.55	0.237
	2.70×10^{-5}	7.44×10^{-8}
	1.95	0.0546
	85.5	2.35
Condition number	3.79×10^9	1.37×10^{12}
$-2\ln(\text{likelihood})$	-846	-846

to be not estimable if the proposal distribution was properly constructed.

Simulation Study

In order to further validate that preconditioning is likely to reveal non-identifiability of the model, we have conducted simulation studies using this model. We have simulated 100 data sets from the model and reestimated the parameters and variance-covariance matrix using the SSE tool in PsN. For 90% of the data sets, the variance-covariance matrix was obtainable, and in 47% of the cases, the maximum RSE was less than 100%. We have preconditioned the model using the **-always** option, and after preconditioning, 62% of the time, the covariance calculation was correctly terminated with the correct detection of the singular R-matrix. For the other 38% of the cases where the variance-covariance matrices were obtainable, all the RSEs of structurally non-identifiable parameters calculated from these matrices were above 100%, clearly indicating the estimability issues.

DISCUSSION

We have introduced a preconditioning technique for non-linear mixed effects models to reduce the computational instability that can potentially influence the results of pharmacometric analyses. Through numerical experiments, we have shown that preconditioning can recover failed variance-covariance matrix computations and reveal non-identifiability of the parameters.

Preconditioning can be thought of as an extension of a normalisation of model parameters. Alternatively, normalisation can be thought of as “preconditioning” with a diagonal matrix. As normalisation of model parameters is implemented in most of the parameter estimation algorithms including the ones implemented in NONMEM, we believe the utilisation of the ideas presented in this paper can be easily translated to many parameter estimation and variance-covariance matrix computation algorithms for non-linear mixed effects models.

Transformation of model parameters, including normalisation, is usually done to allow the order of magnitudes of the parameters to be similar during parameter estimation computations. The essential differences of preconditioning

compared to the usual parameter transformation is that preconditioning scales the parameters to make the second derivatives of $-2 \log$ likelihood with respect to the parameters (*i.e.* diagonal elements of the R-matrix) to be as close to each other as possible. At the same time, preconditioning aims to reduce the parameter-parameter correlations (*i.e.* making the off-diagonal elements of the R-matrix as close to zero as possible).

As shown in experiment 1, preconditioning can be used for various types of models (*e.g.* for a model given as analytic functions, a system of ODEs where the solution is approximated through numerical integration or using matrix exponential) and various non-stochastic estimation methods (*e.g.* FOCE and Laplace).

As shown in experiment 2, the computation issues for variance-covariance matrix calculations can produce potentially misleading results. As such, we recommend the use of preconditioning if the variance-covariance matrix is to be used for a proceeding analysis (*e.g.* calculating and reporting standard errors or constructing the proposal density for SIR analysis). The use of repeated preconditioning until one obtains two variance-covariance matrices with similar standard errors should further guarantee the accuracy of the variance-covariance matrix. Also, it should be reminded to the readers that, as demonstrated in experiment 2, if the parameters are not estimable from the data (*e.g.* the model is over-parameterised), the R-matrix is, in theory, singular and, hence, the variance-covariance matrix does not exist. Thus, the correct behaviour of the computational algorithm is to indicate that the matrix is not obtainable. In addition, if the estimated parameter is at the saddle point, the R-matrix is, in theory, non-positive definite and, hence, the correct behaviour of the computational algorithm is to indicate that the matrix is also not obtainable.

There are various more sophisticated methods that can be used to quantify parameter estimation uncertainties. As more advanced methods usually require considerably more computation than a single variance-covariance matrix computation, we would recommend obtaining the accurately computed variance-covariance matrix first and then proceeding to a further analysis of the uncertainty. Especially as demonstrated in experiment 2, the variance-covariance matrix and the R-matrix can give a strong indication on the non-estimability of the model parameters if computed correctly, while other methods may not be able to do so.

During our investigation, we have observed that although preconditioning remedies most of the computation issues related to R-matrix, there are many other sources of computation instabilities, for example S-matrix computation and parameter estimation. Thus, we believe further investigation into computational stability issues on the numerical algorithms used in pharmacometrics could be beneficial to the field.

ACKNOWLEDGMENTS

The authors would like to thank the Pharmacometrics Research group at Uppsala University for stimulating discussions and the testing of the PsN scripts. We would especially like to thank Mats O. Karlsson, Kajsa Harling, Anne-Gaëlle

Dosne, Anders Kristoffersson, Oskar Alskär and Giulia Corn for discussions and extensive testing. In addition, we would like to thank the authors of (11–14) who gave us permission to use their NONMEM model files and data sets for the numerical experiments.

A. LINEAR ALGEBRA USED IN THIS PAPER

As our idea of preconditioning was inspired by research in the field of numerical linear algebra, to completely understand the technical detail of the proposed method, some knowledge in linear algebra is required. As this is unusual for a paper in pharmacology, in this appendix, we summarise all the linear algebra we have used in this paper. As only real square matrices appear in this paper, we limit our discussions in this section to these matrices.

A.1 Matrix/Vector Notations

In this paper, we denote a scalar quantity with a lower case letter, e.g. a , c ; a matrix with a capital letter, for example A or M ; and a vector quantity by a bold symbol of a lower case letter, e.g. $\mathbf{v}$, $\mathbf{a}$, unless otherwise specifically stated.

In addition, the i th row of the matrix M can be denoted by a vector $\mathbf{m}_{i.}$, the j th column of the matrix M can be denoted by a vector $\mathbf{m}_{.j}$ and the i th row and j th column of the matrix M can be written by a scalar m_{ij} .

Also, we denote a diagonal matrix as $\text{diag}(\mathbf{a})$ where the i th row and i th column element of the diagonal matrix is given as the vector element a_i .

The following symbols are reserved for the special quantities throughout the paper

M : variance-covariance matrix of the estimated model parameters (26)

R : Hessian of the $-2\ln(\text{likelihood})$ (27)

λ_i : i th largest eigenvalue (28)

n : the size of the matrix of interest (29)

In this paper, we exclusively use an l_2 norm as the vector norm which is denoted and defined as follows:

$$\|\mathbf{a}\| = \sqrt{\sum_i a_i^2}. \quad (30)$$

We also use an l_2 vector norm-induced matrix norm that is defined as follows unless specifically stated otherwise:

$$\|A\| := \max_{\|\mathbf{x}\|=1} \|A\mathbf{x}\| \quad (31)$$

It can be shown that the l_2 vector norm-induced matrix norm of a real square matrix is equivalent to the largest absolute eigenvalue of the matrix.

A.2 Eigenvalues, Eigenvectors and Eigendecomposition

In this paper, we have made an extensive use of eigendecomposition, so in this section, we briefly describe the idea of eigendecomposition. As our interest is for the eigendecomposition of a symmetric matrix (e.g. R -matrix), we limit our discussion to a real symmetric matrix.

Definition A.1 Let A be an $n \times n$ real square matrix, we define the following vector $\mathbf{v}_i$ and scalar λ_i to be the eigenvector and eigenvalue:

$$A\mathbf{v}_i = \lambda_i \mathbf{v}_i \quad (32)$$

Note that the eigenvector for a given non-zero eigenvalue is only unique up to a multiplicative constant, so for the rest of this section, we will only consider the normalised eigenvector (i.e. $\|\mathbf{v}_i\|=1$).

Theorem A.1 (eigendecomposition) Let A be an $n \times n$ real symmetric matrix; then, there exist matrices V and Λ such that

$$A = V\Lambda V^T \quad (33)$$

where

$$V = [\mathbf{v}_1 \mathbf{v}_2 \dots \mathbf{v}_n] \quad (34)$$

$$\Lambda = \begin{bmatrix} \lambda_1 & 0 & & 0 \\ 0 & \lambda_2 & & 0 \\ & & \ddots & \\ 0 & 0 & & \lambda_n \end{bmatrix}. \quad (35)$$

Eq. (33) can also be written as follows:

$$A = \sum_{i=1}^n \lambda_i \mathbf{v}_{:,i} \mathbf{v}_{:,i}^T. \quad (36)$$

Given that $\mathbf{v}_i$ are normalised eigenvectors, we have that $\mathbf{v}_i \cdot \mathbf{v}_j = \delta_{ij}$; hence, V is a unitary matrix, i.e. $VV^T = I$. Thus, by following the definition of matrix inverse, we can write the inverse of A as follows:

$$A^{-1} = V\Lambda^{-1}V^T \quad (37)$$

$$= \sum_{i=1}^n \frac{\mathbf{v}_{:,i} \mathbf{v}_{:,i}^T}{\lambda_i}. \quad (38)$$

Notice that if $\lambda_i = 0$, then the inverse of the matrix A^{-1} does not exist (i.e. matrix A is singular if and only if $\lambda_i = 0$ for some i).

A.3 Symmetric Positive (Semi-)Definite Matrices

One way to characterise a matrix is positive definiteness. This can be used to characterise a point (maximum, minimum, saddle point) in a surface when the matrix is the Hessian matrix of a function describing a surface. As we are mainly interested in the characterisation of a symmetric matrix, we limit our discussion to a real symmetric matrix.

Definition A.2 A real symmetric matrix A is said to be symmetric positive definite if and only if

$$\mathbf{z}^T A \mathbf{z} > 0 \quad (39)$$

for any non-zero vector $\mathbf{z}$. A real symmetric matrix A is said to be symmetric positive semi-definite if and only if

$$\mathbf{z}^T A \mathbf{z} \geq 0 \quad (40)$$

for any non-zero vector $\mathbf{z}$.

Proposition A.1 If a real symmetric matrix is positive definite, then all the eigenvalues of the matrix are positive. If a real symmetric matrix is positive semi-definite, then the eigenvalues of the matrix are either zero or positive.

Proof: This follows from the eigendecomposition of the matrix.

Corollary A.1 If a real symmetric matrix is positive definite, then it is invertible (non-singular). If a real symmetric matrix is not positive definite but is positive semi-definite, then it is singular.

Proof: This follows from Proposition and Eq. (38).

Proposition A.2 If a Hessian of a function is positive/negative definite at a point where the gradient vector is a zero vector, then it is a local minimum/maximum. If a Hessian is not singular, not positive definite nor negative definite at a point where the gradient vector is a zero vector, then it is evaluated at a saddle point.

Corollary A.2 If the R-matrix (Hessian of $-\ln(\text{likelihood})$) is non-positive semi definite but non-singular, then the point R-matrix is evaluated not at the local minimum; hence, the estimated parameter is not the maximum likelihood estimator. If the R-matrix is singular, either the R-matrix is evaluated not at the local minimum; hence, the estimated parameter is not the maximum likelihood estimator

or one or more model parameters are not identifiable in a maximum likelihood sense.

Proof: Follows from the definition of R-matrix and proposition.

B. PROOF OF INEQUALITY (24)

In this appendix, we show the proof for Eq. (24) that justifies the R-matrix of the preconditioned model will be close to an identity matrix. By Eqs. (14) and (22), the difference between the preconditioned R-matrix and identity matrix can be written as follows:

$$\begin{aligned} I - \tilde{P}^T R \tilde{P} &= \tilde{P}^T \tilde{R} \tilde{P} - \tilde{P}^T R \tilde{P} \\ &= \tilde{P}^T (\tilde{R} - R) \tilde{P} \\ &= \tilde{P}^T \left((V + \Delta V)(\Lambda + \Delta\Lambda)(V + \Delta V)^T - V\Lambda V^T \right) \tilde{P} \\ &= \tilde{P}^T (V\Lambda V^T + (V + \Delta V)\Lambda\Delta V^T + (V + \Delta V)\Delta\Lambda(V + \Delta V)^T \\ &\quad + \Delta V\Lambda V^T - V\Lambda V^T) \tilde{P} \\ &= \tilde{P}^T \left((V + \Delta V)\Lambda\Delta V^T + (V + \Delta V)\Delta\Lambda(V + \Delta V)^T + \Delta V\Lambda V^T \right) \tilde{P} \end{aligned} \quad (41)$$

We now consider the Frobenius norm of the difference between the preconditioned R-matrix and identity matrix. The Frobenius norm has the following properties (cf. (15)):

$$\|A\|_F = \|A^T\|_F \quad (42)$$

$$\|A + B\|_F \leq \|A\|_F + \|B\|_F \quad (43)$$

$$\|AB\|_F \leq \|A\|_F \|B\|_F \quad (44)$$

$$\|\text{diag}(\mathbf{a})\|_F = \sqrt{\sum_{i=1}^n a_i^2}. \quad (45)$$

In addition, by using the fact V and $(V + \Delta V)$ are orthogonal matrices, we have

$$\|VA\|_F = \|(V + \Delta V)A\|_F = \|A\|_F \quad (46)$$

Using Eqs. (42)–(46), we can bound the Frobenius norm difference between the preconditioned R-matrix and identity matrix as follows:

$$\begin{aligned} \|I - \tilde{P}^T R \tilde{P}\|_F &\leq \|(\Lambda + \Delta\Lambda)^{-\frac{1}{2}}\|_F^2 \left(\|(V + \Delta V)\Lambda\Delta V^T + (V + \Delta V)\Delta\Lambda(V + \Delta V)^T + \Delta V\Lambda V^T\|_F \right) \\ &\leq \|(\Lambda + \Delta\Lambda)^{-\frac{1}{2}}\|_F^2 (\|A\|_F \|\Delta V\|_F + \|\Delta\Lambda\|_F + \|\Delta V\|_F \|A\|_F) \\ &= \left(\sum_{i=1}^n \frac{1}{\lambda_i + \Delta\lambda_i} \right) \left(\sqrt{\sum_{i=1}^n \Delta\lambda_i^2} + 2\|\Delta V\|_F \sqrt{\sum_{i=1}^n \lambda_i^2} \right) \end{aligned} \quad (47)$$

where λ_i and $\Delta\lambda_i$ are the i th diagonal element of the diagonal matrices Λ and $\Delta\Lambda$, respectively.

C. NONMEM TECHNIQUES AND MODELS USED FOR THE NUMERICAL EXPERIMENTS

C.1 Techniques

The current implementation of preconditioning in PsN can only precondition fixed effect parameters. Additionally, all the bounds for the parameters will be removed. To accommodate these limitations, the following tricks for the NONMEM model file can be useful.

C.1.1 Reparameterisation of OMEGA by THETA

In order to precondition the random effects portion of the non-linear mixed effects model using the PsN script, we need to rewrite the variance and covariance of the inter-individual variability (“OMEGA block”) in terms of the fixed effect parameters (“THETAs”). The individual parameters (“ETAs”) can be reparameterised as follows (an example of three random effect parameters with correlation):

```
$PRED, $PK, etc.
ETA_1 = THETA(15)*ETA(1)
ETA_2 = THETA(19)*ETA(1) + THETA(20)*ETA(2)
ETA_3 = THETA(21)*ETA(1) + THETA(22)*ETA(2)
      + THETA(23)*ETA(3)
$OMEGA
1 FIX
1 FIX
1 FIX
```

This gives THETA(18,...23) to be the Cholesky decomposition of the variance-covariance matrix of the inter-individual variability (OMEGA matrix). Similar reparameterisation can be done for the variances of the residual errors (SIGMA).

C.1.2 Boundary of the Parameter

The boundaries of the models are treated using non-linear transformations of the parameter space internally in NONMEM. We have observed that this non-linear transformation can introduce additional computational instability including computational environment dependencies. Thus, we have removed all the boundaries on the fixed effect parameters when running the numerical experiments. In order to avoid termination of parameter estimation due to having the “incorrect” sign of one of the variables (e.g. NONMEM implicitly requires clearance and volume to be positive real values in the predefined models of ADVAN1–4 and if these variables become negative during parameter estimation, the parameter estimation terminates), we can impose the positivity of the variables and parameters by taking the absolute

values. If this technique is used for a parameter, the final estimate of the parameter may be negative; however, the correctly signed parameter estimate can be obtained by taking the absolute value of the final estimate. For example, we can force a clearance value to be positive, independent of the sign of the estimated parameter value for THETA(1), as follows:

```
$PK
CL = ABS(THETA(1))*EXP(ETA(1)).
$THETA 0.1
$OMEGA 0.1
```

C.2 Models

We have used the following models for our numerical experiments. These models were chosen based on run time and accessibility of the data sets and the model files. For all models, the UNCONDITIONAL statement was added to the \$COV line so that the calculation of the variance-covariance matrix would be attempted regardless of the successfulness of the minimisation of the objective function value. In addition, for all models, we have removed the boundaries (cf. [Appendix C.1.2](#)) and set the initial estimate of the parameters to be the previously found best parameter estimates (i.e. published model parameters) to reduce the risk of our numerical experiment being influenced by the numerical instability of the parameter estimation. The step sizes of the finite difference scheme for calculating the R-matrix are set to default values for all the models. Numerical experiments using various step sizes are presented in [Appendix D](#).

C.2.1 Model 1: Pharmacokinetics Model of Paclitaxel of Bergmann et al. (11)

This is a model of the concentration of paclitaxel in blood plasma. It is a two-compartment model with iv infusion with six covariate relationships. The model equation is given as a closed form expression (using ADVAN3 in NONMEM). The data set contains measured data from 93 human subjects with a total of 274 observations. The data set also contains the covariate information of these subjects. The NONMEM model and the data set were provided by the authors of (11). We have added the ABS() statement to the model variables that cannot be negative, and also we have reparameterised the random effect parameters using fixed effect parameters (see [Appendix C.1.1](#) for the details on how this reparameterisation can be performed) so that both fixed effect and random effect parameters can be preconditioned. As this model is a standard PK model, all of the parameters are expected to be identifiable; however, one of the elements in the variance-covariance matrix for the inter-individual variability (omega block) is close to zero, thus introducing the computational instability.

C.2.2 Model 2: Pharmacokinetics Model with Time-Varying Covariates of Paclitaxel of Wahlby et al. (12)

This is a model of the concentration of paclitaxel in blood plasma. The time-varying covariates are included in this model as presented in (12); hence, the drug concentration is modelled using a system of ordinary differential equations (ODEs) and solution of the ODE is approximated using a time-step integration method (using ADVAN6 with TOL=5 in NONMEM). The data set contains measured data from 47 human subjects with a total of 530 observations. The data set also contains the covariate information of these subjects. The NONMEM model file and the data set were provided by the authors of (12). The random effect parameters were reparameterised using fixed effect parameters in order to precondition both fixed effect and random effect parameters.

C.2.3 Model 3: Pharmacokinetics and Pharmacodynamics of Monoclonal Antibody ATM-027 of Zingmark et al. (13)

This is a model of the drug effect of ATM-027. The pharmacokinetics is modelled using two compartments and the drug effect on the receptor expression using the Emax and Sigmoidal-Emax model based on the exposure and plasma concentration. The model equation is solved using the numerical approximation of matrix exponential (using ADVAN7 in NONMEM). The data set is from a phase I study and contains 14 human subjects with total of 301 observations. The NONMEM model file and the data set were provided by the authors of (13).

C.2.4 Model 4: A Time-to-Event Model for Acute Rejections in Paediatric Renal Transplant Recipients Treated with Cyclosporin A of Frobel et al. (14)

This is a parametric survival model used to describe the time to first acute rejection event. This is a fixed effects model without a random effect variable (*i.e.* no inter-individual variability); however, the authors have used NONMEM to estimate the parameters. The model was written as a single ODE (using ADVAN6 with TOL=6 in NONMEM). The data set contains measured data from 87 human subjects with a total of 87 observations. The data set also contains the covariate information of these subjects. The NONMEM model file and the data set were provided by the authors of (14).

C.2.5 Model 5: Pharmacokinetics Model Using Fraction of Administered Amount Data

This model was inspired by published work on the modelling of pharmacokinetics of a substance where the administered amount is unknown but the time-course data of the fraction of the administered amount in the blood is available. It was modelled in NONMEM as a standard PK

model where the observations are in amount, not concentrations, relative to an assumed (and arbitrary) dose. We have simulated the data using a two-compartment model with iv bolus administration with inter-individual variabilities on the clearance (CL) and the volume of distribution of the central compartment (V_1) and proportional residual error. The administered amount is fixed to be 100% and observations were the fraction of the administered amount in % (*i.e.* AMT in data set is set to be 100 at time 0 for all individual). The simulated data set contains 25 subjects with a total of 637 observations. We here show that the parameters of the standard two-compartment model are not structurally identifiable from this type of observation. This proof is inspired by a standard proof of the uniqueness of the solution of a linear system of ODEs (*e.g.* (16)).

Let $\mathbf{u}$ and $\tilde{\mathbf{u}}$ be the solution of the following systems of ODEs and initial conditions (initial value problems) defining the two-compartment model with iv bolus administration with linear elimination:

$$\frac{d}{dt} u_1 = (-CL - Q) \frac{u_1}{V_1} + Q \frac{u_2}{V_2} \quad (48)$$

$$\frac{d}{dt} u_2 = Q \frac{u_1}{V_1} - Q \frac{u_2}{V_2} \quad (49)$$

$$u_1|_{t=0} = 100 \quad (50)$$

$$u_2|_{t=0} = 0 \quad (51)$$

$$\frac{d}{dt} \tilde{u}_1 = (-\tilde{CL} - \tilde{Q}) \frac{\tilde{u}_1}{\tilde{V}_1} + \tilde{Q} \frac{\tilde{u}_2}{\tilde{V}_2} \quad (52)$$

$$\frac{d}{dt} \tilde{u}_2 = \tilde{Q} \frac{\tilde{u}_1}{\tilde{V}_1} - \tilde{Q} \frac{\tilde{u}_2}{\tilde{V}_2} \quad (53)$$

$$\tilde{u}_1|_{t=0} = 100 \quad (54)$$

$$\tilde{u}_2|_{t=0} = 0, \quad (55)$$

where u_1 is the fraction of the administered amount of the substance in blood in %. Assuming all parameters are positive real numbers, one can show that the solutions to the above initial value problems exist and are unique. By subtracting these two initial value problems, we obtain the following initial value problem:

$$\frac{d}{dt} (u_1 - \tilde{u}_1) = \frac{(-CL - Q)}{V_1} u_1 - \frac{(-\tilde{CL} - \tilde{Q})}{\tilde{V}_1} \tilde{u}_1 + \frac{Q_1}{V_2} u_2 - \frac{\tilde{Q}}{\tilde{V}_2} \tilde{u}_2 \quad (56)$$

$$\frac{d}{dt} (u_2 - \tilde{u}_2) = \frac{Q}{V_1} u_1 - \frac{\tilde{Q}}{\tilde{V}_1} \tilde{u}_1 - \frac{Q}{V_2} u_2 + \frac{\tilde{Q}}{\tilde{V}_2} \tilde{u}_2 \quad (57)$$

$$u_1 - \tilde{u}_1 \Big|_{t=0} = 0 \quad (58)$$

$$u_2 - \tilde{u}_2 \Big|_{t=0} = 0. \quad (59)$$

We now supposed the following relationships between the parameters

$$\tilde{CL} = \alpha CL \quad (60)$$

$$\tilde{Q} = \alpha Q \quad (61)$$

$$\tilde{V}_1 = \alpha V_1 \quad (62)$$

$$\tilde{V}_2 = \alpha V_2 \quad (63)$$

where α is a non-zero constant. Substituting Eqs. (60)–(63) into Eqs. (56)–(57) gives the following:

$$\frac{d}{dt}(u_1 - \tilde{u}_1) = \frac{(-CL - Q)}{V_1} u_1 - \frac{(-\alpha CL - \alpha Q)}{\alpha V_1} \tilde{u}_1 + \frac{Q}{V_2} u_2 - \frac{\alpha Q}{\alpha V_2} \tilde{u}_2 \quad (64)$$

$$\frac{d}{dt}(u_2 - \tilde{u}_2) = \frac{Q}{V_1} u_1 - \frac{\alpha Q}{\alpha V_1} \tilde{u}_1 - \frac{Q}{V_2} u_2 + \frac{\alpha Q}{\alpha V_2} \tilde{u}_2 \quad (65)$$

Observe that all α in Eqs. (64)–(65) cancel and obtain the following ODEs

$$\frac{d}{dt}(u_1 - \tilde{u}_1) = \frac{(-CL - Q)}{V_1} (u_1 - \tilde{u}_1) + \frac{Q}{V_2} (u_2 - \tilde{u}_2) \quad (66)$$

$$\frac{d}{dt}(u_2 - \tilde{u}_2) = \frac{Q}{V_1} (u_1 - \tilde{u}_1) - \frac{Q}{V_2} (u_2 - \tilde{u}_2) \quad (67)$$

By rewriting $(u - \tilde{u}) = \hat{u}$, we obtain the following system of ODEs and initial condition:

$$\frac{d}{dt} \hat{u}_1 = \frac{(-CL - Q)}{V_1} \hat{u}_1 + \frac{Q}{V_2} \hat{u}_2 \quad (68)$$

$$\frac{d}{dt} \hat{u}_2 = \frac{Q}{V_1} \hat{u}_1 - \frac{Q}{V_2} \hat{u}_2 \quad (69)$$

$$\hat{u}_1 \Big|_{t=0} = 0 \quad (70)$$

$$\hat{u}_2 \Big|_{t=0} = 0. \quad (71)$$

We can show that $\hat{u} = 0$; hence, $u = \tilde{u}$. (This can be thought of as a two-compartment model without any drug

administration; hence, all amounts are zero at all non-negative time.)

This gives that solutions of the systems of ODEs u and $\tilde{u}$ are the same for any non-zero α ; hence, even if we have perfect observation of all the variables u_1 and u_2 from time 0 to infinity, we can identify the parameters at most up to multiplicative constants. This can be observed from the numerical experiment in Tables II and III that we have found two different sets of parameters with the same likelihood where all the parameters (CL , V_1 , V_2 and Q) differ by approximately a multiple of 6. Thus, if we only have observations from the fraction of the amount u_1 , parameters are structurally non-identifiable. On the other hand, if one has an observation on concentration u_1/V_1 , then V_1 may be identified; hence, the model parameters may be identifiable.

NONMEM Model Code

```

$PROB two compartment model simulation
$INPUT ID TIME DV AMT
$DATA simDatatwoComp IGNORE=@
$SUBROUTINES ADVAN3 TRANS4
$PK
TVCL = ABS(THETA(1))
TVV1 = ABS(THETA(2))
TVQ = ABS(THETA(3))
TVV2 = ABS(THETA(4))
CL = TVCL*EXP(ETA(1))
V1 = TVV1*EXP(ETA(2))
Q = TVQ
V2 = TVV2
S1 = V1
$ERROR
NPRED = A(1)
IPRED = A(1)
IRES = DV - IPRED
W = THETA(5)*A(1)
IWRES = IRES/W
Y = IPRED + EPS(1)*W
$THETA 3;1 CL
$THETA 5;2 V1
$THETA 15;3 Q
$THETA 10;4 V2
$THETA 0.1;5 prop error
$OMEGA BLOCK(2) 0.05 0.02 0.2;IIV CL, V1
$SIGMA 1 FIX
$EST METHOD = 1 INTER MAXEVAL = 9999
      FORMAT = s1PE23.16
$COVARIANCE UNCONDITIONAL PRINT = R

```

D. STEP SIZE OF THE FINITE DIFFERENCE DERIVATIVE APPROXIMATION

The Hessian (R-matrix) is often approximated with finite differencing of the computed gradient and the

Table V. Number of Successful Variance-Covariance Matrix Computations

	Default step size	SIGL = 6	SIGL = 8	SIGL = 10	SIGL = 12
Model 1	64	54	58	61	64
Model 2	69	81	79	76	63
Model 3	80	94	96	88	36

Out of 100 variance-covariance matrix computations from simulated data sets for each model. SIGL is the parameter that can be set to specify the step size of the finite differencing scheme used to compute the R-matrix in NONMEM

accuracy of the approximation depends on the choice of the finite difference step size. One can show that the optimal step size for the finite difference scheme is on the order of the square root of the computation error of the gradient for forward or backward differencing methods and the cubic root of the computation error of the gradient for the central differencing scheme (this can be found in standard textbooks in numerical analysis *e.g.* (17)). Most software provide ways to adjust the step size of the finite difference scheme (*e.g.* SIGL=in NONMEM); however, without quantification of the computational error of the gradient calculation, it is not possible to determine the optimal step size.

We have conducted a simulation study similar to experiment 1 (“**Experiment 1: Recover Failed Variance-Covariance Matrix Computation**”) to show that there is no general trend between the user-specified step size of the finite difference scheme and the rate of successful calculation of the covariance matrix (*cf.* Table V). Hence, case-by-case trial and error is required to find the appropriate step size so that the R-matrix will be neither numerically singular nor non-positive semi-definite so that the variance-covariance matrix can be calculated.

We have also investigated if we can obtain the correct indication of the non-estimability of the parameters of model 5 by changing the finite difference step size. As can be seen in Table VI, for all the step sizes, we have obtained small RSEs. As parameters of this model are shown to be not identifiable, all these calculated RSEs using different step sizes are clearly incorrect.

Table VI. Relative Standard Error (RSE) of the Estimated Parameter of Model 5 Deduced from Computed Variance-Covariance Matrix

	V1	CL	V2	Q	PRV
RSE with SIGL=6 (%)	10.6	6.90	4.70	6.17	2.67
RSE with SIGL=8 (%)	11.7	8.04	5.19	4.97	2.68
RSE with SIGL=10 (%)	32.9	30.5	27.8	24.3	2.68
RSE with SIGL=12 (%)	36.9	34.4	31.9	28.2	2.68

With all values of SIGL we have tried, the RSEs indicated all of the model parameters can be estimated with a reasonable level of uncertainty despite that most of the model parameters of model 5 are proven to be non-identifiable

Lastly, we would like to point out that preconditioning can be used together with properly adjusted finite step size to further increase the accuracy of the analysis.

REFERENCES

1. Benzi M. Preconditioning techniques for large linear systems: a survey. *J Comput Phys.* 2002;182(2):418–77.
2. Morikuni K, Hayami K. Convergence of inner-iteration GMRES methods for rank-deficient least squares problems. *SIAM J Matrix Anal Applic.* 2015;36(1):225–50.
3. Keizer RJ, Karlsson MO, Hooker AC. Modeling and simulation workbench for NONMEM: tutorial on Pirana, PsN, and Xpose. *CPT: Pharmacomet Syst Pharmacol.* 2013;2:e50.
4. Huber PJ. The behavior of maximum likelihood estimates under nonstandard conditions. *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability.* 1967;1(1).
5. Kauermann G, Carroll RJ. A note on the efficiency of sandwich covariance matrix estimation. *J Am Stat Assoc.* 2001;96(456):1387–96.
6. Wang Y. Derivation of various NONMEM estimation methods. *J Pharmacokinet Pharmacodyn.* 2007;34:575–93.
7. Karlsson MO, Hooker AC, Nordgren R, Harling K. PsN. 2015. [Online; accessed 16-April-2015]. <http://psn.sourceforge.net>.
8. Beal S, Sheiner L, Boeckmann A, Ludden T, Bachman W, Bauer R, *et al.* NONMEM 7.3; 1980–2013. <http://www.iconplc.com/innovation/solutions/nonmem/>.
9. Dosne AG, Bergstrand M, Karlsson MO. Application of sampling importance resampling to estimate parameter uncertainty distributions. In: Abstracts of the Annual Meeting of the Population Approach Group in Europe. Abstr 2907 [www.page-meeting.org/?abstract=2907]; 2013.
10. Dosne AG, Bergstrand M, Karlsson MO. Determination of appropriate settings in the assessment of parameter uncertainty distributions using Sampling Importance Resampling (SIR). In: Abstracts of the Annual Meeting of the Population Approach Group in Europe. Abstr 3546 [www.page-meeting.org/?abstract=3546]; 2015.
11. Bergmann TK, Brasch-Andersen C, Gréen H, Mirza M, Pedersen RS, Nielsen F, *et al.* Impact of CYP2C8*3 on paclitaxel clearance: a population pharmacokinetic and pharmacogenomic study in 93 patients with ovarian cancer. *Pharmacogenom J.* 2011;11:113–20.
12. Wahlby U, Thomson AH, Milligan PA, Karlsson MO. Models for time-varying covariates in population pharmacokinetic-pharmacodynamic analysis. *Br J Clin Pharmacol.* 2004;58:367–77.
13. Zingmark PH, Edenius C, Karlsson MO. Pharmacokinetic/pharmacodynamic models for the depletion of V β 5. 2/5.3 T cells by the monoclonal antibody ATM-027 in patients with

- multiple sclerosis, as measured by FACS. *Br J Clin Pharmacol.* 2004;58(4):378–89.
14. Frobel AK, Karlsson MO, Backman JT, Hoppu K, Qvist E, Seikku P, *et al.* A time-to-event model for acute rejections in paediatric renal transplant recipients treated with ciclosporin A. *Br J Clin Pharmacol.* 2013;76:603–15.
 15. Golub GH, Van Loan CF. Matrix computations. vol. 3. JHU Press; 2012.
 16. Brauer F, Nohel JA. Ordinary differential equations: a first course. WA Benjamin Advanced Book Program; 1973.
 17. Burden RL, Faires JD. Numerical analysis. USA: Brooks/Cole; 2001.