



# Implementation of non-linear mixed effects models defined by fractional differential equations

Christos Kaikousidis<sup>1</sup> · Aristides Dokoumetzidis<sup>1</sup>

Received: 5 December 2022 / Accepted: 20 February 2023 / Published online: 21 March 2023

© The Author(s) 2023

## Abstract

Fractional differential equations (FDEs), i.e. differential equations with derivatives of non-integer order, can describe certain experimental datasets more accurately than classic models and have found application in pharmacokinetics (PKs), but wider applicability has been hindered by the lack of appropriate software. In the present work an extension of NONMEM software is introduced, as a FORTRAN subroutine, that allows the definition of nonlinear mixed effects (NLME) models with FDEs. The new subroutine can handle arbitrary user defined linear and nonlinear models with multiple equations, and multiple doses and can be integrated in NONMEM workflows seamlessly, working well with third party packages. The performance of the subroutine in parameter estimation exercises, with simple linear and nonlinear (Michaelis–Menten) fractional PK models has been evaluated by simulations and an application to a real clinical dataset of diazepam is presented. In the simulation study, model parameters were estimated for each of 100 simulated datasets for the two models. The relative mean bias (RMB) and relative root mean square error (RRMSE) were calculated in order to assess the bias and precision of the methodology. In all cases both RMB and RRMSE were below 20% showing high accuracy and precision for the estimates. For the diazepam application the fractional model that best described the drug kinetics was a one-compartment linear model which had similar performance, according to diagnostic plots and Visual Predictive Check, to a three-compartment classic model, but including four less parameters than the latter. To the best of our knowledge, it is the first attempt to use FDE systems in an NLME framework, so the approach could be of interest to other disciplines apart from PKs.

**Keywords** Fractional differential equations · Anomalous kinetics · Numerical solver

## Introduction

Fractional calculus is a mathematical branch investigating the properties of derivatives and integrals of non-integer orders and goes back to the Leibniz's note in his letter to L'Hospital in which the meaning of the derivative of order one half is discussed [1]. Though for a long of time, fractional derivatives have existed only as theoretical mathematical objects, in the last few decades many authors

have pointed out that mathematical models containing non-integer order derivatives and integrals can more adequately describe properties and phenomena that govern real materials e.g. polymers. It has also been pointed out that fractional differential equations (FDEs) that arise from the use of such non-integer derivatives, can be used to describe anomalous diffusion (i.e. diffusion not described by Fick's law) and as a result, the anomalous kinetics that arise from it. In fact several experimental datasets have been described by fractional models better than by models of ordinary differential equations. One of the main features of FDE models, are the memory effects that arise by the fact that the value of a fractional derivative depends on a history of previous values and only on neighbouring ones [1].

In the field of pharmacokinetics (PKs), Dokoumetzidis and Macheras [2] introduced fractional calculus in 2009 to describe the PK of amiodarone that follows power law

✉ Aristides Dokoumetzidis  
adokoum@pharm.uoa.gr

Christos Kaikousidis  
chriskaik@pharm.uoa.gr

<sup>1</sup> Department of Pharmacy, National and Kapodistrian University of Athens, Panepistimiopolis, 15771 Athens, Greece

kinetics, and after that a few applications from the same and other authors followed, such as, to name a few, dealing with multicompartmental aspects of FDEs [3, 4], describing propofol PK [5], dealing with dose adjustment when powerlaw kinetics apply [6] and also extending the approach to pharmacodynamics, PDs [7]. Some applications have also appeared in neighbouring fields such as drug release kinetics [8] and skin drug absorption [9]. Recent applications of FDEs such as in the oral absorption of gentamicin [10] and bone remodeling [11] demonstrate that this is an active field of research. However, the number of fractional PK or PD models published is relatively limited, despite the fact that FDEs have attractive features and have the advantage that are extensions of classic ODEs which appear as special case when the order is set to 1. This should give fractional models important flexibility for empirical modelling even in cases where mechanistic description is not the main objective. One of the reasons for the limited uptake of fractional calculus in pharmacometrics literature is the lack of appropriate software: (a) efficient numerical solvers and (b) integration in parameter estimation software. A particular class of models of high importance in pharmacometrics is nonlinear mixed effects models (NLMEs) where variability components are also estimated for each parameter, typically by analysing longitudinal data (such as PK data) from many subjects and therefore characterising the interindividual variability of the parameters. For NLME models complex algorithms are used with various strategies for the calculation or approximation of the objective function and its minimization to the optimal parameter solution, and very rarely do scientists use homemade code for such a task. As a result, a few specialized software packages, mainly commercial and some open source, dominate all the NLME applications, with the leading one being NONMEM (Icon plc), which was the first one that appeared in the seventies and has continued to be the industry standard since then. It is therefore of particular importance to implement a NONMEM extension for FDEs that readily allows the description of fractional models.

In the present paper an extension of NONMEM is introduced that allows the definition of models including FDEs. The extension allows the user to try out FDEs in a model development workflow much like any other model using the same likelihood approximation and minimisation algorithms. Furthermore an implementation of an efficient numerical solver has been carried out, after extensive evaluation of various numerical solvers that appear in literature, a few of which are reviewed here too. The performance of the extension in parameter estimation exercises with linear and nonlinear fractional models has been evaluated by simulations and an application in a real clinical dataset of diazepam is presented. For completeness

a brief introductory theoretical section on fractional calculus is also included.

## Fractional calculus

### Theory

The basis of fractional calculus is the definition of a fractional order of differentiation  $\alpha$  - in our case  $0 < \alpha < 1$ . This  $\alpha$ th derivative is defined through fractional integration and successive ordinary differentiation. Hence firstly a fractional integral formula is defined through the Riemann–Liouville (LR) integral [1]

$${}^R L I_t^\alpha f(t) = {}_0 D_t^{-\alpha} f(t) = \frac{1}{\Gamma(\alpha)} \int_0^t (t - \tau)^{\alpha-1} f(\tau) d\tau, \quad \alpha > 0, \quad (1)$$

where  $\Gamma$  is the gamma function. Here  $f$  is assumed such that the involved integral is well defined. The left-side subscript of the  ${}^R L I_t^\alpha$  and  ${}_0 D_t^{-\alpha}$  operators, denotes the lower end of the integration limits, which in this case has been assumed to be zero. Different bounds can be used but result in slightly different definitions and properties. Using ordinary differentiation of the above integral, the fractional derivative is defined through

$${}_0 D_t^\alpha f(t) := D^1 {}_0 D_t^{\alpha-1} f(t) = D^1 {}^R L I_t^{1-\alpha} f(t), \quad (2)$$

where  $D^1 = \frac{d}{dt}$ , the ordinary derivative. The calculations above yield the following definition for the LR fractional derivative [1]

$${}_0 D_t^\alpha f(t) = \frac{d}{dt} \left[ \frac{1}{\Gamma(\alpha-1)} \int_0^t \frac{f(\tau)}{(t-\tau)^{2-\alpha}} d\tau \right], \quad 0 < \alpha < 1. \quad (3)$$

This definition is a convolution between the function  $f$  and a power law function of time and is the source of the memory effects that arise through the process.

Though widely known, the LR definition of fractional derivatives is not the most useful one when it comes to real applications. That is because when used in differential equations, for  $t = 0$  a fractional integral of the function  $f$  is involved in the initial condition which is not easy to interpret physically. That is why another definition is mostly used when dealing with FDEs, and that is the Caputo derivative which is defined as

$${}^C D_t^\alpha f(t) := \frac{1}{\Gamma(1-\alpha)} \int_0^t \frac{f'(\tau)}{(t-\tau)^\alpha} d\tau, \quad (0 < \alpha < 1), \quad (4)$$

where the  $C$  in the upper index indicates the Caputo definition of the fractional derivative. With this definition, the

initial value problems for fractional problems can be written in the familiar form

$${}^C_0D_t^\alpha y(t) = f(t, y), \quad y(0) = y_0. \tag{5}$$

Another advantage this definition has is that it carries properties that are consistent to the ordinary derivative such as that the fractional derivative of a constant is zero  ${}^C_0D_t^\alpha C = 0$  which is not the case for LR. For these reasons in the rest of this paper, the fractional derivatives used are assumed to be Caputo derivatives.

A simple example of an FDE from PKs is the classic one-compartment model with first order elimination

$$\frac{dA}{dt} = -k_e A, \quad A(0) = A_0. \tag{6}$$

The fractional equivalent of this equation is

$${}^C_0D_t^\alpha A(t) = -k_e A(t), \quad A(0) = A_0 \tag{7}$$

and its solution can be expressed analytically through the Mittag–Leffler function [1]

$$A(t) = A_0 E_\alpha(-k_e t^\alpha), \tag{8}$$

where

$$E_\alpha(x) = \sum_{k=1}^{\infty} \frac{x^k}{\Gamma(\alpha \cdot k + 1)}. \tag{9}$$

This is a generalised form of the exponential function since for  $\alpha = 1$  we see that  $E_1(x) = \exp(x)$ . This means that the fractional version of the simple PK model collapses to the ordinary one for  $\alpha = 1$ ,  $A(t) = A_0 e^{-k_e t}$ . In [12] it is shown that (8) behaves as a stretched exponential for small times i.e.  $\sim \exp(-k_e t^\alpha)$  while for larger times behaves like a power law. In general the solution of this fractional equation can describe anomalous processes or drug release in heterogeneous media [12].

In actual applications few problems of the form of (5) have a closed-form analytical solution. In fact even (7) which does have an analytical solution, still needs to be computed numerically in order to approximate the Mittag–Leffler function efficiently, since the analytical definition involves an infinite power series which is more difficult to compute. Hence the necessity of numerical methods which can approximate the trajectories of such fractional systems arises naturally. Numerical solvers for FDEs must be efficient both in terms of accuracy as well as speed. In the next section a few of these methods are presented with emphasis on the Grünwald–Letnikov (GL) scheme [13] that was the one chosen for the implementation in NONMEM and was used to produce the results presented in this paper.

## Numerical methods

### The Laplace transform

A function  $f(t)$  is of exponential order  $a$  when we can find positive constants  $M$  and  $T$  such that  $e^{-at}|f(t)| \leq M$  for all  $t > T$  so  $f(t)$  does not grow faster than an exponential function for  $t \rightarrow \infty$ . The Laplace transform for such a function is defined [1]

$$F(s) = \mathcal{L}[f(t); s] := \int_0^\infty e^{-st} f(t) dt. \tag{10}$$

The inverse Laplace transform can also be defined, though it is not possible to write it down in most cases. The inverse transform gives the original function

$$\mathcal{L}^{-1}[F(s); t] := \int_{c-i\infty}^{c+i\infty} e^{st} F(s) ds = f(t), \quad c = \text{Re}(s) > c_0, \tag{11}$$

where  $c$  is the path of integration. In classic calculus, the Laplace transform is a useful technique for solving linear ordinary differential equations. In the Laplace domain one can transform ODEs to algebraic expressions which are much easier to solve, and then go back to the time domain by performing the inverse Laplace transform. Fractional linear differential equations can also be written in the Laplace domain easily since—like ordinary derivatives—fractional derivatives can be transformed in the Laplace domain. For  $0 < \alpha \leq 1$  it can be shown that [1]

$$\mathcal{L} [ {}^C_0D_t^\alpha f(t); s ] = s^\alpha F(s) - s^{\alpha-1} f(0). \tag{12}$$

A simple example of an FDE that can be solved using the Laplace transform is found in [3]

$${}^C_0D_t^{1/2} y(t) = -y(t) \tag{13}$$

with  $y(0) = 1$ . Using (12) it can be written as

$$s^{1/2} Y(s) - s^{-1/2} y(0) = -Y(s), \tag{14}$$

$y$  where  $Y(s)$  is the Laplace transform of  $y(t)$ . Using standard algebra it gives the following result in the Laplace domain

$$Y(s) = \frac{1}{s + \sqrt{s}}. \tag{15}$$

The final solution can now be obtained by performing an inverse Laplace transform. For this simple example, there are expressions that can be found in tables with inverse Laplace transform formulas such as [1] which yield the known result

$$y(t) = E_{1/2}(-t^{1/2}). \tag{16}$$

Unlike the previous example, solving (11) explicitly for the system of variables and obtaining an analytical solution for  $Q(s)$  in the time domain, is generally difficult—if not impossible. To address this problem algorithms to carry out numerical inverse Laplace transform (NILT) exist. Several such algorithms have been developed such as [14] which is based on the Euler algorithm [15]. Another popular choice is the Talbot method [16] where the path of integration is chosen to be a special cotangent contour [17]. These algorithms are a useful tool for solving linear FDEs and have been used successfully in fractional PKs estimation problems [3], however they are typically very demanding computationally. This makes them unattractive for estimation problems where the speed of the algorithm is of paramount importance.

**Fractional linear multi-step methods**

An alternative to the various NILT methods that provides more efficiency as well as universality, is the family of methods known as fractional linear multi-step methods (FLMMs). These are extensions of the classic Linear Multi-Step methods used for numerically solving classic ODEs. The name multi-step stems from the fact that they use the solution at several steps previous to the current step for which the solution is computed. The idea behind these methods is that an initial value problem

$$y'(t) = f(t, y) \quad y(0) = y_0 \tag{17}$$

can be written as

$$y(t) = y_0 + \int_{t_0}^t f(t, y) dt \tag{18}$$

which can be approximated by [13]

$$\sum_{j=0}^k \rho_j y_{n-j} = h \sum_{j=0}^k \sigma_j f(t_{n-j}, y_{n-j}), \tag{19}$$

where  $\rho(z) = \rho_0 z^k + \rho_1 z^{k-1} + \dots + \rho_k$  and  $\sigma(z) = \sigma_0 z^k + \sigma_1 z^{k-1} + \dots + \sigma_k$  are the first and second characteristic polynomials. The fraction of these two polynomials give the generating function of the LMM

$$\delta(\xi) = \frac{\rho(1/\xi)}{\sigma(1/\xi)}, \tag{20}$$

where meaning of  $\delta(\xi)$  is that the weights  $\omega_n$  can be written as the coefficients of the power series of the generating function. Depending on whether  $\rho_0 \neq 0$  and  $\sigma_0 \neq 0$ , the method is either explicit [ $y_n$  appears only on the left side of (19)] or implicit ( $y_n$  appears on both sides). Finally this can be reformulated in terms of convolution quadrature formulas in:

$$y_n = h \sum_{j=0}^n \omega_{n-j} f(t_j). \tag{21}$$

This method is a special case of a general methodology used for solving problems of type (18) or otherwise known as Volterra integral equation of the second kind [18]. The extension to FDEs of the general LMMs is based on the fact that the general problem (5) can also be written as a Volterra integral equation simply by the fractional integral operator  $({}_0I_t^\alpha)$  on both sides and obtaining

$$y_n = y_0 + \frac{1}{\Gamma(\alpha)} \sum_{j=1}^n \int_{t_{j-1}}^{t_j} (t_n - \tau)^{\alpha-1} f(\tau, y(\tau)) d\tau. \tag{22}$$

The details on deriving the fractional equivalent of (21) can be found in [19]. There, the final formula for the FLMMs is derived

$$y_n = y_0 + h^\alpha \sum_0^s w_{n,j} f(t_j, y_j) + h^\alpha \sum_0^n \omega_{n-j}^{(\alpha)} f(t_j, y_j), \quad \alpha \leq 1 \tag{23}$$

and  $w_{n,j}$  are correction terms which maintain the order of convergence when  $f$  is non-smooth [19]

In this work, the method that was used for the numerical solution of the FDEs was the so called GL scheme based on (23). It is basically a backwards differentiation formula of first order (BDF1). The correction terms are omitted and using the classic BDF1 where the generating function takes the form  $\delta(\xi) = 1 - \xi$ , it can be shown [18] that the  $\omega$  coefficients are those of a binomial series. Hence

$$\omega_n^{(\alpha)} = (-1)^n \binom{-\alpha}{n} = (-1)^n \frac{\Gamma(1 - \alpha)}{\Gamma(n + 1)\Gamma(-\alpha - n + 1)} \tag{24}$$

and so the formula for  $y_j$  is

$$y_n = y_0 + h^\alpha \sum_{j=0}^n (-1)^{n-j} \binom{-\alpha}{n-j} f(t_j, y_j), \quad \alpha \leq 1, \tag{25}$$

where  $y_0 = y(0)$ . Since this method is implicit there is need for an approximation of  $y_n$  to be used in  $f(t_j, y_j)$  for  $j = n$ . In this work, an iterative Newton–Raphson method was used for this purpose. Given an initial approximation for  $y_n$ —let this be  $y_n^{(0)}$ —the method can approximate  $y_n$  using the following relation

$$y_n^{(k)} = y_n^{(k-1)} - \left[ 1 - h^\alpha f'(t_n, y_n^{(k-1)}) \right]^{-1} \times \left[ y_n^{(k-1)} - h^\alpha f(t_n, y_n^{(k-1)}) - \sum_{j=1}^{n-1} \omega_{n-j}^{(\alpha)} f(t_j, y_j) - y_0 \right], \tag{26}$$

where  $k \in \mathbb{N}$  is the number of iterations. This method is highly popular because of its quadratic convergence [20]. Since its convergence properties are local, a good initial estimate  $y_n^0$  is required for the method to perform well. A good approximation for it is usually  $y_n^{(0)} = y_{n-1}$ . In rapidly changing solutions (i.e. non-smooth functions) a smaller step is required for this approximation to provide accurate results.

### Impulsive fractional differential equation

A particular class of problems that is of interest in PK and PK/PD applications, are the impulsive differential equations which are described by

$$\begin{aligned} {}^C_0 D_t^\alpha y(t) &= f(t, y(t)), \quad t \in J' := J \setminus [t_1, \dots, t_m], \quad J := [0, T] \\ y(t_k^+) &= y(t_k^-) + u_k, \quad k = 1, 2, \dots, m, \quad y(0) = y_0 \end{aligned} \quad (27)$$

which means that for every time point  $t_k$  there is an impulse  $u_k$  that mainly describes physical phenomena that have a sudden change in their states, i.e. an additional dose, where the total number of doses is  $m$ . These types of equations are therefore able to describe multiple dose kinetics. In the classic case ( $\alpha = 1$ ), the method of solving these problems is trivial, since one can simply solve ODEs for time intervals between doses and then for  $t_k$  restart the solver with initial conditions containing the extra dose. For ( $\alpha < 1$ ) that is not the case since as mentioned, each  $y_i$  depends on the solution of each previous time point, therefore restarting the solver for each dose, would result in loss of that information. For the numerical solution of this class of problems, we follow [21] that suggests that the solution of (27) is

$$\begin{aligned} y(t) &= y_0 + \sum_{i=1}^k u_k + \frac{1}{\Gamma(\alpha)} \int_0^t (t-s)^{\alpha-1} f(s, y(s)) ds, \\ \text{for } t &\in (t_k, t_{k+1}). \end{aligned} \quad (28)$$

This problem is identical in its numerical solution to (22) but for the addition of the summation term of the doses  $u_k$  which is implemented in the existing algorithm with a simple *if* statement which adds each dose to each  $y_n$  for  $t_n > t_k$ .

## Methods

### Numerical solver for fractional differential equations

The GL scheme was the method chosen for the solution of the FDEs that described the models tested in NONMEM. The algorithm was picked for its efficiency in terms of speed and accuracy compared to other algorithms. More specifically, the Adams–Bashforth–Moulton predictor–corrector (ABMPC) method [12] which was also tested, proved both slower and less accurate for small  $t$  than the GL algorithm. The nature of the ABMPC algorithm is such that more computations accumulate in the process thus increasing computational time. Also for small times there were terms that needed small steps  $h$  in order to converge which was time costing by itself. However it is worth mentioning that this algorithm was robust and efforts have been made to reduce computational costs [22], making it a possible option for use on more complex problems.

In order to validate the algorithm upon which the numerical solver was based, various tests were performed. The rationale was to compare the numerical solution preferably to an analytical solution, so for cases where an analytical solution was not available such as non-linear models comparison for the classic case of  $\alpha = 1$  was used. First, the numerical solution was computed for a linear fractional PK model like (7) and compared to the analytical solution given by (8). In order to verify that the algorithm gives reliable results for non-linear models as well, it was tested on a fractional Michaelis–Menten equivalent model (29), by setting  $\alpha = 1$  and comparing the results to the classic Michaelis–Menten solution, since neither an analytical solution nor a NILT solution is available for this case. The capability of the algorithm to handle multiple FDEs were tested by simulating a classic two-compartment PK model with  $\alpha = 1$  and compared to the analytical solution. Finally the capability of the algorithm to handle multiple doses was tested with the linear fractional model of Eq. (7) compared to the superposition of multiple instances of the analytical solution of Eq. (9). Indeed, in linear FDEs, superposition principle holds, but the multiple dose problem in the nonlinear case, is nontrivial due the memory effects. All the above mentioned algorithms were developed and tested in MATLAB before final implementation in FORTRAN. The GL algorithm was optimized for speed and accuracy.

### NONMEM implementation and interface

The numerical routine for the solution of FDEs was developed and tested in MATLAB, and ported to FORTRAN to make it

available as a subroutine in NONMEM. It supports linear and nonlinear models, with multiple FDEs, as well as multiple doses. The implementation of the fractional models in NONMEM is done using \$PRED control record in the control file in order to provide a user supplied subroutine for the solution of the model equations. These are not defined in the control file like in the \$DES case but in the subroutine itself. The subroutine FDEGL(. . .) can be called anywhere in \$PRED using a brief verbatim code ``CALL FDEGL(VECTRA, VECTRIB, TIME, ff, AMT, int(Np), int(Ns), int(Nd)), given that all the arguments used, are defined. The vector names VECTRA, VECTRIB are used in order to be compliant with what the NONMEM abbreviated code supports. Arguments are: All model parameters including the order of the fractional derivative  $\alpha$  are passed in VECTRA; VECTRIB stores the initial values of the dependent variables; TIME is the value of the NONMEM data record for which the solution is to be computed; AMT is the data record needed for each dose, as usually used in NONMEM; Np the number of the parameters, Ns is the number of steps for the solver and Nd the number of FDEs. Once called, it returns the solution of the equation, ff, for each time point requested by NONMEM (i.e. the TIME value for each data record).

The FORTRAN subroutine consists of two parts: the main solver algorithm and the definition of the FDE. The first part is fixed and is not supposed to be changed by the user. The second part of the subroutine is the definition of the problem and consists of two FORTRAN functions.

In the first FORTRAN function, FFUN, the user has to input right hand side of the FDE (5) exactly as with all ODE solving software. In the second FORTRAN function, FFUNJ, the user inputs the derivative  $\partial f(t, y)/\partial y$  in the same manner. For example, in the simple system of (7), the first function is FFUN=-ke\*y, while the second function is FFUNJ=-ke, which is the Jacobian of the right hand side of (7).

This—apart from the number of steps of the solution—is the only user input in the process, which makes the use of the subroutine easy and familiar thus requiring little or no knowledge of fractional calculus on behalf of the user. Also note that doses are handled naturally with in the data file in the AMT data item, as usual.

The FORTRAN subroutine FDEGL together with a template NONMEM control stream, as well as user instructions, can be found in a GitHub page (<https://github.com/PMXathens/FDE4NONMEM>). All work in this paper was carried out using NONMEM version 7.4 compiled with gfortran version 4.6.0.

## Evaluation by simulation study

Several tests on single patient runs were performed to make sure the algorithm was called properly by NM-TRAN. Afterwards one hundred data sets were simulated for each

of the two models studied and estimation of the population parameters was carried out for each one of them in order to calculate the bias and precision of the methodology, by the relative mean bias (RMB) and relative root mean square error (RRMSE), respectively. In both models—linear and non-linear—a population of 24 subjects was simulated for  $t = 40$  h and both fixed and random effects were considered. The batch estimation process was achieved using the Bootstrap tool of Perl Speaks NONMEM (PSN) with data sets simulated in MATLAB. Namely, the datasets of the simulation study prepared in MATLAB, were placed in the bootstrap directory of a PSN bootstrap run, as bootstrap datasets and then the bootstrap command was run to execute them.

## Linear model

The linear model is given by the expression (7) with an additional parameter volume of distribution,  $V$ , as the proportionality constant between the drug amount  $A$  and the measured concentration  $C$ , as  $C(t) = A(t)/V$ . Variability was assumed in  $k_e$  and  $V$ , whereas  $\alpha$  was kept the same for all subjects. A log-normal distribution was considered for the two random variables and a proportional error model for the residual variability. More specifically, the  $\theta$  values were:  $\theta_{k_e} = 0.5 h^{-\alpha}$ ,  $\theta_V = 6.5$  L and  $\theta_\alpha = 0.5$  while for the random distributions:  $\omega_{k_e} = 0.25$ ,  $\omega_V = 0.25$  for the interindividual variabilities and  $\sigma = 0.1$  for the proportional residual error and a dose of 9 mg. For the estimation process of each data set, the stochastic approximation expectation maximization (SAEM) method with INTERACTION was used since it proved the most robust of all other estimation methods for this problem, while the standard errors were estimated by the importance sampling (IMP) method.

## Non-linear model

As a second case study, a non linear model was chosen in order to show that the methods can be used in a wider category of problems. A fractional analog of Michaelis–Menten kinetic was used, described by the following equation

$${}^c_0D_t^\alpha A(t) = -\frac{V_{max}C(t)}{K_m + C(t)} \quad (29)$$

and  $A(0) = dose$ ,  $C(t) = A(t)/V$ , where  $K_m$  and  $V_{max}$  are the Michaelis–Menten constant and the maximum rate of elimination, respectively. In the population study, log-normal variability was considered in  $V_{max}$ ,  $K_m$  and  $V$  whereas  $\alpha$  was kept the same for all patients. The simulation parameters were:  $\theta_{V_{max}} = 3.5$  mg/h<sup>2</sup>,  $\theta_{K_m} = 0.6$  mg/L,  $\theta_V = 2$  L,  $\theta_\alpha = 0.9$  with  $\omega_{V_{max}} = 0.25$ ,  $\omega_V = 0.25$ ,  $\omega_{K_m} =$

0.25 and a dose of 9 mg. Once again a proportional error model was used with  $\sigma = 0.1$  and the SAEM method with INTERACTION was chosen for the estimation process.

### Diazepam data analysis

For the application on a real data set, Diazepam data from [23] were used. The data set used consisted of 12 patients for  $t = 72$  h for an iv dose of 5–10 mg. A fractional one-compartment model was used for the fitting process, initially using three parameters: volume of distribution ( $V$ ), elimination rate ( $k_e$ ) and the order of the fractional derivative ( $\alpha$ ). For comparison purposes, classic (non-fractional) 1, 2 and 3-compartment models with linear elimination were also fitted to the data. The comparison of the fits was done using diagnostic plots such as predicted vs observed and residuals vs time as well as Visual Predictive Check (VPC) plots using PSN.

## Results

### Numerical results

In order to test the numerical solvers used for the solution of FDEs the GL scheme was compared to known methods. The results are shown in Fig. 1 with the compared solutions completely overlapping, demonstrating that the GL method is successful in solving both the linear and non-linear problems, while it can handle successfully multiple equations and multiple doses. Other than its accuracy, the method was efficient in terms of speed. More specifically, for the linear example of (7), the numerical solution was compared to the analytical solution (8) (Fig. 1A), while for the non-linear equation (29) a comparison was made for  $\alpha = 1$  i.e. the classic Michaelis–Menten equation, using a typical MATLAB solver (*ode15s*) (Fig. 1B). The multi-compartment case was tested also for the classic  $\alpha = 1$  against the solution with the same MATLAB solver (Fig. 1C). The main aspect tested in the two latter cases is the ability of the algorithm to give accurate results and it is thought that since in the classic case the GL method gives correct results, it is unlikely that it could produce a wrong solution when  $\alpha < 1$ . In the last case the multiple dose scenario was tested for the linear model for  $\alpha = 0.6$ , against the superposition of the respective analytical solution. While the superposition holds in linear FDEs and is a good approach to handle multiple doses, in the nonlinear case multiple doses need to be handled within the algorithm, while restarting the solution using as initial values the

final point of the previous dose session, which is the common practice for ODEs, is not feasible in FDEs due to the memory effects and the presence of history. Figure 1D shows that multiple doses can be handled well within the GL solver for the fractional linear case and there is no reason to believe that this could change for the non-linear case, despite the fact that there is no analytical solution for the latter case to test it against.

### Simulation study

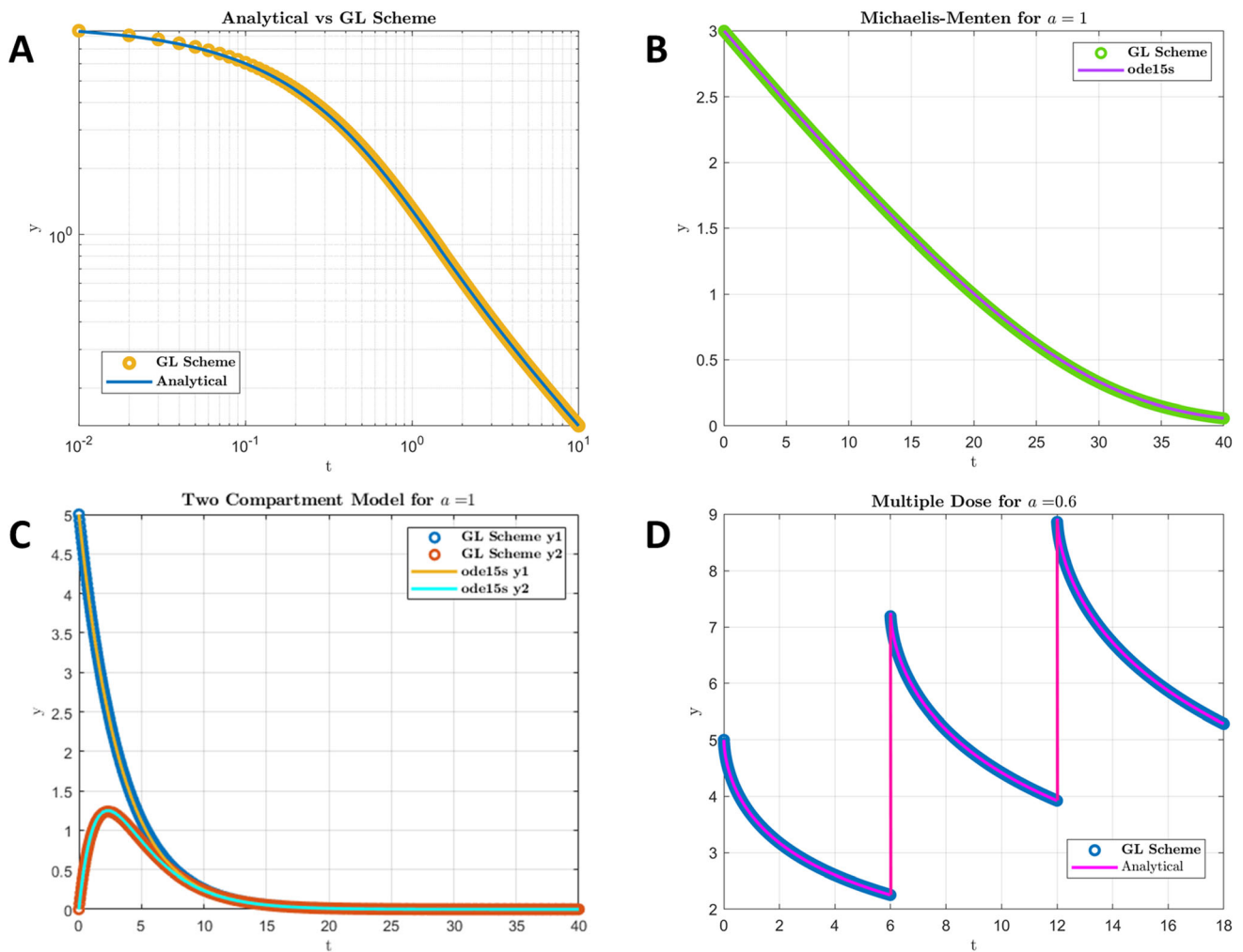
The NONMEM implementation was evaluated by a simulation/estimation study for the two models (7) and (29). The first is the simplest PK relationship, the iv bolus with linear elimination, which in its fractional version can account for various diffusion processes that are anomalous or of slower diffusion in the deeper tissues. The second model is the fractional version of a typical non-linear Michaelis–Menten model which is usually used to describe saturable PK processes. While its physical meaning is unclear at the moment, here it is used as a typical non-linear model to test the performance of the method.

Parameters were estimated for each of the one hundred simulated data sets for the two models. The RMB and RRMSE were calculated using the results from each estimation in order to assess the bias and precision of the methodology. The results for the linear and non-linear models are shown in Tables 1 and 2, respectively. For the linear model, RMB was below or near 1% while RRMSE below 10% for the THETAs and in the region of 15% for the OMEGAs and the SIGMA. In the non-linear model more elevated values for  $V_{max}$  and  $K_m$  were obtained for RMB, 5% and 11%, respectively, while for RRMSE the highest value was obtained for  $K_m$  and was 19%. Overall, the results show high accuracy and precision for the estimates. Also the expected values for each parameter, i.e. the mean of the one hundred estimates, are very close to the simulated values, all of which indicate a good performance in the simulation study.

The performance of NONMEM while using the user function for the solution of the fractional equations, was satisfying since it did not produce unusual errors and warnings while the speed of the optimization was found to be relative to the speed of the solver.

### Diazepam data analysis

The fractional model that best described the drug kinetics was a one-compartment linear model with an elimination rate  $k_e$  and fractional order  $\alpha$ , i.e. (7). The volume of distribution  $V$  was not estimated but set to 1 since there was



**Fig. 1** Typical profiles generated with GL scheme compared to various alternative methods. **A** GL scheme vs. Analytical ML function for the linear model. **B** GL scheme vs. MATLAB ode15s for classic (non-fractional) Michaelis–Menten model. **C** GL

scheme vs. MATLAB ode15s for classic (non-fractional) two-compartment model. **D** GL scheme for multiple dose vs. superposition of analytical ML functions

**Table 1** Results of the simulation study for the Linear Model

	$\theta_{k_e} (h^{-\alpha})$	$\theta_V (L)$	$\theta_x$	$\omega_{k_e}$	$\omega_V$	$\sigma$
RRMSE (%)	9.14	0.95	2.72	16.53	12.03	14.62
RMB (%)	−1.017	0.322	0.393	0.041	−0.611	−0.036
Sim. values	0.5	6.5	0.5	0.25	0.25	0.1
Exp. values	0.506	6.251	0.501	0.259	0.25	0.099

For each model, all 100 estimation runs were successful and all results were used for the calculation of RMB and RRMSE

an identifiability issue. This is to be expected since, as already mentioned, the analytical solution of this equation is a Mittag–Leffler function given by (9). This function has two main trends: For small values of  $t$  it has the form of a stretched exponential function, while for large values of  $t$  it

has the form of a power law. The analytical form of this power law function is given by the following expression [24]

$$y(t) = \frac{t^{-\alpha}}{\Gamma(1-\alpha)} \frac{V}{k_e} \tag{30}$$

For the case of Diazepam, data were available for 72 h, while for initial times there was limited information, hence the solution was required for large times, following (30). This means that we expect only the ratio  $V/k_e$  to be estimated properly since the solution is the same for any combination of  $V$  and  $k_e$  that results in the same ratio. Thus in NONMEM only a  $k_e$  parameter was estimated and the volume of distribution was arbitrarily set to 1.

The estimation process was performed in NONMEM while the most successful method was SAEM with INTERACTION. Standard errors were calculated in



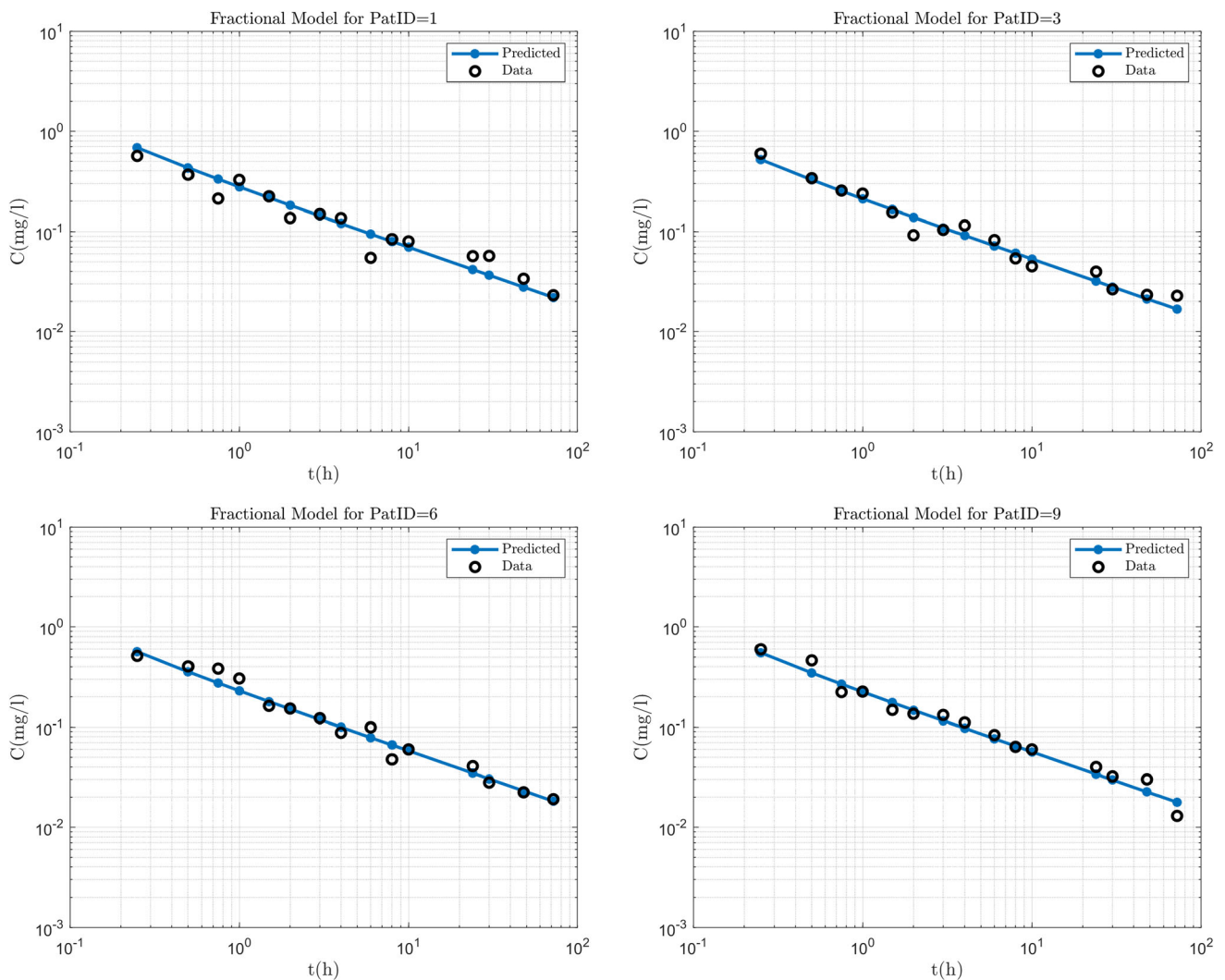
**Table 2** Results of the simulation study for the Non-Linear Model

	$\theta_{V_{max}}$ (mg/h <sup>2</sup> )	$\theta_{K_m}$ (mg/h <sup>2</sup> )	$\theta_V$	$\theta_x$	$\omega_{V_{max}}$	$\omega_{K_m}$	$\omega_V$	$\sigma$
RRMSE (%)	2.94	19.13	2.82	0.7	7.2	11.11	8.21	12.06
RMB (%)	-4.98	-11.03	-2.36	-0.62	-0.63	-1.54	0.12	-0.12
Sim. values	3.5	0.6	2	0.9	0.25	0.25	0.25	0.1
Exp. values	3.422	0.55	1.998	0.893	0.249	0.244	0.25	0.099

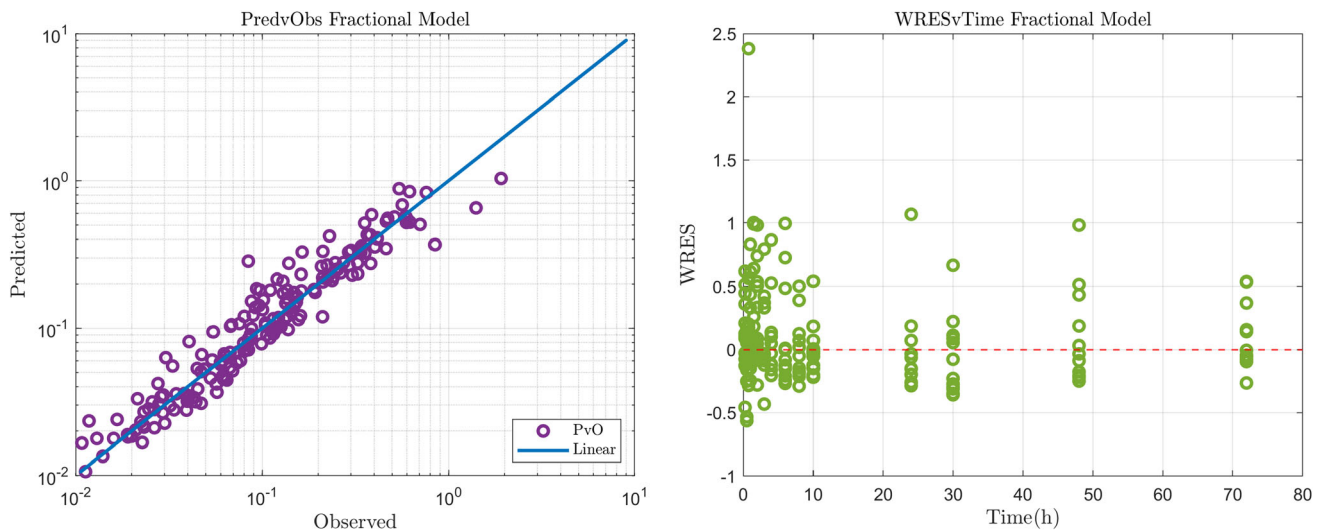
For each model, all 100 estimation runs were successful and all results were used for the calculation of RMB and RRMSE

NONMEM by the IMP method. The results of the NLME model constructed to fit the Diazepam data, were inspected visually and with the use of diagnostic plots. Predicted vs. Observed, Residuals vs. Time and VPC graphs were used to determine the goodness of fit as well as the ability of the model to predict the data. Some individual fits obtained by NONMEM can be seen in Fig. 2. The results are presented in log–log plots in order to assess the fit more accurately

for the small concentration values. As shown by the plots, the Diazepam data follow a power law shape. The predicted versus observed and residual versus time graphs are presented in Fig. 3 showing all 12 patients together with the identity line and show good agreement between the predictions and the observations. The residual versus time plot shows that the points are randomly distributed evenly around the x axis which provides evidence that the



**Fig. 2** Individual fits for 4 out of the 12 patients, showing the power law behaviour of the data



**Fig. 3** Predicted vs. Observed (left) and Residual versus Time (right) graphs

estimator is not biased and also that the predicted points follow the same trend as the observed points.

In Table 3 together the parameter estimates, the standard errors of the estimates are shown, which take small values indicating that the estimator is precise.

Finally the ability of the model to predict the data was evaluated with the use of a VPC, which also tests whether misspecification is present in the structural, variability or error models. For this procedure, 500 data sets of the same format as the observed data set were simulated and 95% prediction intervals around each of the 5th, 50th, and 95th percentiles of the simulated data sets were plotted together with the corresponding percentiles of the observed data. The results appear in Fig. 4 and show that the observed percentiles fall within the prediction intervals of the simulated data. The VPC plot was generated in PSN which demonstrates that companion third party software packages such as PSN work well with the present FDE NONMEM extension, without modifications.

The fractional model was also compared to a classic compartment model. An optimization was performed using the same data set for Diazepam with a classic three-compartment model with iv bolus administration. The fit for the same patients as for the fractional model is presented for the classic case in Fig. 5. The fit with the three-

compartment model is comparable to the fractional model but using six parameters as opposed to only two of the fractional model.

## Discussion

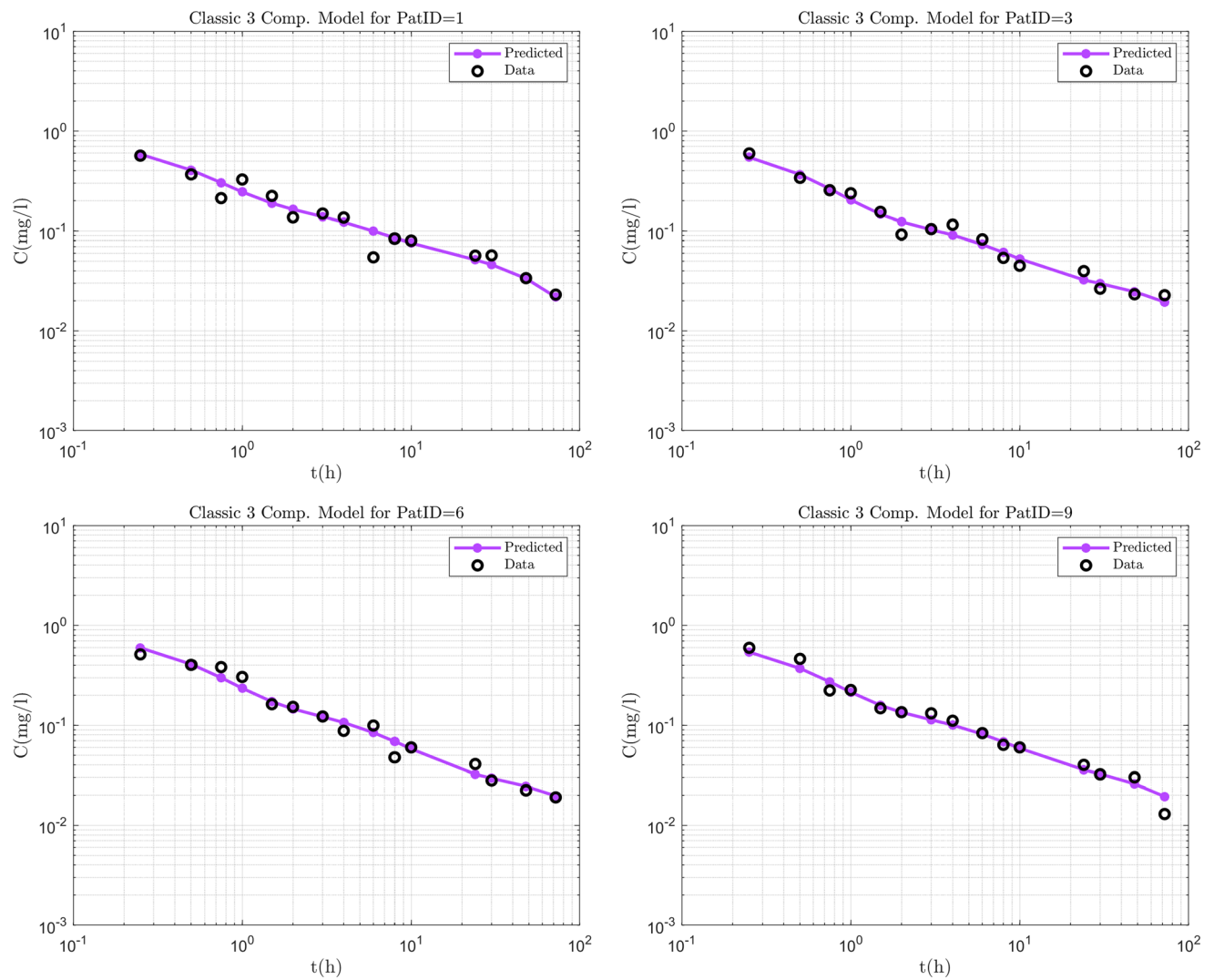
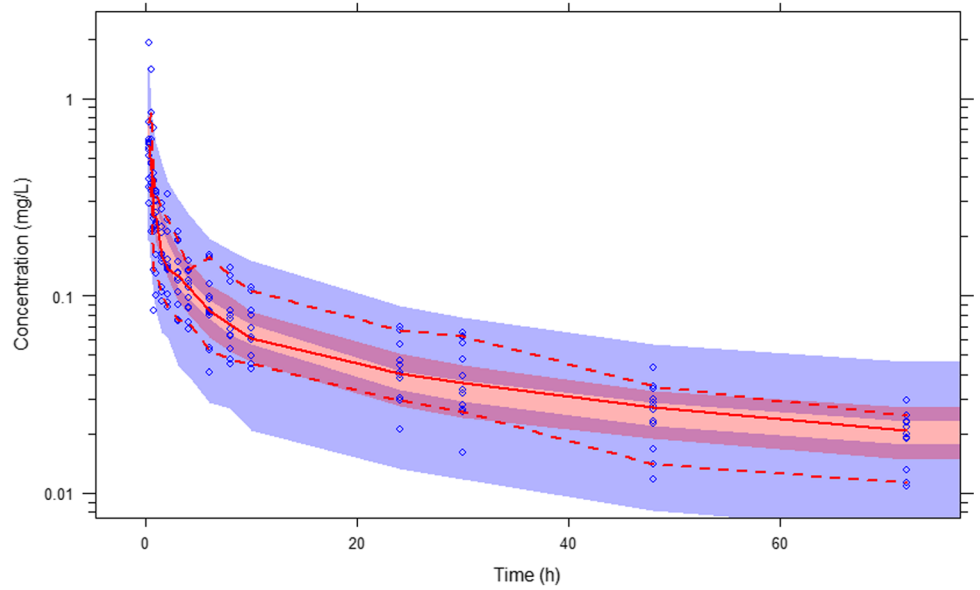
The main deliverable of the present work is a general purpose subroutine, written in FORTRAN, which works as a plugin to NONMEM and allows the definition of arbitrary, user-defined, linear and nonlinear FDE models. The actual numerical algorithm used for the FDE solver is based on the GL method [19], but has been modified to suit our needs. Furthermore the final choice of the numerical method is a result of an extensive survey and testing of many different algorithms in terms of performance and flexibility. To evaluate the approach, we presented a couple of simulation–estimation exercises for linear and nonlinear systems, respectively, and an application to a real PK clinical dataset.

Although successful in this study, this is a first approach of the use of fractional models in population analysis and the method still needs to be tested on more complex problems, where stiff differential equations or larger data sets appear. Also the method has limitations. At the moment only IV bolus dosing is supported, with multiple doses too. Other routes of administration or infusion would need adding to the model depot compartments which is not trivial. Also at the moment only FDEs of the form of Eq. (5) are supported, i.e. with the fractional derivative on the left hand side. More general FDEs such as those proposed in [3] are not supported. Despite the fact that the examples shown use a single FDE, the subroutine provided can be used for multiple FDEs. However, it is worth

**Table 3** Parameter estimates for the diazepam data analysis

	$\theta_{k_e}$	$\theta_\alpha$	$\omega_{k_e}$	$\sigma$
Estimated values	17.1	0.54	0.1	0.083
Standard errors	2.04	0.017	0.0512	0.0084
CV (%)	11.92	3.15	51.2	10.1

**Fig. 4** VPC for Fractional model



**Fig. 5** Individual fits for 4 out of the 12 patients, for a classic 3 compartment model with absorption for comparison

mentioning that regarding the multi compartment analysis, one needs to take precautions when using FDEs in PK problems. As discussed in [4], if the fractional derivatives used in each equation have different orders  $\alpha_1, \alpha_2$  etc., i.e. the non-commensurate case, then mass balance issues arise. This happens because usually the outgoing flux from one compartment is an incoming flux in another. If different orders are used, then the transfer rate constants will have different units (i.e.  $h^{-\alpha_1}, h^{-\alpha_2}$ , etc.) and mass balance will not hold. Fractional multi-compartmental models can be useful if the same order  $\alpha$  is used (commensurate case), in which case the equations are the same as in the classic case, simply by changing the order of the derivatives on the left hand side of the ODEs, or when studying problems where mass balance is not an issue such as PK/PD models, where there is no mass balance between the PK and the PD. This gives plenty of opportunities for future studies on PK/PD data [7]. Pharmacological signals are often characterised by delayed response and the history of previous states of involved variables needs to be taken into account in the models, instead of the instantaneous local state of the variables. FDEs provide this capability, along with other approaches which differ mathematically, but offer a similar flavour, such as delay differential equations, which have found important applications in pharmacometrics [25].

A final interesting comment about the current analysis is that in all cases the order of the FDE,  $\alpha$ , has been kept the same for all subjects, without IIV. Indeed it is not clear what could be the physical meaning of IIV on  $\alpha$ , since this parameter has so fundamental impact on the problem and even affects the units of other parameters. Considering a statistical distribution on a parameter such as  $\alpha$  seems to run into theoretical problems even by attempting to define it, despite the fact that from a numerical point of view it could be possible to estimate an OMEGA associated to  $\alpha$ .

## Conclusions

The present study implements FDE systems in NONMEM with the aim to allow the use of fractional PK models in NONMEM and to the best of our knowledge it is the first attempt to use FDE systems in a NLMEs framework, so the approach could be of interest to other disciplines apart from PKs. The method supports linear and non-linear models with multiple equations and multiple doses. In simulation studies the method proved capable to provide unbiased and precise estimates of the parameter values, while with a real PK data set, performed well, and gave an alternative, similarly good fit to a classic 3-compartment model, but more parsimonious. As a first attempt to introduce FDEs in the NLME framework there are limitations that will be

addressed in future work, furthermore it is hoped that the present provided software will motivate more applications.

**Acknowledgements** Acknowledgments are not compulsory.

**Funding** Open access funding provided by HEAL-Link Greece.

## Declarations

**Conflict of interest** The authors declare no conflicts of interest.

**Supplementary information** The code used in the article can be found in the GitHub page: <https://github.com/PMXathens/FDE4NONMEM>

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

- Podlubny I (1999) Fractional differential equations. Academic, New York
- Dokoumetzidis A, Macheras P (2009) Fractional kinetics in drug absorption and disposition processes. *J Pharmacokinet Pharmacodyn* 36:165–178. <https://doi.org/10.1007/s10928-009-9116-x>
- Dokoumetzidis A, Macheras P, Magin R (2010) Fractional kinetics in multi-compartmental systems. *J Pharmacokinet Pharmacodyn* 37:507–524. <https://doi.org/10.1007/s10928-010-9170-4>
- Dokoumetzidis A, Macheras P, Magin R (2010) A commentary on fractionalization of multi-compartmental models. *J Pharmacokinet Pharmacodyn* 37(2):203–207. <https://doi.org/10.1007/s10928-010-9153-5>
- Copot D, Chevalier A, Ionescu CM, Keyser RD (2013) A two-compartment fractional derivative model for propofol diffusion in anesthesia. In: IEEE international conference on control applications, 2013, pp. 593–660
- Hennion M, Hanert E (2013) How to avoid unbounded drug accumulation with fractional pharmacokinetics. *J Pharmacokinet Pharmacodyn* 40(6):691–700. <https://doi.org/10.1007/s10928-013-9340-2>
- Verotta D (2010) Fractional dynamics pharmacokinetics–pharmacodynamic models. *J Pharmacokinet Pharmacodyn* 37(3):257–276. <https://doi.org/10.1007/s10928-010-9159-z>
- Yin C, Li X (2011) Anomalous diffusion of drug release from a slab matrix: fractional diffusion models. *Int J Pharm* 418(1):78–87. <https://doi.org/10.1016/j.ijpharm.2010.12.009>
- Caputo M, Cametti C (2021) Diffusion through skin in the light of a fractional derivative approach: progress and challenges. *J Pharmacokinet Pharmacodyn* 48:3–19. <https://doi.org/10.1007/s10928-020-09715-y>
- Miskovic-Stankovic V, Janev M, Atanackovic TM (2022) Two compartmental fractional derivative model with general

- fractional derivative. *J Pharmacokinet Pharmacodyn*. <https://doi.org/10.1007/s10928-022-09834-8>
11. Neto JP, Alho I, Costa L et al (2021) Dynamic modeling of bone remodeling, osteolytic metastasis and PK/PD therapy: introducing variable order derivatives as a simplification technique. *J Math Biol*. <https://doi.org/10.1007/s00285-021-01666-3>
  12. Sopaşakis P, Sarimveis H, Macheras P, Dokoumetzidis A (2018) Fractional calculus in pharmacokinetics. *J Pharmacokinet Pharmacodyn* 45(1):107–125. <https://doi.org/10.1007/s10928-017-9547-8>
  13. Suli E, Mayers DF (2003) *An introduction to numerical analysis*. Cambridge University Press, Cambridge
  14. Tucker M (2022) Numerical inverse Laplace transform. MATLAB Central File Exchange
  15. Abate J, Whitt W (2006) A unified framework for numerically inverting Laplace transforms. *INFORMS J Comput* 18(4):408–421
  16. Talbot A (1979) The accurate numerical inversion of Laplace transforms. *J Inst Math Appl* 23(1):97–120
  17. Dingfelder B, Weideman JAC (n.d.) An improved Talbot method for numerical Laplace transform inversion. <https://doi.org/10.48550/ARXIV.1304.2505>
  18. Lubich C (1983) On the stability of linear multistep methods for Volterra convolution equations. *IMA J Numer Anal* 3:439–465
  19. Garrappa R (2015) Trapezoidal methods for fractional differential equations: theoretical and computational aspects. *Math Comput Simul* 110:96–112
  20. Chapra SC (2018) *Applied numerical methods with MATLAB® for engineers and scientists*. McGraw-Hill Education, New York
  21. Wang J, Feckan M, Zhou Y (2016) A survey on impulsive fractional differential equations. *Fract Calc Appl Anal*. <https://doi.org/10.1515/fca-2016-0044>
  22. Garrappa R (2010) On linear stability of predictor–corrector algorithms for fractional differential equations. *Int J Comput Math* 87(10):2281–2290. <https://doi.org/10.1080/00207160802624331>
  23. Greenblatt DJ, Allen MD, Harmatz JS, Shader RI (1980) Diazepam disposition determinants. *Clin Pharmacol Ther* 27(3):301–312
  24. Macheras P, Iliadis A (2010) *Modeling in biopharmaceutics, pharmacokinetics and pharmacodynamics: homogeneous and heterogeneous approaches*. Springer, New York
  25. Yan X, Bauer R, Koch G, Schropp J, Perez-Ruixo J, Krzyżanski W (2021) Delay differential equations based models in NON-MEM. *J Pharmacokinet Pharmacodyn* 48(6):763–802. <https://doi.org/10.1007/s10928-021-09770-z>

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.