

# Automated analysis of routinely generated preclinical pharmacokinetic and pharmacodynamic data

Emma Lindhardt

CVMD iMED DMPK AstraZeneca R&D,  
SE-431 83 Mölndal, Sweden

Peter Gennemark

CVMD iMED DMPK AstraZeneca R&D,  
SE-431 83 Mölndal, Sweden  
[peter.gennemark@astrazeneca.com](mailto:peter.gennemark@astrazeneca.com)

## Abstract

Model based analysis of routinely generated pharmacokinetic and pharmacodynamic (PKPD) data is a key component of preclinical drug discovery. The work process of such analyses can be automated by properly designed computer programs that reduce the number of manual steps, resulting in time savings and significantly fewer errors. Critical decisions can still be made by modelers. Using concrete animal data examples this paper illustrates when, and demonstrates how, automated PKPD approaches can be used and what benefits they offer to the modeling and simulation community. Specifically, we describe two compound optimization case studies from drug discovery projects, and also demonstrate how a subsequent optimization step to predict the human dose can be coupled to an automated approach.

Keywords: mathematical modeling, automated PKPD analysis, human dose prediction.

This manuscript was published in:

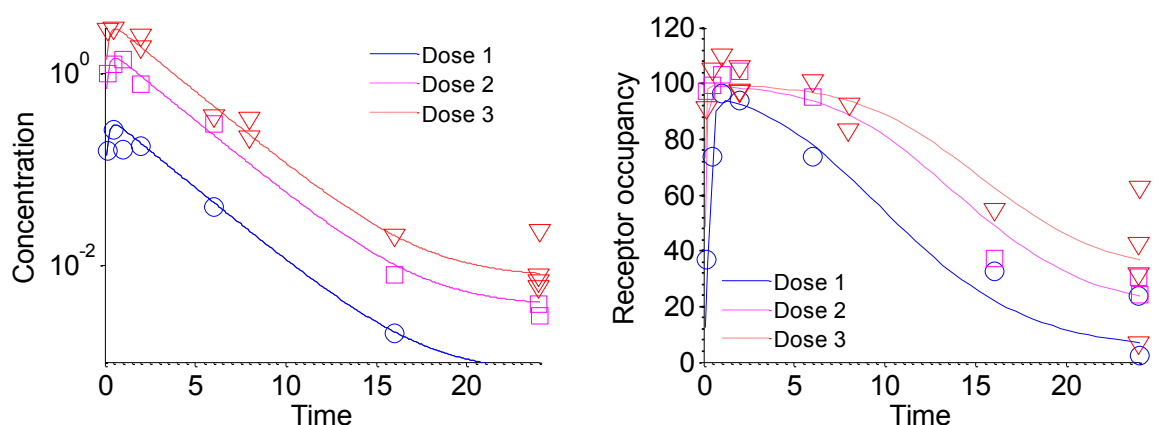
Lindhardt E, Gennemark P, Automated analysis of routinely generated preclinical pharmacokinetic and pharmacodynamic data, *J Bioinform Comput Biol* 12(3): 1450010, 2014

DOI: 10.1142/S0219720014500103

## Introduction

Traditionally, drug discovery and development mainly focus on interactions between a candidate drug and its immediate target. Information is collected from several sources, such as *in silico* predictions based on chemical structure of the drug candidate, cell-based assays, animal studies, and clinical studies. However, information integration tends to be non-quantitative and incomplete. As a consequence, decisions are not always well-grounded, and many initially promising drug candidates are found to lack efficacy or to have excessive toxicity in clinical studies. One way of improving the drug development process is through the use of model-based drug discovery and development (MBDD). Here, mathematical models are used to formally integrate all available information in order to gain system-level understanding of pharmacological effects and to make informed decisions. The benefit of this approach has been widely reported [1, 2], and MBDD is in the process of being implemented in industry.

Mathematical models in this area are usually composed of two parts; the pharmacokinetics (PK) part representing what the body does to the drug, and the pharmacodynamics (PD) part representing what the drug does to the body [3]. Models are frequently defined by ordinary differential equations (ODEs). A typical preclinical data set for a drug candidate is depicted in Fig 1.

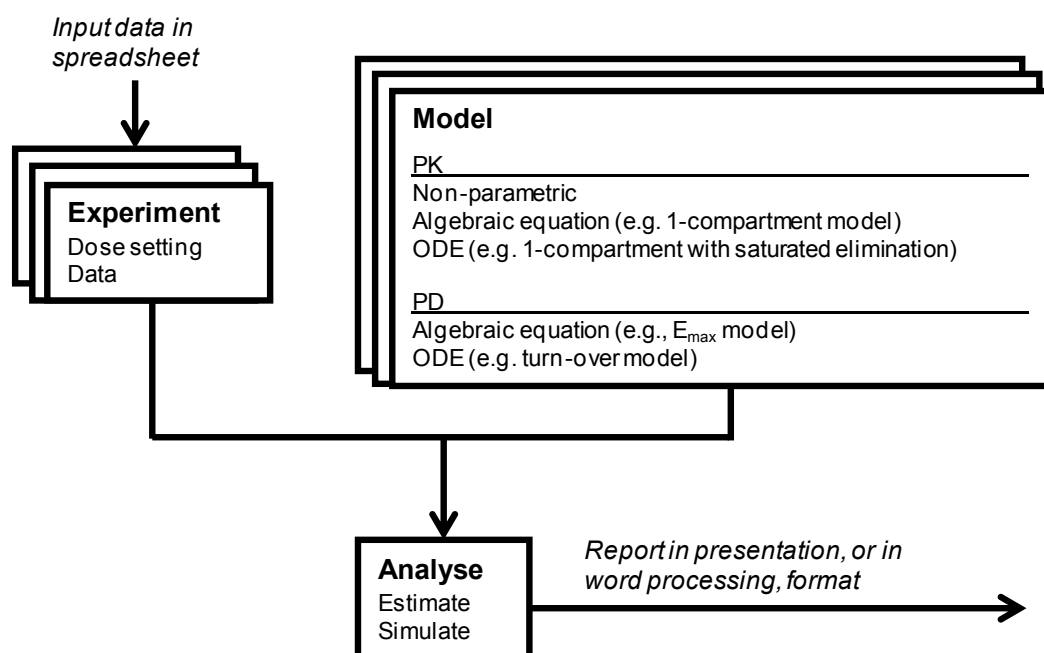


**Fig. 1** Pharmacokinetics and pharmacodynamic data sampled from mice exposed to various oral dose levels at time zero. Left: pharmacokinetics data (markers) for various doses of the drug candidate. Solid lines indicate the fit of a mathematical PK model. Right: pharmacodynamic data (markers) in form of occupancy of the receptor that is the target of the drug. Solid lines indicate the fit of a mathematical PKPD model.

Pharmacokinetic and pharmacodynamic modeling underpins drug discovery decision making by providing dynamical predictions with accompanying uncertainties. Routine PKPD work in preclinical drug discovery typically includes collecting data from several experiments, potentially with different dose administrations and dosing schedules, selecting a mathematical model, applying numerical methods for parameter estimation and simulation (prediction), and finally preparing a report for project review. In practice, data are typically stored in spreadsheets and results are reported as PowerPoint presentations (or occasionally in word processing formats).

The modeling process involves several manual steps; collect and order data, plot data, implement a possible mathematical model, find initial parameters, execute a numerical parameter estimation routine, evaluate the model fit to data, revise the model structure and/or initial parameters, prepare a report, etc. It is difficult to automate all these steps because data, model, and the scientific question differ from case to case.

However, in the late drug discovery phase, several drug candidates with similar chemical structure are typically screened in the same animal experiment. The purpose is to optimize the drug with respect to efficacy and safety before first time in man administration. For such routinely generated pharmacokinetic and pharmacodynamic data sets, we can assume a fixed data input file format, a fixed mathematical model structure, a fixed way of analyzing the data, and a fixed form for presentation. Under these conditions, we can describe the work process by a few well-defined steps (Fig. 2) and automate this process by a tailored computer program. The main benefit of the automatic approach is a reduction in the number of manual steps which results in time savings and significant fewer errors in the process.



*Fig. 2 Routine PKPD analyses in preclinical settings can typically be described by a few well-defined steps.*

Naturally, the more application-specific the program is implemented, the faster the modeling process can be executed. However, the benefit of an automated approach must be compared with the cost of developing the program. A complicating fact is that the number of drug candidates to be tested is usually unknown in advance, and may differ from only a handful up to tens. It may therefore be difficult to predict how many times the program will be run, and hence to make an exact cost-benefit analysis.

The purposes of this paper are:

- To describe how automation of PKPD analysis in two drug discovery compound optimization case studies has resulted in time savings.
- To demonstrate that automation of PKPD analyses is feasible even for projects with few expected drug candidates to evaluate.
- To demonstrate how subsequent optimization steps, here represented by human dose prediction, can be coupled to the automated approach.

Automation of mathematical modeling is a wide research area [4-10], and includes many aspects such as finding the model structure (model selection), searching data bases to find parameter values, and generating reports. Concerning relevant software for automated analysis of PKPD data one can consider standard numerical and statistical tools like Mathematica [11], Matlab [12], S-plus [13-15], R [16-17] and SAS [18]. Furthermore, dedicated PKPD tools like Monolix [19-21], Phoenix WinNonlin [22-23], and SimBiology [12,24] can be controlled by command line interaction. For Phoenix there is also a built in tool for automation referred to as the AutoPilot Toolkit. Additionally, one can consider general systems biology tools like PottersWheel [25], and for models represented by ODEs for which closed-form analytical solutions exist, one can in principle also take advantage of standard spreadsheet programs like Excel (Microsoft) given relevant parameter estimation add-ins.

## Materials and methods

We consider the following task (compare with Fig. 2):

**Implement** a computer program to automate a specific PKPD modeling analysis where the mathematical model structure is known but the parameters depend on input data.

**Input:** Spreadsheet (e.g. Excel) with temporal PKPD data, dosing schedule, dose levels, and initial parameter estimates.

**Output:** A user-defined report in presentation form (e.g. PowerPoint).

The usefulness of the program is measured by the number of data-sets  $N$  that are analyzed using the program, times the average time saving of each analysis  $T_{\text{saved per analysis}}$  in comparison to a traditional approach with manual steps, and finally, minus the time of program development  $T_{\text{development}}$ . In general, not only the

development cost, but also the maintenance cost and the user training cost contribute to the overall cost. In our case, the maintenance cost is significantly lower in comparison to  $T_{development}$ , mainly because the programs are very small and lack graphical user interface. The user training cost is also minimal. A typical program can be introduced to the user in a 30-minutes demonstration. Therefore,  $T_{development}$  is a reasonable approximation of the overall cost. Formally, the gain measured in time can be expressed as

$$T_{gain} = N \times T_{saved\_per\_analysis} - T_{development} \quad (1)$$

We aim at maximizing the gain by minimizing  $T_{development}$  and maximizing  $T_{saved\_per\_analysis}$ . The first is addressed by a proposing a carefully designed small library of useful building blocks (classes/functions). The latter is addressed by automation by a program built on the library. In addition, we expect fewer errors in the automatic approach compared to a manual approach. In order to minimize  $T_{development}$ , a software library for automated analysis was designed by a few compact and reusable generic classes. The program was implemented in Matlab R2012a [12]. The library contains classes for representing:

- dosing information, including route of administration (e.g. infusion or oral) and dosing schedule (e.g. one dose of 2 mg at time zero, and a second dose of 3 mg at time 48 hours).
- a time-series data set (data for some observed variable as depicted in Fig. 1).
- an experiment, including dosing information and one to many time-series data sets.

a model specifying the mathematical equations, and with methods for simulating the model (input to this method is dosing information and length of simulation), and for parameter estimation (input to this method is a set of experiments to which the model is fitted).

Furthermore, the library completes the full PKPD analysis work stream depicted in Fig. 2 by supporting reading and writing data from and to Excel, as well as writing presentations to PowerPoint. Based on the classes of the library, a tailored program can be built for a particular application. The resulting programs are small, easy to modify or extend and do not require advanced programming skills to use.

Main features of the library can be summarized as:

- Usefulness: compact yet functional programs based on the library can be written rapidly and such programs significantly reduces the number of manual steps (copy and paste, file management, polishing of figures etc.) of routine PKPD analysis.
- Usability: clear object-oriented design, well documented with examples.
- Generalization: The object-oriented implementation is designed for generalization.
- Extension: Matlab offers enormous potential to extend by arbitrary functionality, e.g. experimental design, dose optimization, model selection and graphical user interface, using a single software environment.
- Maintenance: it is easy to maintain a small set of well-documented modular classes.

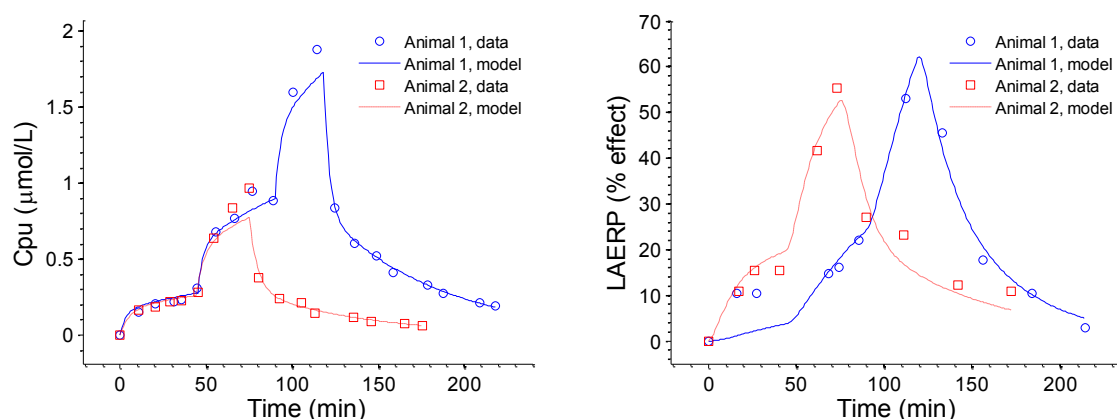
The library takes advantage of the Matlab functions `ode15s` for numerical integration of ODEs and `lsqnonlin` for non-linear least-squares. In this work, non-linear mixed effects modeling was not considered but support for that is available (`nlmefit`). To validate that the mathematical models were implemented correctly, data for several compounds previously analyzed by a traditional method (WinNonlin 5) were re-analyzed by the automated approach. The main limitation of the library is its dependence on Matlab. In principle, the publically available package GNU Octave could be used as an alternative. However, the rich extension possibilities in Matlab made us choosing that framework. The program files together with documentation and examples are available online [26].

## Results

Using the software library, we developed two project specific applications for automated PKPD analysis. Both programs take as input an Excel file with raw data, process the data (PK and PKPD modeling) and output a PowerPoint presentation summarizing the results. The application programs are called from Matlab, and for the two presented examples no graphical user interface was implemented. The second example extends PKPD analysis by a subsequent dose optimization step.

### *Hysteresis from a time-delayed PD effect*

To optimize the properties of a drug candidate, compounds with similar chemical structure were routinely screened in rabbit using intravenous infusions of varying dose levels and durations. Several PD measures were calculated from electrocardiography data, e.g. electrophysiological variables like atrial effective refractory period and QT prolongation (the QT interval is a measure of the time between the start of the Q wave and the end of the T wave in the heart's electrical cycle). Fig. 3 depicts a typical set of data for PK and one PD variable for one tested compound. In the most intense phase of the project, data for up to three different compounds were generated weekly.



**Fig. 3** Pharmacokinetics and pharmacodynamic data sampled from a rabbit model for one tested compound. Left: Pharmacokinetics data (markers) for two animals after infusions of the drug candidate (for animal 1; low concentration infusion 0-45 min, medium 45-90 min, and high 90-120 min; for animal 2; low concentration infusion 0-45 min, medium 45-75 min). Solid lines indicate the fit of a mathematical PK model. Right: pharmacodynamic data (markers) in form of percentage change of the left atrial effective refractory period (LAERP). Solid lines indicate the fit of a mathematical PKPD model.

Plotting observed effect versus observed drug plasma concentration typically revealed hysteresis (the PD effect depends not only on the current plasma concentration but also on past plasma concentrations), as depicted in the lower left graph of Fig. 4. To compare the potency of different compounds it is fundamental to infer the relationship between target site concentration and effect. To achieve this, the following mathematical modeling approach was taken for the compound series. First, the pharmacokinetics was modeled by a linear two-compartment model as

$$C_{pu}'(t) = \frac{D}{V} \times k_a \times e^{-k_a \times t} - k_{12} \times C_{pu}(t) + k_{21} \times C_2(t) - k_{10} \times C_p(t), \quad (2)$$

$$C_2'(t) = k_{12} \times C_{pu}(t) - k_{21} \times C_2(t), \quad (3)$$

where  $C_{pu}$  and  $C_2$  represent unbound (not bound to plasma proteins like albumin) concentration in the plasma compartment and in a second (tissue) compartment. The parameters  $V$ ,  $k_a$ ,  $k_{12}$ ,  $k_{21}$ , and  $k_{10}$  are fitted in the regression. The left part of Fig. 3 gives an example of a fitted curve.

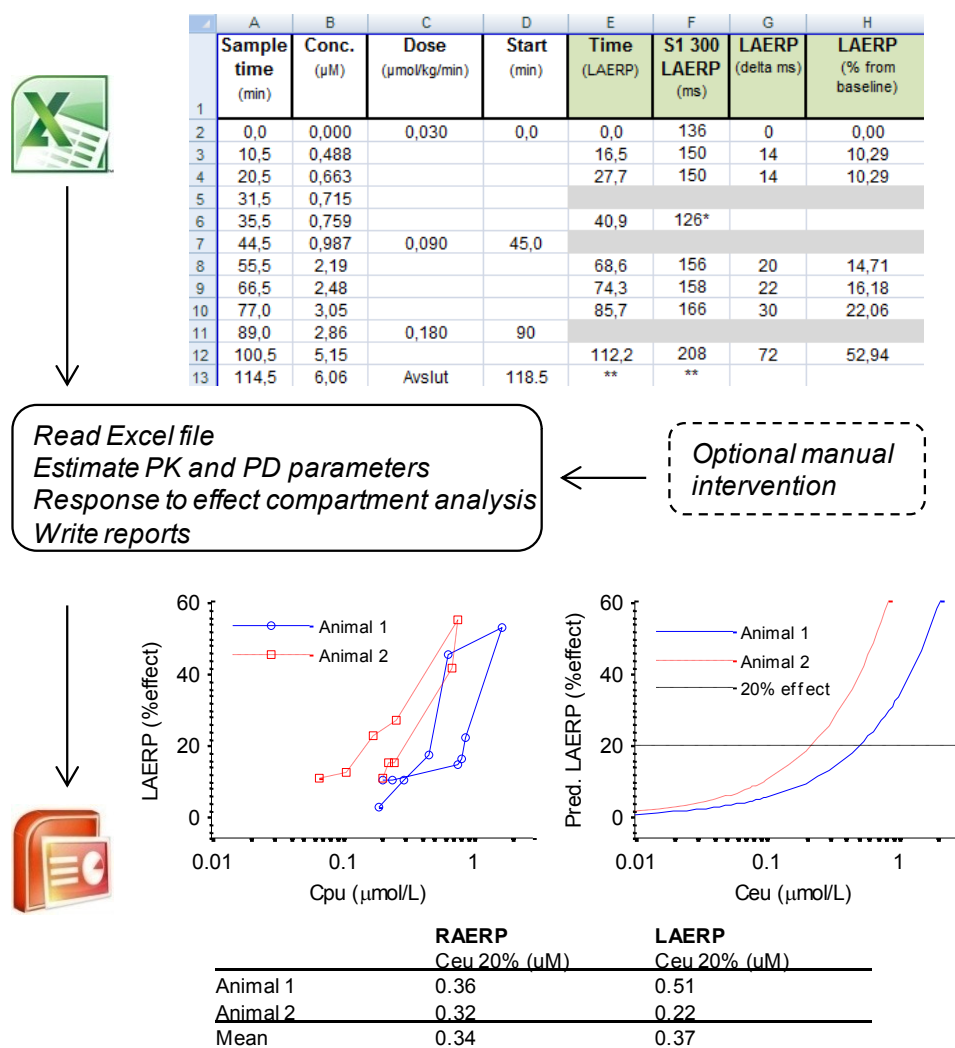
The pharmacodynamics of the drug was modeled as follows. A first-order distribution delay of compound to and from the biophase was assumed to be the rate-limiting step, and an effect-compartment model was used, where the concentration in the effect compartment was linked to the plasma compartment by a first-order equilibrium rate constant  $k_e$ . The rate of change of compound concentration in the effect compartment could then be expressed by

$$C_{eu}'(t) = k_e \times (C_{pu}(t) - C_{eu}(t)), \quad (4)$$

where  $C_{eu}$  denotes the unbound effect compartment concentration. The observed pharmacodynamic effect was modeled by a power function as

$$E(t) = a \times C_{eu}(t)^b, \quad (5)$$

where  $E$  denotes the effect variable, and  $a$  and  $b$  are empirical parameters. The effect compartment represents the target site concentration, and is used to infer concentration-effect relationship without ambiguity from hysteresis. A PKPD model fit is depicted in the right part of Fig. 3. Graphs of predicted effect versus predicted unbound effect compartment concentration were generated, as shown in the lower right graph of Fig. 4. From such data, summary measures, e.g. the unbound effective concentration at 20 % effect ( $C_{eu20}$ ), were tabulated as depicted at the bottom of Fig. 4. Finally, these measures were used to rank compounds and used as effective plasma concentrations in human dose predictions.



**Fig. 4** Automated PKPD analysis to study hysteresis from a time-delayed PD effect. The program takes as input an Excel file and produces a PowerPoint file as well as a text report. Part of a typical Excel file is shown in the upper part of the figure. The presentation file includes graphs of the hysteresis (left) and predicted effect vs. free effect compartment concentration (right), and summary tables as exemplified at the bottom of the figure. Optional manual intervention in the automated process includes viewing individual PK and/or PD fits and residuals, and adjusting default initial parameter values and/or bounds. L(R)AERP = Left(right) atrial effective refractory period

A key risk of a fully automated analysis is that inexperienced modelers ignore evaluating intermediate steps in the analysis and focus their attention on final output. A counter measure is to force the program to ask for manual intervention at specific points. For example, the program can automatically plot residual plots and request the user to confirm that these plots have been observed, or the program can ask for new parameter

bounds should a parameter hit the current lower or upper bound in the estimation. Hence, critical decisions can still be made by modelers.

In this particular program, manual intervention in the automated process occurs optionally at user specified points, e.g. after fitting of PK or PD parameters of individual animals and/or experiments. The user can view data fitting and residuals, and adjust initial parameters and parameter bounds. Alternatively, the program can iteratively and automatically try several initial conditions (randomly chosen or taken from a pre-specified set). The program is started from the Matlab command line, and a few input parameters specify what data to consider and potential points of user interaction. These points are hence entirely user controlled.

The program is small (a few hundred lines of code) and has been routinely used to significantly reduce the time of PKPD data analysis. With a traditional approach much time was spent on arranging data in Excel, moving or importing data to a parameter estimation/simulation software, estimating parameters for the different PD effects, copying data and figures to a PowerPoint presentation, as well as polishing that presentation. The estimated time saving of analyzing data for one compound using the program is  $T_{\text{saved per analysis}} = 1$  h (from 1.5 h to 0.5 h), and the program has been used for about  $N = 20$  compounds. Using the library, the tailored application was developed in  $T_{\text{development}} = 4$  h. Then, the benefit of the program includes an estimated total time saving of about 16 h (Eq. 1 with  $T_{\text{saved per analysis}} = 1$  h,  $N = 20$ ,  $T_{\text{development}} = 4$  h) as well as an improved accuracy since we expect fewer errors in the automatic approach compared to a manual approach.

### ***Receptor occupancy and human dose prediction***

In another drug candidate optimization program, time-series data for drug exposure and drug targeted receptor occupancy were generated from orally dosed mice. Data were reported on a standard spreadsheet format as illustrated in the upper part of Fig. 5. For the compound series, PK could be reliably modeled by a linear two-compartment model (Eq. 2 and Eq. 3; previous section), and the distribution between drug-receptor complex (RC) and free receptor (R) was modeled by a receptor kinetic model with elementary reactions as

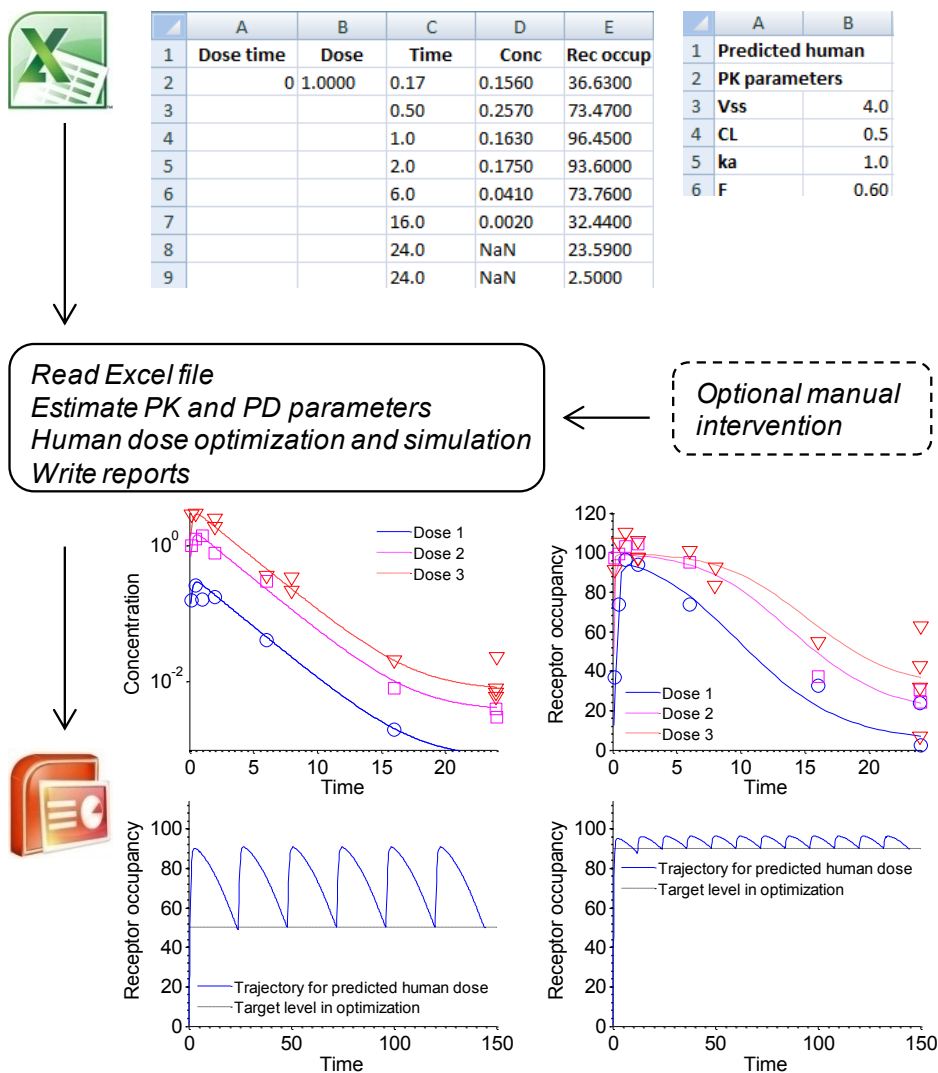
$$RC'(t) = k_{on} \times C(t) \times (R_{tot} - RC(t)) - k_{off} \times RC(t), \quad (6)$$

where  $C$  denotes drug plasma concentration,  $R_{tot}$  denotes the total receptor concentration, and  $k_{on}$  and  $k_{off}$  are kinetic parameters. Fig. 1 illustrates data for one compound and the corresponding model fit for plasma concentration and receptor occupancy. Besides rodent PKPD data, the program takes as input the predicted human PK parameters obtained from empirical scaling from both in vitro (hepatocytes) and in vivo (mouse/rat/dog) data [27]. Assuming the inferred PD parameters  $k_{on}$  and  $k_{off}$  to be species independent, a human dose prediction was calculated for several scenarios, e.g. requiring receptor occupancy of at least 50 % or 90 % for once daily or twice daily dosing schedules. As an extension to traditional PKPD analysis, the program automatically searched for the minimal required human dose for each combination of receptor occupancy coverage and dosing schedule. Traditionally, this step was performed by testing several doses and picking the one that approximately fulfilled the criterion. Here, this calculation was defined as an optimization problem: find the minimal predicted dose that ensures a certain level of receptor occupancy, given a certain number of doses per day, and the predicted human PK parameters. A simple half-interval search was used to find the minimal dose. For example, the lower part of Fig. 5 depicts predicted receptor occupancy time courses resulting in 50 % and 90 % receptor occupancy for once daily and twice daily administration, respectively. A summary of predicted doses for various scenarios is also tabulated in the presentation file.

Manual intervention in the automated process optionally occurs in the same way as described for the first example. Also this program is small (a few hundred lines of code) and has been routinely used to significantly reduce the time of PKPD data analysis and dose prediction.

The estimated time saving of analyzing data for one compound is  $T_{\text{saved per analysis}} = 3$  h (from 4 h to 1 h), and the program has been used for about  $N = 30$  compounds. Using the library, the tailored application including the human dose optimization part was developed in  $T_{\text{development}} = 16$  h. Then, the benefit of the program includes an estimated total time saving of about 74 h (Eq. 1 with  $T_{\text{saved per analysis}} = 3$  h,  $N = 30$ ,  $T_{\text{development}} = 16$  h) as well as an improved accuracy as discussed for the previous example. Specifically, accuracy and consistency of the human dose prediction is significantly improved.

Potential extensions within the proposed framework include incorporating the empirical PK scaling into the application, and enlarging the space of model structures to run automatic model selection.



**Fig. 5** Automated PKPD analysis in receptor occupancy modeling with a subsequent human dose prediction. The program takes as input an Excel file and produces a PowerPoint file. The upper part shows a typical Excel sheet for one rodent experiment as well as a sheet with human PK parameters. The presentation file includes graphs of fitted PK and PKPD rodent data (the two upper graphs), as well as predicted human trajectories, here exemplified by receptor occupancy (the two lower graphs). Human doses are based on scaling of human PK parameters from *in vitro* and *in vivo* data, and automatically optimized to maintain a certain occupancy level (50 % in the lower left graph for once daily administration, and 90 % in the lower right graph for twice daily administration). Optional manual intervention is available as described in Fig. 4

## Discussion

Using concrete drug discovery examples this paper illustrates when, and demonstrates how, automated PKPD approaches can be used and what benefits they offer to the industrial modeling and simulation community. Specifically, we have presented two applications in the late preclinical discovery phase. The first example is restricted to traditional PKPD analysis and automated solutions can relatively easily be implemented in a range of software. This example shows that customized computer programs can significantly accelerate the workflow of analyzing routinely generated PKPD data. We also note that automated PKPD analysis have had a significant impact in the area of cardiovascular safety, since experimental set-ups and PKPD models in that area are similar to those of this example. The second example shows how automated processes can be connected to other calculations, here illustrated by an optimization step to predict of the human dose. Such an extension is presently not supported by standard PKPD tools.

While automated PKPD analysis is most beneficial in the late preclinical discovery phase, it can also play a role in earlier phases of the discovery process. In our experience, the modeling analysis of a specific compound is



often repeated when additional in vitro and/or in vivo data become available. If such situations occur frequently, automated approaches are beneficial also in early phases. For example, we have applied the same concepts to the following recurring work process in an early drug discovery project. The computer program first analyzes PK data from a single dose experiment, and then predicts a PKPD experiment with repeated doses, assuring that the exposure is sufficiently high. This optimization is guided by in vitro potency data and a continuously adjusted in vitro-in vivo relationship based on data from previously studied compounds targeting the same receptor.

When designing the library of software classes, we have prioritized the possibility for extension (fundamental for the human dose optimization in the second example), and with respect to usefulness, we require output to PowerPoint which is the standard presentation format in our drug discovery projects. We further had a preference for using an object-oriented programming approach to ensure modularity which improves usability and to ensure a low maintenance cost of the library as well as a low development cost of tailored applications based on the library. Based on this prioritization we have chosen to use Matlab which is a standard engineering software that can easily be extended by arbitrary functionality using a single software environment.

Future developments of the software may include richer model libraries, library classes for translation between species and human dose prediction, as well as for uncertainty quantification in human dose prediction.

The benefit of an automated approach should be compared with the cost of developing the application. Based on data in this paper, as well as our experience from other projects where the library has been used [28], we argue that the initial investment of software development can be low by reusing generic classes and by modifying project specific code from projects like the ones presented here. In this way, automated analysis of preclinical PKPD data can be beneficial to a wide range of discovery projects. Together, the two examples demonstrate that automated PKPD data analysis and extensions can be cost-efficiently implemented. We see a great potential to apply the proposed or similar methods generally in late drug discovery phase projects, and potentially also in projects in other phases of drug discovery.

## Acknowledgments

We thank Philip Werner for help with the software implementation, and Johan Gabrielsson and Ann-Charlotte Egnell for constructive feedback.

## References

1. Lalonde RL, Kowalski KG, Hutmacher MM, Ewy W, Nichols DJ, et al. Model-based drug development. *Clin Pharmacol. Ther.*, **82**(1):21–32, 2007.
2. Visser SA, Aurell M, Jones RD, Schuck VJ, Egnell AC, Peters SA, Brynne L, Yates JW, Jansson-Löfmark R, Tan B, Cooke M, Barry ST, Hughes A, Bredberg U. Model-based drug discovery: implementation and impact. *Drug Discov Today*, **18**(15-16):764-75, 2013.
3. Gabrielsson J, Weiner D, Pharmacokinetic & Pharmacodynamic, Data analysis, concepts and applications. 4th edition, 2nd print, Swedish Pharmaceutical Press, Stockholm. ISBN 139789197651004, 2010.
4. Schaefer P, Automated reporting of pharmacokinetic study results: gaining efficiency downstream from the laboratory. *Bioanalysis* **3**(13):1471–78, (2011)
5. Logg A, Automation of Computational Mathematical Modeling, ISBN: 91-7291-436-X, Chalmers University of Technology, Göteborg, 2004
6. Gennemark P. and Wedelin D. Benchmarks for identification of ordinary differential equations from time series data. *Bioinformatics* **25**(6):780-6.
7. Borger S, Liebermeister W, Uhlendorf J, Klipp E, Automatically generated model of a metabolic network. *Int Conf Genome Inform* **18**:215–224, 2007
8. Dräger A, Adrian Schröder A, Zell A, Automating Mathematical Modeling of Biochemical Reaction Networks. Systems Biology for Signaling Networks (Editor: Choi S) ISBN: 978-1-4419-5796-2. *Systems Biology* **1**:159-205, 2010
9. Castillo O, Melin P, Automated mathematical modelling, simulation and behavior identification of robotic dynamic systems using a new fuzzy-fractal-genetic approach, *Robotics and Autonomous Systems*, **28**(1), 19-30(12), 1999

10. Castillo O, Melin P, An intelligent system for automated mathematical modelling and simulation of dynamical engineering systems, in: Proceedings of AIENG'96, 1996, pp. 9-10.
11. Wolfram Research, Inc., Mathematica, Version 8.0, Champaign, IL (2010).
12. MATLAB 2012b, The MathWorks, Inc., Natick, Massachusetts, United States.
13. Becker, RA, Chambers JM, Wilks AR, The New S Language: A Programming Environment for Data Analysis and Graphics. Pacific Grove, CA, USA: Wadsworth & Brooks/Cole. ISBN 0-534-09192-X, 1988
14. Chambers JM, Hastie TJ, Statistical Models in S. Pacific Grove, CA, USA: Wadsworth & Brooks/Cole. ISBN 0-412-05291-1, 1991
15. TIBCO Software Inc. Palo Alto, CA, USA
16. R Core Team, R: A Language and Environment for Statistical Computing, R Foundation for Statistical Computing, Vienna, Austria, <http://www.R-project.org>, last visited 2013-11-27
17. Wright DB, London K. Modern Regression Techniques Using R: A Practical Guide. SAGE, London, UK, 2009
18. SAS 9.3, SAS Institute, Cary NC
19. Lixoft, Orsay, France
20. Lavielle M, Mentré F. Estimation of population pharmacokinetic parameters of saquinavir in HIV patients with the MONOLIX software. *J Pharmacokinet Pharmacodyn.* **34**(2):229-49, 2007
21. Chan PL, Jacqmin P, Lavielle M, McFadyen L, Weatherley B. The use of the SAEM algorithm in MONOLIX software for estimation of population pharmacokinetic-pharmacodynamic-viral dynamics parameters of maraviroc in asymptomatic HIV subjects. *J Pharmacokinet Pharmacodyn.* **38**(1):41-61, 2011
22. Certara, St. Louis MO, USA
23. Yan X, Chen Y, Krzyzanski W. Methods of solving rapid binding target-mediated drug disposition model for two drugs competing for the same receptor. *J Pharmacokinet Pharmacodyn.* **39**(5):543-60, 2012
24. Mol M, Patole MS, Singh S. Immune signal transduction in leishmaniasis from natural to artificial systems: Role of feedback loop insertion. *Biochim Biophys Acta.* **1840**(1):71-79, 2013
25. Maiwald T, Timmer J, Dynamical Modeling and Multi-Experiment Fitting with PottersWheel. *Bioinformatics* **24**(18):2037-2043, 2008
26. Gennemark P, <http://www.mathworks.com/matlabcentral/fileexchange/43521-preclinical-pkpd-modeling>, last visited 2013-11-27
27. Hosea NA, Collard WT, Cole S, Maurer TS, Fang RX, Jones H, Kakar SM, Nakai Y, Smith BJ, Webster R, Beaumont K, Prediction of human pharmacokinetics from preclinical information: comparative accuracy of quantitative prediction approaches, *J Clin Pharmacol.*, **49**(5):513-33, 2009
28. Gennemark P, Jansson-Löfmark R, Hyberg G, Wigstrand M, Kakol-Palm D, Håkansson P, Hovdal D, Brodin P, Fritsch-Fredin M, Antonsson M, Ploj K, Gabrielsson J. A modeling approach for compounds affecting body composition. *J Pharmacokinet Pharmacodyn.* **40**(6):651-67, 2013