Supplementary Information 1

Title: Lung Physiologically Based Pharmacokinetic Modelling to Predict Sublingual Buprenorphine Kinetics Following Oral Inhalation.

Tobechi Brendan Nnanna[1], Daniel Okwudili Nnamani[2], Adikwu Elias[3], Chisom Anthony Nnanna[4]

[1]Aston Pharmacy School, College of Health and Life Sciences, Aston University, Birmingham B47ET, UK

[2]Department of Pharmaceutical Sciences, University of Tennessee Health Science Centre, Memphis, TN 38103, USA.

[3]Department of Pharmacology and Toxicology, Faculty of Pharmacy, Niger Delta University, Bayelsa State, Nigeria.

[4]Department of Pharmacology and Toxicology, Faculty of Pharmacy, Madonna University, Nigeria.

Table of Contents.

1. OSPSuite-R Ecosystem and Package Dependencies

The computational scaffolding of the investigation leverages the OSPSuite-R framework, an R-based conduit that mechanistically interfaces with the Open Systems Pharmacology (OSP) Suite (PK-Sim® and MoBi®). The package constellation as depicted in Figure 1, underpins model construction, execution, and parameter analytics through an ensemble of bridging, numerical, and utility modules. The table below (Table 1) delineates the principal packages, their prerequisites, and their functional contributions to the modelling workflow in the study.

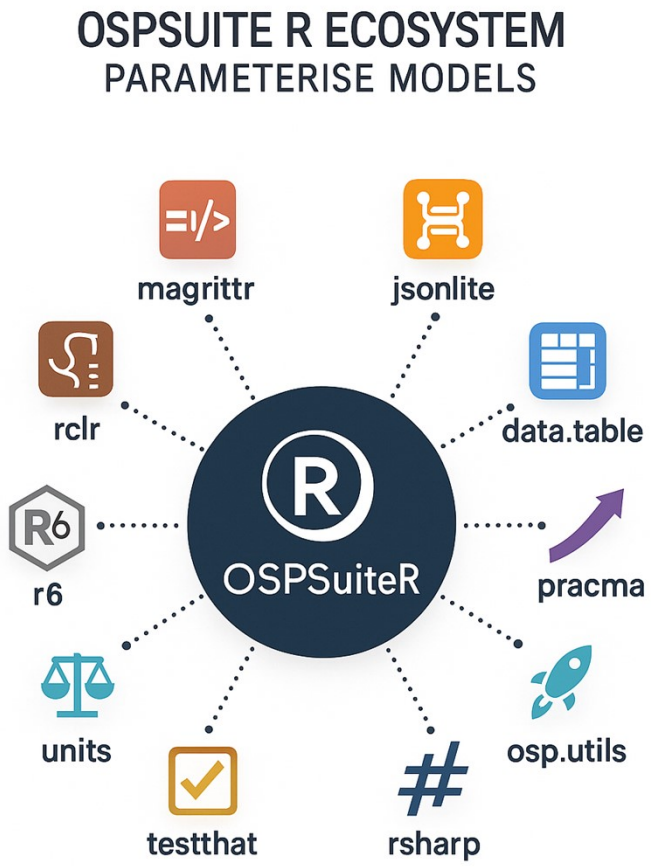


Figure 1. OSPSuite Dependency packages.

Table 1. Principal OSPSuite-R ecosystem packages

R Package	Role in Workflow	Type
ospsuite **v11.2.251*	This package forms the nucleus of the computational framework. It provides high-level application programming interfaces (APIs) to load, parameterise, and simulate PBPK models constructed in PK-Sim/MoBi. It encapsulates model entities (compartments, parameters, processes) as R objects, thereby enabling programmable access, manipulation, and simulation orchestration within the R environment.	Core Package
rClr *v.0.9.2*	This package establishes the interlanguage interface between R and the .NET runtime (on which PK-Sim/MoBi are built). It essentially facilitates the execution of .NET methods from within R, thus ensuring seamless communication between model engines and the R analytical layer.	Pre-requisite
rsharp *V1.12*	The package extends rClr functionality by simplifying and stabilising communication calls between R and the OSP Suite's .NET Core. Ensures that parameter retrieval, assignment, and execution routines are computationally efficient and error resilient. It is based on the rClr package and utilises some of the code base.	Utility
osp.utils *v1.5.0*	Invocation of this package provides auxiliary functions specific to OSP workflows, including standardised wrappers for simulation management, data export, logging, and error handling. It acts as a scaffolding library that harmonises repetitive tasks across multiple OSP-related analyses.	Utility
pracma *v2.4.4*	An activation of this package supplies advanced numerical algorithms (integration, matrix algebra, optimisation, interpolation) requisite for differential equation handling, sensitivity analyses, and system-level parameter calibration. It complements core OSP simulation routines with high-performance mathematical support.	Numerical method
data.table *1.17.0*	It enables memory-efficient handling of large pharmacokinetic datasets (for instance, concentration-time–time profiles, Monte Carlo simulations). Its high-performance tabular structure supports rapid subsetting, aggregation, and reshaping operations.	Data Handling
jsonlite *v2.0.0*	The package activation facilitates serialisation and deserialization of model configuration files and simulation outputs in JSON format, ensuring interoperability between OSP software, R scripts, and external data repositories.	Data Exchange
R6 *v2.6.1*	Following the activation of this package, it implements encapsulated object-oriented programming within R, essential for representing hierarchical PBPK model entities (compartments, organs, processes) and preserving state across simulation workflows.	Object-Oriented structure
magrittr *v2.0.3*	Activation of the package introduces the pipe operator (%>%) in R, which enhances clarity and readability in sequential simulation commands and data processing workflows.	Workflow Streamlining
units *v0.8.7*	Activation of the package reinforces comprehensive unit management by ensuring dimensional correctness in model inputs/outputs (for instance, µg/L, L/h), thereby reducing epistemic error in cross-simulation analyses.	Dimension consistency
testthat *v3.2.3*	On activation, it provides a formalised suite for regression testing of simulation pipelines, ensuring reproducibility and integrity across computational experiments.	Testing framework
knitr v1.50 / *markdown v2.0*	When activated, it enables literate programming practices, allowing for the reproducible integration of model code, simulation outputs, and narrative reporting into unified, dynamic documents. Crucial for transparent dissemination of PBPK model performance and results.	Documentation tool

v – denotes the version number of the package used for the PBPK modelling and simulation study.

2. Human Lung Model Development

2.1 Human lung Morphometry

The corresponding pulmonary PBPK model for humans adhered to the morphometric framework of the Weibel lung model, with specific parameters given in referenced literature (Weibel, 1963). The segmentation and calculation of lung volumes and surface areas adhered to the methodologies established by Borger et al. (2018). Furthermore, Table 2 depicts the generation characterisation metrics of the Weibel model.

2.2 Surface areas

The bronchial passages have been considered as hollow cylindrical structures. The surface area of each ELF compartment was computed as follows:

$$S_{n,ELF} = \pi \cdot N_n \cdot D_n \cdot L_n \qquad \text{(Eq. S1)}$$

N_n represents the number of airways, D_n denotes the diameter, and L_n signifies the length of the airway of generation *n*. The surface area ($S_{n,ept}$) for each epithelial compartment was computed as follows.

$$S_{n,ept} = \pi \cdot N_n \cdot (D_n + 2 \cdot height\ of\ epithelium) \cdot L_n \qquad \text{(Eq. S2)}$$

The total surface area of the alveolar region of the human lung is 7000 dm² (Combs and Dickson, 2020). The surface areas of both ELF compartments and epithelial compartments in the alveolar region were modified, with additional surface areas allocated to each generation based on its degree of alveolarization f_n.

The additional surface areas ΔS were calculated as follows:

$$\Delta S = S_{total,alv} - \sum_{n=17}^{24} S_n \qquad \text{(Eq. S3)}$$

$S_{total,alv}$ represents the actual area of the alveolar region as reported in the literature, while S_n denotes the surface area of each segment prior to adjustment. The surface area of compartment *n* in the alveolar region $S_{ALV,n}$ was subsequently calculated as:

$$S_{ALV,n} = S_n + f_n \cdot \Delta S \quad \text{(Eq.S4)}$$

The f_n values were applied similarly to the method described by Borger et al.(2018) Values for generations 17 to 24 were as follows: f_{17}= 0, $f_{18} = 0.2\%$, $f_{19} = 0.7\%$, $f_{10} = 2.0\%$, $f_{21} = 7.0\%$, $f_{22} = 13.9\%$, $f_{23} = 28.2\%$, and $f_{24} = 48.0\%$.

2.3 Compartmental Volumes

Given the cylindrical architecture of pulmonary airways, the volume of the epithelial lining fluid in compartment n V_n was computed as follows:

$$V_n = N_n \cdot L_n \cdot S_{cross} \quad \text{(Eq.S5)}$$

L_n denotes the length of each airway. The cross-sectional areas S_{cross} were calculated as follows:

$$S_{cross,ELF} = \pi \cdot \left(\left(\frac{D_n}{2}\right)^2 - \left(\frac{D_n}{2} - h_{ELF}\right)^2\right) \quad \text{(Eq. S6)}$$

$$S_{cross,ept} = \pi \cdot \left(\left(\frac{D_i}{2} + h_{ept}\right)^2 - \left(\frac{D_i}{2}\right)^2\right) \quad \text{(Eq.S7)}$$

Where h_{ELF} and h_{ept} denote the height (cross-sectional thickness) of the epithelial lining fluid and the epithelium, respectively.

The cross-sectional area of the subepithelium ($S_{cross,sub}$) for the human model is calculated as follows:

$$S_{cross,sub} = \pi \cdot \left(\left(\frac{D_n}{2} + h_{ept} + h_{sub}\right)^2 - \left(\frac{D_n}{2} + h_{ept}\right)^2\right) \quad \text{(Eq.S8)}$$

h_{sub}represents the height of each subepithelium segment. The literature indicates that the average lung volume in humans is approximately 0.53 litres (Brown et al., 1997). The modified volume of subepithelial compartments correlates with the initial volume of each compartment prior to adjustment $V_{sub,n}$. The adjusted volume of each subepithelial compartment $V_{sub,adj,n}$ was determined using the following calculation:

$$V_{sub,adj,n} = (V_{total,lung} - \sum_{n=1}^{24}(V_{ELF,n} + V_{ept,n})) \cdot \frac{V_{sub,n}}{\sum_{n=1}^{24} V_{sub,n}} \quad \text{(Eq.S9)}$$

$V_{total,lung}$represents the actual volume of the lung as documented in the literature.

A typical path model developed by Pellowe was utilised to forecast drug deposition across each generation (Pellowe, 2023). The model incorporated tidal breathing parameters in human subjects, specifying a tidal volume of 1000 ml, a breathing frequency of 15 breaths per minute, and an inspiratory to expiratory ratio of 1, executed without pause (Boger and Wigström, 2018).

Human physiological parameters were incorporated as well. In the investigation of orally inhaled xenobiotics in human subjects, the extrathoracic compartment, designated as generation 0, was defined as the oral cavity. An absorption process to the gastrointestinal tract was incorporated, characterised by a rate of absorption (ka).
The cardiac output measured 5.2 L/min, while the maximum MCC rate was established at 3.6 mm/minute at the apex of the human lung (Boger and Wigström, 2018).

2.4 Local Blood Perfusion

Literature reported tracheobronchial region blood perfusion $Q_{BF}$is circa 2% (Chamarthy et al., 2018) of the cardiac output (Q_{co}) for humans (indicated above). The local blood flow in the tracheobronchial region of the lung is directly proportional to the subepithelial volume of the generation, along with a weighting factor suggested by Bernard et al., (1996).

For each generation *n* at the TB region, the blood perfusion $Q_{BF,n}$is estimated using the following equation:

$$Q_{BF,n} = Q_{BF} \times \frac{V_{sub,n}}{\sum_{n=1}^{16} V_{sub,n}} \times F_n \quad \text{(Eq.S10)}$$

The weighting factor, denoted as F_n, can be determined based on the diameter of the airway as follows:

$$F_n = 0.19 + 2.8e^{-5.1D_n} \quad \text{(Eq.S11)}$$

For each generation *n* in the alveolar region, the blood flow ($Q_{CO,n}$) is estimated as follows:

$$Q_{CO,n} = Q_{co} \times \frac{V_{sub,n}}{\sum_{n=17}^{24} V_{sub,n}} \quad \text{(Eq.S12)}$$

Table 2. Lung Generation Characteristics of the Weibel Model.

n	N_n	D_n [cm]	L_n [cm]
1	1	1.539	10.26
2	2	1.043	4.07
3	4	0.71	1.624
4	8	0.479	0.65
5	16	0.385	1.086
6	32	0.299	0.915
7	64	0.239	0.769
8	128	0.197	0.65
9	256	0.159	0.547
10	512	0.132	0.462
11	1024	0.111	0.393
12	2048	0.093	0.333
13	4096	0.081	0.282
14	8192	0.07	0.231
15	16384	0.063	0.197
16	32768	0.056	0.171
17	65536	0.051	0.141
18	131072	0.046	0.121
19	262144	0.043	0.1
20	524288	0.04	0.085
21	1048576	0.038	0.071
22	2097152	0.037	0.06
23	4194304	0.035	0.05
24	8388608	0.035	0.043

Here, n refers to the generation number, N_n refers to the number of airways, D_n refers to the airway diameter, and $L_n$refers to the length of the lung generation. Note that these values have been scaled to match a functional residual capacity of 3000 mL for the lung.

3.0 Mucociliary Clearance and Xenobiotic Dissolution

3.1 Xenobiotic Dissolution in Airways.

The particle dissolution function within PKSim® was utilised in the present model. Dissolution was observed in each slice unit within the airway. Upon deposition of a particle in the airway, it is subject to dissolution within the epithelial lining fluid (ELF) compartments. In each slice of the generation, the dissolved mass per time unit (dA/dt) is characterised as:

$$\frac{\partial A}{\partial t} = 4 \cdot \pi \cdot r^2 \cdot N_p \cdot \frac{D}{h}\left(\frac{S}{MW} - C_{lumen}\right) \qquad \text{(Eq.S13)}$$

N_p represents the number of particles within the slice, h indicates the thickness of the unstirred water layer. S represents the solubility of the molecule in ELF, D denotes the diffusion coefficient, and C_{lumen} represents the concentration of the drug in the epithelial lining fluid (ELF). The estimation of Np was conducted using the particle deposition function. The value was established at 20 µm. D represents the aqueous diffusion coefficient, which is estimated based on molecular weight in Mobi®. **r** was calculated using the subsequent function in Mobi®:

$$r = \sqrt[3]{\left(\frac{3 \times mass}{4 \times N_p \times \rho \times \pi}\right)} \qquad \text{(Eq.S14)}$$

where $mass$ represents the entire quantity of the drug assigned to the segment by the particle deposition algorithm. ρ denotes the density of the investigated xenobiotic.

3.2 Mucociliary Clearance.

Boger et al. argue that the MCC rate in the TB region is directly proportional to the cross-sectional area of each generation, achieving its maximum at the first generation (Boger and Wigström, 2018).

$$MCC\ rate = \frac{S_{cross,n}}{S_{cross,n=1}} \times MCC_{max} \qquad \text{(Eq.S14)}$$

The maximum MCC rate for humans is 3.6 mm/min (Boger and Wigström, 2018). The MCC rates for each generation were calculated based on their cross-sectional areas and lengths. The residence time

(RT) for particles in each generation was also estimated.

$$RT(min) = \frac{L_n}{MCC\ rate} \quad \text{(Eq.S15)}$$

where L_n denotes the length of each generation in the TB area. The model delineates the MCC movement by propelling undissolved particles upward to the extrathoracic area every fifteen minutes. Each generation was segmented into several slices N_{slices} of uniform residence duration, determined as follows:

$$N_{slices} = \frac{RT}{15} \quad \text{(Eq.S16)}$$

The MCC model would subsequently propel particles from the base of the TB region every fifteen minutes until they arrive at the extrathoracic region. Metrics of the MCC model are depicted in Table 3.

Table S3. Metrics for Mucociliary Clearance Modelling in Human.

GN	Residence Time [min]	Approximated RT [min]	Number of Slices
1	28.5	30	2
2	24.62	30	2
3	21.2	15	1
4	18.64	15	1
5	48.2	45	3
6	67.34	60	4
7	88.57	90	6
8	110.19	105	7
9	140.35	135	9
10	174.45	180	12
11	209.86	210	14
12	253.31	255	17
13	282.78	285	19
14	310.16	315	21
15	326.56	330	22
16	358.75	360	24

GN – generation number, RT - residence time

4.0 Particle Deposition

The particle deposition mechanisms implemented by the inhalational PBPK model established by Pellowe (2023) notably includes Inertial impaction, gravitational sedimentation and diffusion. Theoretical underpinning of these mechanisms and the relevant equations are briefly described thus:

4.1 Inertial Impaction

In the respiratory system, inertial impaction serves as the predominant deposition mechanism for particles over 5 microns in diameter and is a significant mechanism for particles as tiny as 2 microns, contingent upon flow rate (Darquenne, 2020). Inertial impaction transpires when there is an abrupt alteration in the flow direction, resulting in particles deviating from the air streamlines due to their inertia, maintaining their original paths. Consequently, particles may collide with airway walls and be extricated from the airflow. The likelihood of a particle deviating from the air streamlines can be articulated as a function of the Stokes number (Stk) defined by:

$$Stk = \frac{\rho_p \, d_p^2 u}{18 \mu d} \qquad \text{(Eq.S17)}$$

where ρ_p and d_p denote the particle diameter and density (kg/m^3), u and μ represent the mean velocity and dynamic viscosity of the carrier gas, respectively, and d signifies the airway diameter. An increased Stokes number correlates with enhanced efficiency of inertial transport and a greater likelihood of particle deposition via inertial impaction. In instances of respirable particles within the upper conducting airways of the human lung, the Stk typically ranges from 0.001 to 0.1 (Zhang et al., 1997).

4.2 Gravitational Sedimentation

Gravitational sedimentation denotes the deposition of particles influenced by gravitational forces. When the gravitational force matches the opposing viscous resistive forces of the air, particles attain their terminal settling velocity, represented by:

$$v_S = \frac{\rho_p d_p^2}{18 \mu} \cdot g \qquad \text{(Eq.S18)}$$

Where g denotes gravitational acceleration with a corresponding value of 9.81m/s^2

4.3 Diffusion

The third essential deposition mechanism in the lung is diffusion, which arises from the random movements of particles due to their collisions with gas molecules. The diffusion rate is directly proportional to the Brownian diffusion coefficient (Darquenne, 2020).

$$D = \frac{ckT}{3\pi\mu d_p} \qquad \text{(Eq.S19)}$$

In this context, k represents Boltzmann's constant (1.380649 × 10^{-23} m^2 kg s^{-2} K^{-1}), T denotes the absolute temperature, and the statistic c represents the Cunningham correction factor. This factor accounts for the decrease in air resistance caused by slippage when the particle diameter nears the mean free path of gas molecules. Consequently, the particle behaves as a distinct entity among individual gas molecules instead of as part of a continuous medium within the gas.

Diffusion deposition accelerates as particle size diminishes. Diffusion deposition primarily transpires within the lung acinar region ; nonetheless, it is important to acknowledge that for particles of diminutive size (dp < 0.01 μm), diffusion deposition is also considerable in the nasal cavity, oral cavity, and pharyngeal passages (Darquenne, 2020).

5.0. Effective Permeability Calculation in Mobi

The estimated effective permeability of the investigated compound in the study was ascertained through the default implemented PKsim permeability calculation method (Thelen et al., 2011) in Mobi, with the equation (and term description) as follows:

$$P_{eff}(MW_{eff}, MA) = A \times \frac{MW_{eff}^{-\alpha-\beta} \times MA}{MW_{eff}^{-\alpha} + B \times MW_{eff}^{-\beta} \times MA} + C \times \frac{MW_{eff}^{-\gamma}}{D^{-\gamma} + MW_{eff}^{-\gamma}} (cm/s) \qquad \text{(Eq.S20)}$$

The first term on the right side denotes transcellular absorption, whereas the following term relates to paracellular passive intestinal absorption. The exponents α and β indicate the mass dependence of the diffusion coefficients in water (α) and in the membrane (β). The Stokes–Einstein law posits that an α value of 3/2 serves as a plausible approximation for the exponent indicative of the mass dependence of the diffusion coefficient in water. The parameter indicating the mass dependence of the diffusion coefficient in the membrane exhibits values ranging from β = 2 to 6 in the literature, contingent upon

the model system employed. The expression "$MW_{eff}^{-\alpha} + B \times MW_{eff}^{-\beta} \times MA$" represents the diffusion process within the unstirred water layer, while the term "$MW_{eff}^{-\gamma} / (D^{-\gamma} + MW_{eff}^{-\gamma})$" characterises a sigmoid function with values ranging from 0 to 1 and a slope of γ. "D" can be understood as a threshold value for the molecular weight that permits paracellular transit via tight junctions. Consequently, the value of "C" represents the permeability coefficient for unadulterated paracellular transit.

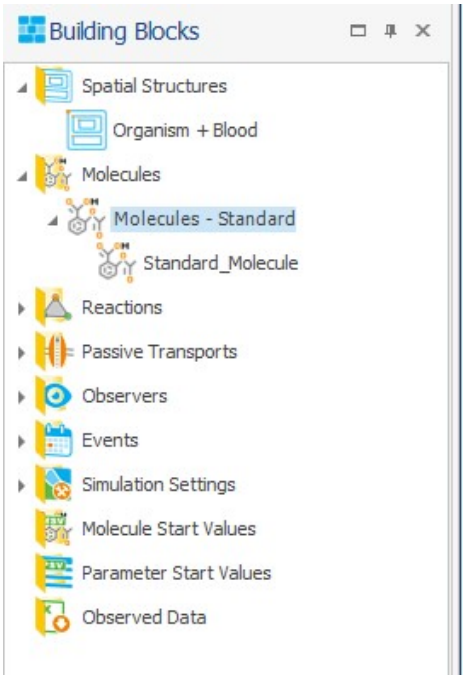
6.0 Inhalation PBPK Model Walk Through in Mobi

6.1 PBPK Model Set Up and Simulation in Mobi

This computational procedural highlight will outline a step-by-step implementation of a buprenorphine inhaled simulation using the inhalation PBPK model connected to a two-compartment model. This inhalation model will require both MoBi and R with the ospsuite R package (as well as with accompanying dependencies) installed. Note that files mentioned below may be referenced without their version number as these files may be updated in the future.

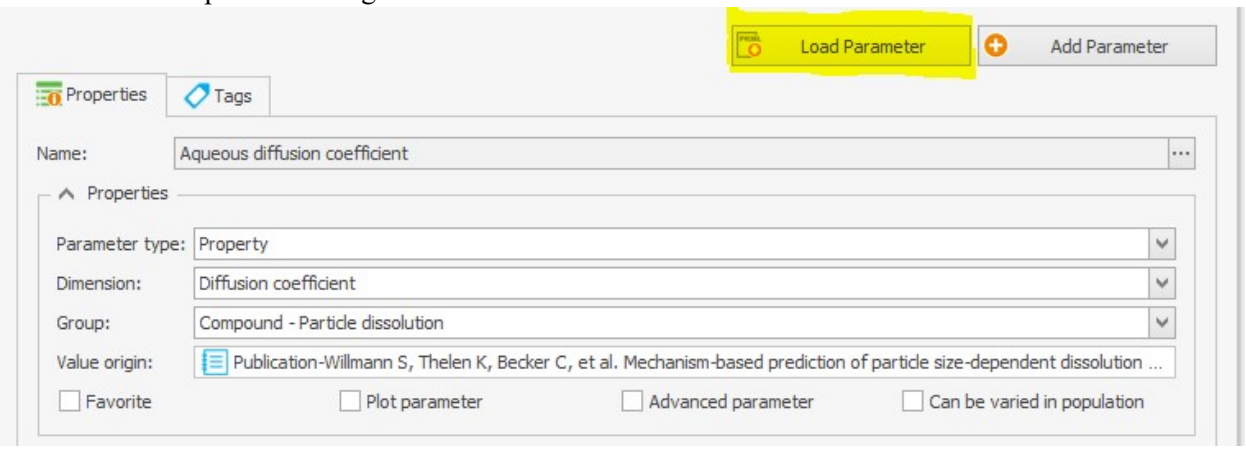
1. Download the MoBi project file **inhalation_model_two_compt_1_bin.mbp3** and rename it **Buprenorphine.mbp3**. Open the MoBi project file.

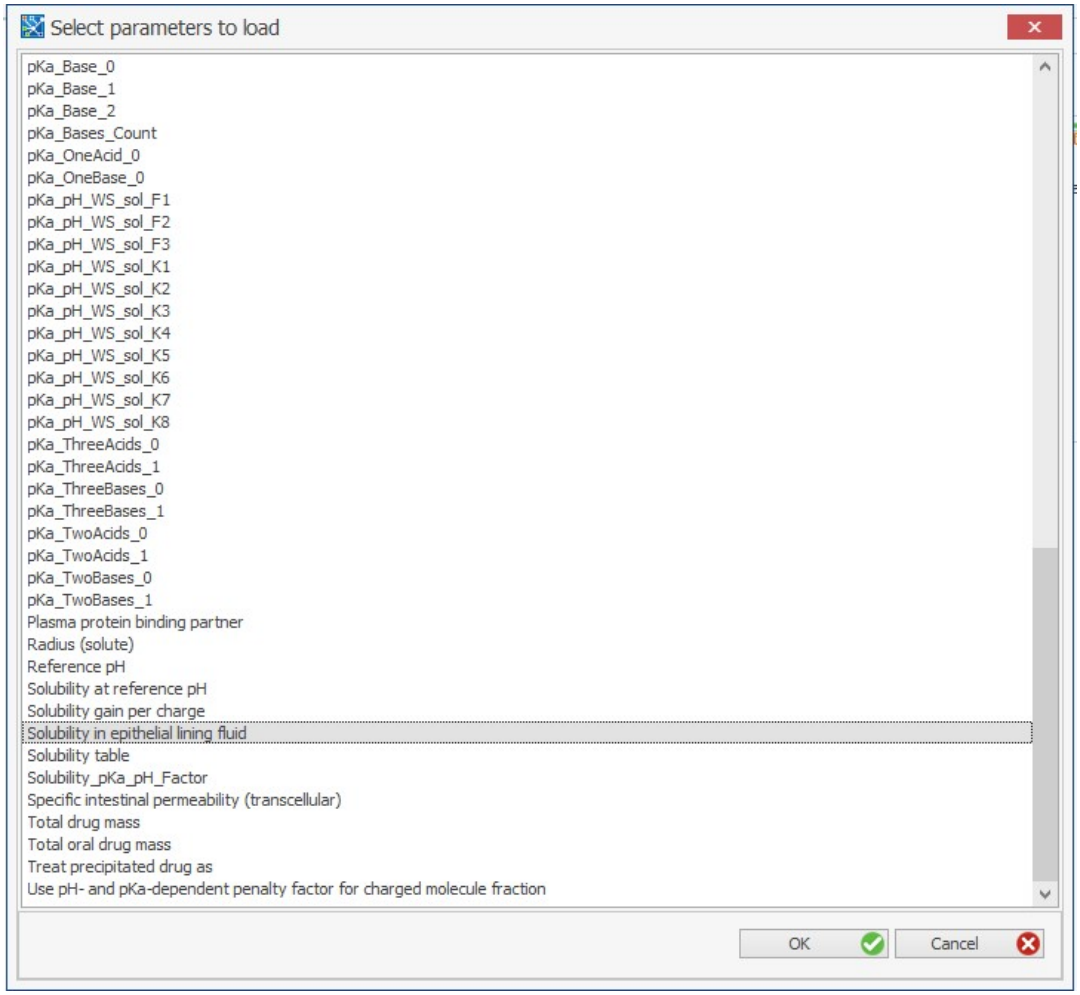
2. Set up the molecule of interest, i.e. Ciprofloxacin.
    a. Right-click the current Molecules building block "Molecules – Standard" and **Delete…** this building block.



    b. Right-click the "Molecules" building block folder and **Create Molecule Building Block…** and call the new molecule building block "Molecules".
    c. In the white space below **Favorites** and **User Defined**, right-click then **Add PK-Sim Molecule…** and call the new PK-Sim Molecule "Buprenorphine". Fill in the physicochemical properties of Buprenorphine as given below. These physicochemical properties will be used to set up the other molecule parameters.
        i. **Molecular weight**: 467.64
        ii. **Lipophilicity**: 4.98 Log units
        iii. **Fraction unbound (plasma, reference value)**: 0.04
        iv. **Solubility at reference pH**: 0.0168 mg/mL
    d. Next, we need to add a parameter to the molecule building block that is specific to the inhalation model. Open the **Buprenorphine** molecule building block and navigate to **Parameters**. Download the **Standard_Molecule.pkml** file, which contains the parameter that we need. Click **Load Parameter** and navigate to where you have downloaded the **Standard_Molecule.pkml** file and select this file. Then navigate down to "Solubility in

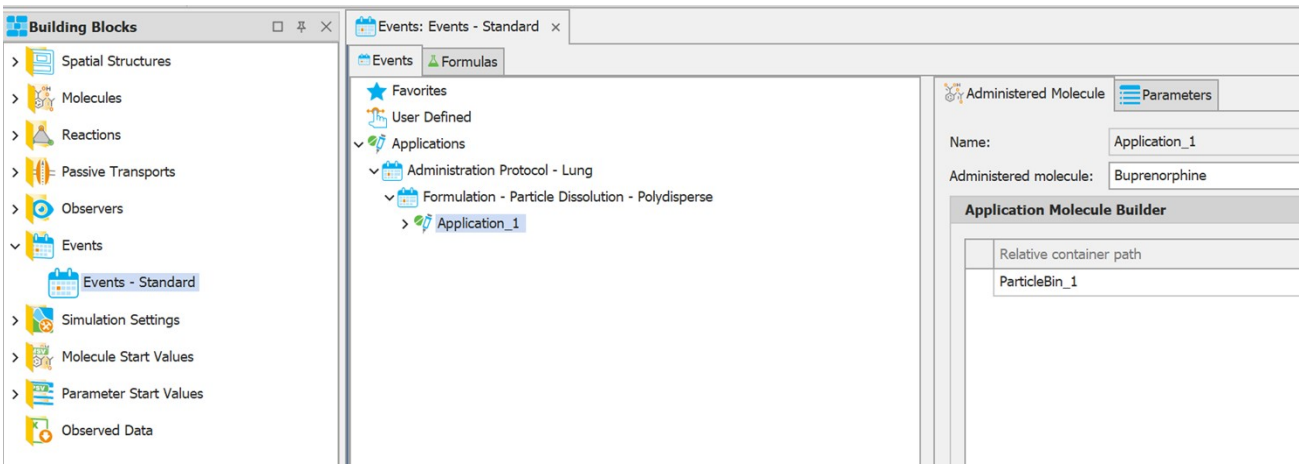
epithelial lining fluid" and click OK.





e. Next, we set up the Events building block to work with the Buprenorphine molecule that has just been created.

- **i.** Open the **Events – Standard** building block under the "Events" building block folder.
- **ii.** Open up the tree and navigate to **Application_1** .
- **iii.** Change the **Administered molecule** to Buprenorphine.



3. Fill in the parameters for the two compartment model and oral absorption. These parameter values are the result of fitting the two compartment model to Buprenorphine Sublingual Data.
    a. Right-click the "Parameter Start Values" building blocks folder and **Create Parameter Start Values Building Block…**, and name this parameter start values building block "Buprenorphine Parameters" with the following configuration:
        i. **Molecule building block**: Molecules
        ii. **Spatial structure**: Organism + Blood
    b. Update the following parameter values within this parameter start values building block:
        i. **Absorption rate – k_a:** 3.13 1/h
            - Note that this refers to oral absorption.
        ii. **Oral bioavailability – F_oral**: 0.51
        iii. **Time lag of absorption:** 0.22 h
        iv. **Volume [Path Element 1: CentralCompartment]**: 1225.0L
        v. **Elimination rate**: 0.12 1/h
        vi. **k_CP – First-order constant for transfer from central to peripheral compartments**: 0.354 1/h
        vii. **Volume [Path Element 1: PeripheralCompartment]**: 6225.01 L
        viii. **k_PC – First-order constant for transfer from peripheral to central compartments**: 0.05 1/h

Parameter Name	Path Element 0	Path Element 1	Path Element 2	Path Element 3	Path Element 4	Start Value
Use pH- and pKa-dependent penalty factor for charg...	Buprenorphine					0
Kp_u_lung - Unbound tissue-plasma partition coeffici...	Organism	Lung				6.50
Fraction unbound in the epithelial lining fluid	Organism	Lung				1.00
OralApplicationsEnabled	Organism	Lung				1.00
Volume	Organism	ExtrathoracicRegion				1.00 l
Absorption rate - k_a	Organism	ExtrathoracicRegion				0.42 1/h
Oral bioavailability - F_oral	Organism	ExtrathoracicRegion				0.51
Time lag of absorption	Organism	ExtrathoracicRegion				0.14 h
Volume	Organism	CentralCompartment				1225.00 l
Blood-plasma ratio	Organism	CentralCompartment				1.00
Cardiac output	Organism	CentralCompartment				5.20 l/min
Elimination rate	Organism	CentralCompartment				0.12 1/h
k_CP - First-order constant for transfer from central t...	Organism	CentralCompartment				0.35 1/h
Volume	Organism	PeripheralCompartment				6225.01 l
k_PC - First-order constant for transfer from peripher...	Organism	PeripheralCompartment				0.05 1/h
Volume	Organism	ClearedDrug				203.00 l

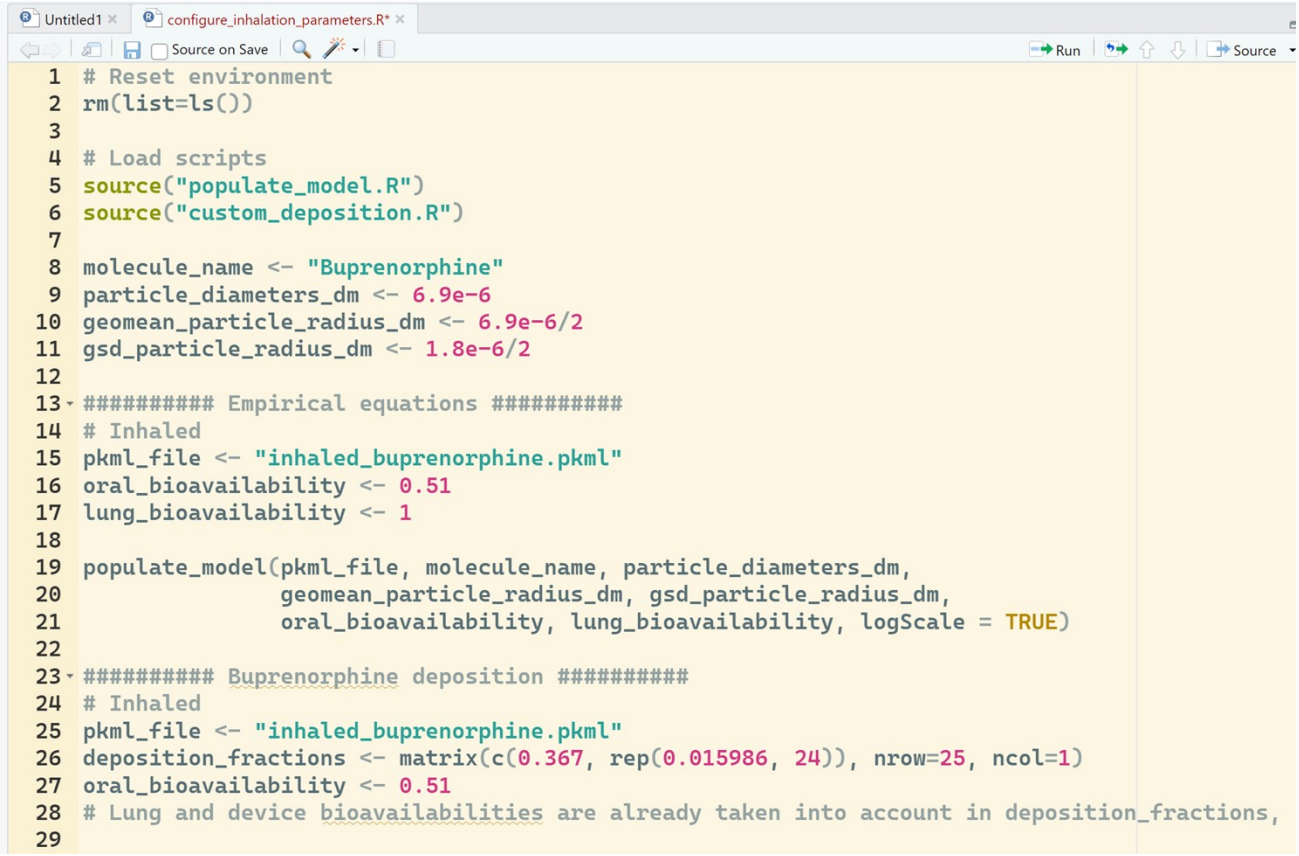
4. Set up the molecule start values building block for the simulation.
    a. Right-click the "Molecule Start Values" building block and **Create Molecule Start Values Building Block…** , and name this molecule start values building block "Buprenorphine MSV" with the following configuration:
        i. **Molecule building block**: Molecules
        ii. **Spatial structure**: Organism + Blood
    b. The inhaled (Sublingual) dose will not be configured here but in the **Events** building block at a later step. However, a molecule start values building block is required to create a simulation. Thus, we Utilze this empty one.
5. The crux of the PBPK Model Setup & implementation considers the monodisperse case, i.e. all particle sizes are of the same size. At this step in the simulation setup process, the user can optionally add particle bins if considering a polydisperse case, i.e. having particles of different sizes.
6. Set up the inhalation simulation and export the simulation.
    a. Right-click the "Simulations" folder and **Create Simulation…** , and name the simulation "Inhaled Buprenorphie" with the following configuration. Note that this should already be the configuration since these are the only building blocks available:
        i. **Spatial Structure:** Organism + Blood
        ii. **Molecules:** Molecules
        iii. **Reactions:** Reaction
        iv. **Passive Transports:** Passive Transports
        v. **Observers:** Observer
        vi. **Events:** Events - Standard
        vii. **Simulation Settings:** Simulation Settings 1
        viii. **Molecule Start Values:** Buprenorphine MSV
        ix. **Parameter Start Values:** Buprenorphine Parameters
    b. A simulation run execution at this stage will NOT produce any results. This is because the administration of inhaled drug has not been configured yet. This will be done outside of MoBi in R. To do this, we will need to export the pkml file.
        i. Right-click the simulation and **Save Simulation to MoBi pkml file Format…** and save as **inhaled_buprenorphine.pkml** .
7. In R, use the **populate_model.R** and **configure_inhalation_parameters.R** scripts included with the inhalation model to populate the initial conditions of the simulation. You may do this with any code editor or IDE (integrated development environment), but for yhe purpose of this modelling endeavor, the instructions and screenshots are from using RStudio.
    a. Download the files **populate_model.R** , **custom_deposition.R** , and **configure_inhalation_parameters.R** into the same directory.

    b. Open RStudio and navigate to the folder where you have saved **inhaled_buprenorphine.pkml** .

    c. The following arguments will need to be configured in order to run the **populate_model.R** script. Note that these values have already been configured in the **configure_inhalation_parameters.R** script.
        i. Pkml file name : **inhaled_buprenorphine.pkml**
        ii. Molecule name : Buprenorphine
        iii. Particle diameters of each bin (in dm) : 6.9e-6
        iv. Geometric mean of particle radius (in dm) : 6.9e-6/ 2

v. Geometric standard deviation of particle radii (in dm) : 1.8e-5 / 2
vi. Oral bioavailability : 0.51
vii. Lung bioavailability : 1
viii. Log-normal distribution : TRUE
ix. Note that there are additional parameters that can be configured using the **populate_model.R** script including breathing frequency, fraction of breath that is inspiratory, breath hold time, delay volume, tidal volume, and bolus volume.

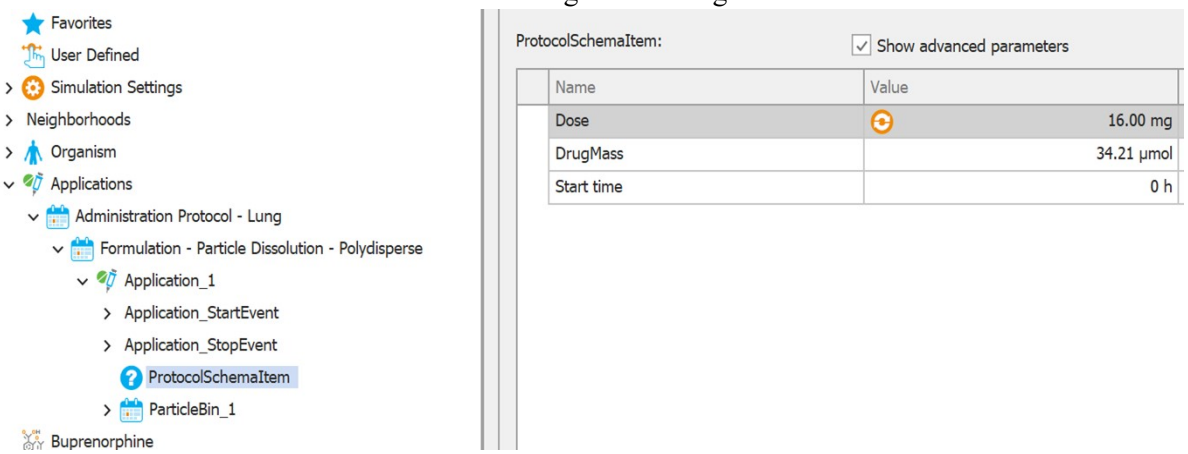
d. Run the first 21 lines of the **configure_inhalation_parameters.R** script by highlighting the code and clicking **Run**. Alternatively, you can run each individual line of code by pressing **Ctrl + Enter** while your cursor is on each line of code.



e. Note that in the output, there are 25 generations shown. This is because the extrathoracic region is included here as the first generation. Additionally, you should find that a pkml file has been generated called **populated_inhaled_buprenorphine.pkml .** This is the updated simulation file.

f. Note that this process will need to be repeated and a different file generated for every particle size distribution under consideration.

8. Load the newly generated file **populated_inhaled_buprenorphine.pkml** into MoBi.
    a. Double-click the newly generated file **populated_inhaled_buprenorphine.pkml** and MoBi will open with the new simulation already loaded.
    b. **Important note:** Only the simulation parameters have been updated. Thus, do NOT update the simulation building blocks from the individual building blocks as they will over-write the parameters that have been updated by the R script.
9. Update the dose within the simulation, run the simulation, and compare to observed data.
    a. Open the simulation **Inhaled buprenorphine** and click the **Parameters** tab and the **Tree** view.
    b. Navigate to the dose by expanding the following hierarchy: **Applications → Administration Protocol – Lung → Formulation – Particle Dissolution – Polydisperse → Application_1 → ProtocolSchemaItem** .
    c. Click the **Parameters** tab in the adjacent window to the right. For the parameter **Dose**, update the value to 16mg. Note that this change has only been made in the simulation and not to the original building block.



    d. Navigate to the **Results** tab, and click **Define Settings and Run**. Scroll down to the central compartment concentration (see screenshot below) and check the corresponding box. Click OK. The simulation will then run and produce a plot of the concentration in the central compartment following an inhaled dose.

Drag a column header here to group by that column

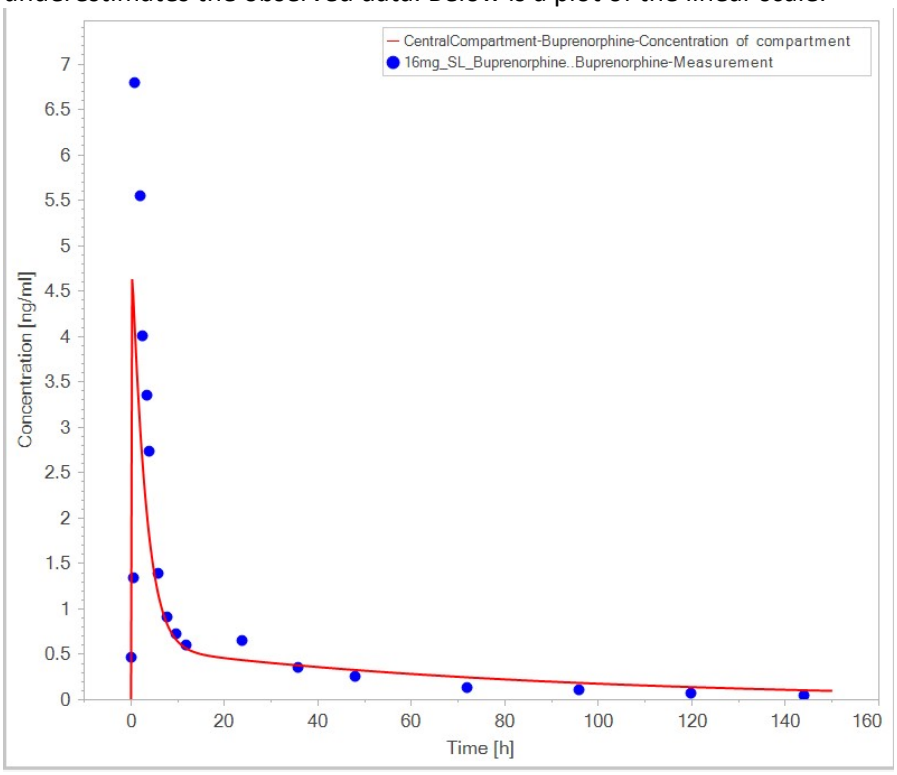
Simulation	Top Container	Container	Compartment	Molecule	Name	Dimension	QuantityType	Selected
Inhaled B...	Organism	Lung-Alve...	Subepithel...	Buprenorp...	Amount in con...	Amount	Drug	☐
Inhaled B...	Organism	Lung-Alve...	Subepithel...	Buprenorp...	Concentration ...	Concentration ...	Drug, Observer	☐
Inhaled B...	Organism		Extrathora...	Buprenorp...	Amount in con...	Amount	Drug	☐
Inhaled B...	Organism		Extrathora...	Buprenorp...	Concentration ...	Concentration ...	Drug, Observer	☐
Inhaled B...	Organism		CentralCo...	Buprenorp...	Amount in con...	Amount	Drug	☐
Inhaled B...	Organism		CentralCo...	Buprenorp...	Concentration ...	Concentration ...	Drug, Observer	☑
Inhaled B...	Organism		Peripheral...	Buprenorp...	Amount in con...	Amount	Drug	☐
Inhaled B...	Organism		Peripheral...	Buprenorp...	Concentration ...	Concentration ...	Drug, Observer	☐
Inhaled B...	Organism		ClearedDrug	Buprenorp...	Amount in con...	Amount	Drug	☐
Inhaled B...	Organism		ClearedDrug	Buprenorp...	Concentration ...	Concentration ...	Drug, Observer	☐
Inhaled B...	Applications	Administra...		Buprenorp...	Amount in con...	Amount	Drug	☐
Inhaled B...	Applications	Administra...		Buprenorp...	Concentration ...	Concentration ...	Drug, Observer	☐

Drag a column header here to group by that column

Simulation	Top Container	Compartment	Molecule	Name	Dimension	QuantityType	Selected
Inhaled Bupr...	Organism	CentralComp...	Buprenorphine	Concentration of ...	Concentration (m...	Drug, Observer	☑

e. Upon running the simulation and obtaining the output (a plasma concentration time profile, it is overlayed with the observed data. This is done by Right clicking the "Observed Data" building block folder and **Add Observed Data…** The digitized *.csv* file for the Dong et al (16mg Single Ascending Dose) Study is then loaded.

i. After Loading the data successfully, drag the observed data building block onto the simulation plot to add the observed data for comparison. The corresponding visualization depicts that the inhalation simulation underestimates the observed data. Below is a plot of the linear scale.



ii. You can now Close this Mobi File and save it if you wish.

10. To improve this simulation, we will now make use of additional deposition information

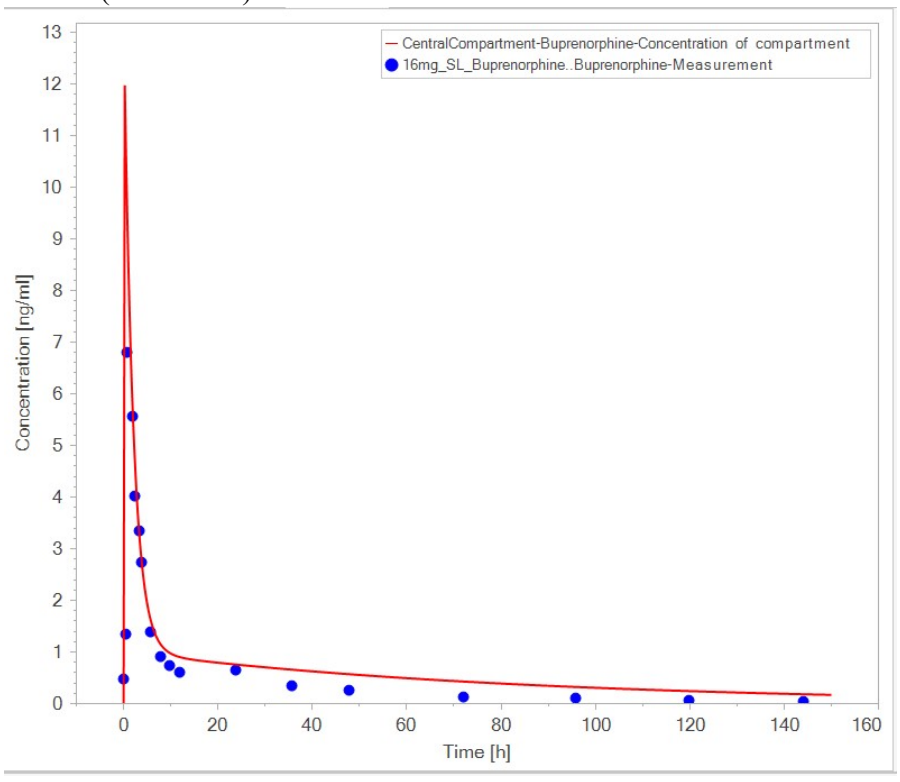
that we have and use the **custom_deposition.R** script to use that information instead of the empirical equations used in the **populate_model.R** script.

a. Again, the following arguments will need to be configured in order to run the **populate_model.R** script. Note that these values have already been configured in the **configure_inhalation_parameters.R** script.
   - **i.** Pkml file name : **inhaled_buprenorphine.pkml**
   - **ii.** Molecule name : Buprenorphine
   - **iii.** Particle diameters of each bin (in dm) : 6.9e-6
   - **iv.** Mean particle radius (in dm) : 6.9e-6 / 2
   - **v.** Standard deviation of particle radii (in dm) : 1.8e-5 / 2
   - **vi.** Oral bioavailability : 0.51
   - **vii.** Deposition fractions (note that this is configured as a matrix of values as in the **configure_inhalation_parameters.R** script):
     - Extrathoracic proportion : 0.367
     - Proportion in each of 24 lung generations : 0.015986
       a. Total lung proportion: 0.381

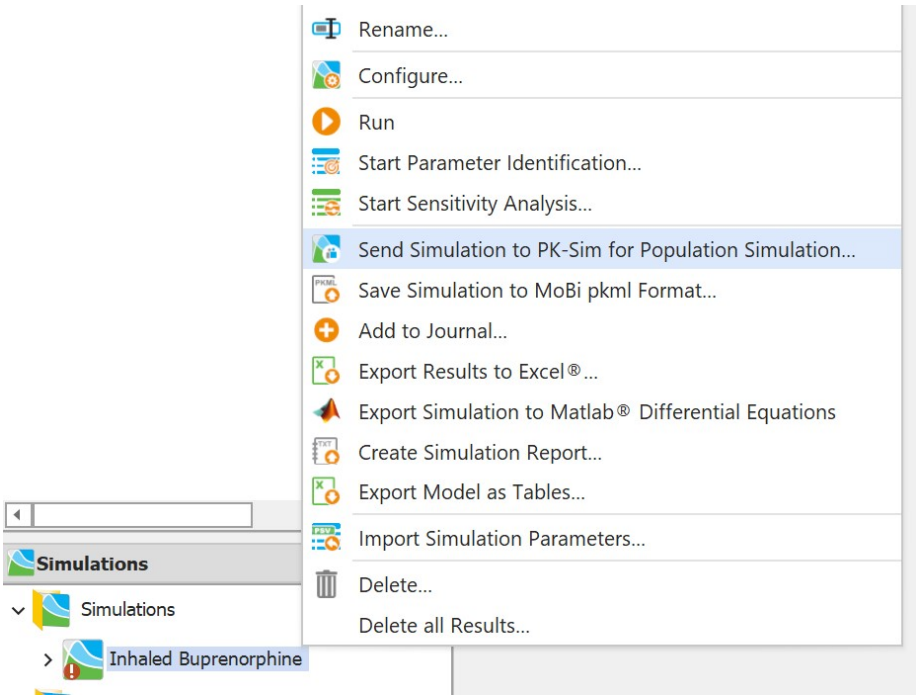
b. Run lines 1-11 and lines 24-31 of the **configure_inhalation_parameters.R** script by highlighting the sections of the code and clicking Run. Alternatively, you can run each individual line of code by pressing Ctrl + Enter while your cursor is on each line of code.

c. As with the **populate_model.R** script, there are 25 generations shown. This is because the extrathoracic region is included here as the first generation. Additionally, you should find that a pkml file has been generated called **custom_inhaled_buprenorphine.pkml .** This is the updated simulation file.

11. Repeat steps 8 and 9 with the newly generated file **custom_inhaled_buprenorphine.mbp3** . Below, we show the simulation results (linear scale) from this file.



12. Following the visualization of the file in step 11, a Parameter Identification run is executed by Right clicking on the **"Parameter Identification"** module in the Simulation building block and Add Parameter Identification.
    i. A new window is opened where the "**Data"** from the Inhaled Buprenorphine simulation output is linked (with the overlayed data). Further a Scaling preference & weighting are calibrated.
    ii. In the "**Parameters"** window, relevant parameters that are Ideal to the Parameter Identification for the inhalation PBPK Model are selected (LHS) & parsed (RHS)
    iii. In the Configuration Window, an algorithm is selected based on the optimization protocol & iteration maximum time. The Algorithm for the purpose of this work is the Monte Carlo and is set at 1000 iterations. Other algorithmic options are left **as is**.
    iv. After the Calibration, the "RUN" Icon (an orange PLAY icon) in the Upper LHS of the window is activated & the Parameter identification process follows.
13. After the Parameter ID Process, the optimized parameters are transferred to the Simulation by **"Clicking on the Transfer to Simulation Green Tick Icon"** on the RHS of the Parameter Identification Results window.
14. The Simulation is Re-run and compared against the Observed data (with further Optimization computationally executed as need).
15. After this, the simulation is sent to PKSim (by **Right clicking on the Simulations folder in the Simulations Building Block, Clicking on the Send Simulation To PKSim for Population Simulations** and further simulated on created population representatives of the study under consideration.



References.

Bernard, S.L., Glenny, R.W., Polissar, N.L., Luchtel, D.L., Lakshminarayan, S., 1996. Distribution of pulmonary and bronchial blood supply to airways measured by fluorescent microspheres. Journal of Applied Physiology 80, 430–436. https://doi.org/10.1152/jappl.1996.80.2.430

Boger, E., Wigström, O., 2018. A Partial Differential Equation Approach to Inhalation Physiologically Based Pharmacokinetic Modeling. CPT Pharmacometrics Syst Pharmacol 7, 638–646. https://doi.org/10.1002/psp4.12344

Brown, R.P., Delp, M.D., Lindstedt, S.L., Rhomberg, L.R., Beliles, R.P., 1997. Physiological parameter values for physiologically based pharmacokinetic models. Toxicol Ind Health 13, 407–484. https://doi.org/10.1177/074823379701300401

Chamarthy, M.R., Kandathil, A., Kalva, S.P., 2018. Pulmonary vascular pathophysiology. Cardiovascular Diagnosis and Therapy 8, 20813–20213. https://doi.org/10.21037/cdt.2018.01.08

Combs, M.P., Dickson, R.P., 2020. Turning the Lungs Inside Out: The Intersecting Microbiomes of the Lungs and the Built Environment. Am J Respir Crit Care Med 202, 1618–1620. https://doi.org/10.1164/rccm.202007-2973ED

Darquenne, C., 2020. Deposition Mechanisms. J Aerosol Med Pulm Drug Deliv 33, 181–185. https://doi.org/10.1089/jamp.2020.29029.cd

Pellowe, M., 2023. Inhalation-model/configure_inhalation_parameters.R at main · Open-Systems-Pharmacology/Inhalation-model.

Thelen, K., Coboeken, K., Willmann, S., Burghaus, R., Dressman, J.B., Lippert, J., 2011. Evolution of a detailed physiological model to simulate the gastrointestinal transit and absorption process in humans, Part 1: Oral solutions. JPharmSci 100, 5324–5345. https://doi.org/10.1002/jps.22726

Weibel, E.R., 1963. Geometry and Dimensions of Airways of Conductive and Transitory Zones, in: Weibel, E.R. (Ed.), Morphometry of the Human Lung. Springer, Berlin, Heidelberg, pp. 110–135. https://doi.org/10.1007/978-3-642-87553-3_10

Zhang, L., Asgharian, B., Anjilvel, S., 1997. Inertial Deposition of Particles in the Human Upper Airway Bifurcations. Aerosol Science and Technology 26, 97–110. https://doi.org/10.1080/02786829708965417