

12 In silico-Modellierung mit MEDICI-PK

Im Programmpaket MEDICI-PK ist sowohl das in Kapitel (10) beschriebene Kalkül zur Aufstellung des Gesamtdifferentialgleichungssystems zur Beschreibung der Konzentrationsverläufe von Wirkstoffen innerhalb eines Gesamtorganismus als auch die Idee der orthogonalen Parameterbasen implementiert. Dies läßt sich anhand des Codes verifizieren, wobei sich natürlich nicht allein an Datenstrukturen verfolgen lässt, ob die beschriebene Intelligenz ebenso Eingang gefunden hat in den Code. Dies wollen wir aber nun voraussetzen. Neben der Umsetzung eines Algorithmus ist es auch Aufgabe eines Softwaretools, dem Benutzer mindestens die Strukturen zur Verfügung zu stellen, die notwendigerweise definiert werden müssen bzw. mit Inhalt gefüllt werden können. Wir werden im folgenden aufzeigen, in welcher Form die beschriebenen Strukturen in MEDICI-PK wiederzufinden sind. Da wir die Bedeutung der Strukturen bereits dargelegt haben, liegt hier der Schwerpunkt auf implementationstechnischen Aspekten.

12.1 Zuordnung der Strukturen

Folgende Begriffe, die wir bisher eingeführt haben, entsprechen bestimmten Bezeichnungen im MEDICI-PK-Kontext. Für die Strukturen, die oberflächenmäßig verfügbar sind, werden wir die Präsentation in MEDICI-PK-Dialogen zeigen.

Struktur	MEDICI-PK-Realisierung
Subkompartiment (Verteilungsraum) (Kap. 9.1)	Verteilungsraum, kleinste Betrachtungseinheit, in der Konzentrationsverläufe berechnet werden, siehe Kap. 12.1.1
Kompartimentstruktur (Kap. 10.2, Definition 16)	Kompartiment, siehe Kap. 12.1.2
Vollständig orthogonale Kompartimentierung, Kompartimenttopologie (Kap. 9.2.3)	Topologie, siehe Kap. 12.1.3
Elementarer Prozess (Kap. 9.2.4)	Modell, siehe Kap. 12.1.4
Modellbasis (Kap. 10.4.1, Definition 19)	Modellbasis, siehe Kap. 12.1.5
Parameterbasis (Kap. 11, Definition 21)	Parameterbasis, siehe Kap. 12.1.6
Vollständiges Körpermodell (Kap. 10.4, Definition 18)	FullBodyTemplate, siehe Kap. 12.1.7
Dosierung (Kap. 10.1.5)	Dosierung, siehe Kap. 12.1.8
Wirkstoff(objekt), Compound (Kap. 11, Definition 26)	Compound, siehe Kap. 12.1.9
Individual (Kap. 11, Definition 27)	Individual, siehe Kap. 12.1.10
Mix parameter set (Kap. 11, Definition 29)	Mix parameter, siehe Kap.12.1.11
General parameter set (Kap. 11, Definition 28)	General parameter, siehe Kap. 12.1.11
Gesamtdifferentialgleichungssystem (Kap. 10.5)	Simulationsobjekt, siehe Kap. 12.2

12.1.1 Subkompartiment (Verteilungsraum)

Subkompartimente sind in MEDICI-PK die betrachteten Verteilungsräume. Es gibt eine offene Liste von Subkompartimenten, die zur Zeit aus den 4 Subkompartimenten Blood Cell, Plasma, Interstitial und Cellular gebildet wird. Nur Blood Cell und Plasma sind als flowfähig gekennzeichnet, d.h. nur aus diesen wird ein Fluss von Wirkstoff von einem Kompartiment in ein anderes installiert. Subkompartimente existieren nur innerhalb eines Kompartiments. Jedes Kompartiment hat dieselben Subkompartimente; nur diese sind oberflächenmäßig innerhalb eines FBTs (siehe 12.1.7) für den Benutzer zugänglich.

12.1.2 Kompartiment

Ein Kompartiment enthält eine Liste von Subkompartimenten. In der Regel sprechen wir von den Organen als den Kompartimenten, da dies im Allgemeinen am anschaulichsten ist, und bezeichnen die Kompartimente im Programm auch als „organs“. Die Kompartimente werden innerhalb eines MEDICI-PK-Projektes beliebig angelegt und benannt. Es ist daher möglich, Kompartimente anzulegen, die eine Zusammenfassung von Organen meinen (Clusterbildung wie gutdurchblutete Organe, schlecht durchblutete Organe, Magen-Darm-Trakt) oder Kompartimente anzulegen, die nur bestimmte Teile eines Organs meinen (obere Muskelschicht, untere Muskelschicht). Über die Namen der Kompartimente werden die Zuordnungen der Werte von organ- bzw. kompartimentabhängigen Parametern ermöglicht.

Ebenfalls aus Gründen der Anschaulichkeit gibt es unterschiedliche Typen von Kompartimenten: den „org“-Typ, der ein Standardkompartiment bezeichnet, und die Typen „arterial_blood“ bzw. „venous_blood“. Kompartimente diesen Typs besitzen nur die zwei flowfähigen Subkompartimente Blood cell und Plasma.

12.1.3 Topologie

Aus allen oder einem Teil der definierten Kompartimente (siehe 12.1.2) kann eine (nahezu) beliebige Verschaltung angegeben werden. Dies bezeichnen wir im Programm als „topology template“. Dazu werden jedem teilnehmenden Kompartiment die Kompartimente angegeben, die einen Zufluss liefern, oder zu denen ein Zufluss existiert. Die Kompartimentierung in die Breite kann damit wie gewünscht, d.h. mit den Kompartimenten, die man betrachten will, angelegt werden, die Kompartimentierung in die Tiefe ist implizit durch die Liste der Subkompartimente gegeben.

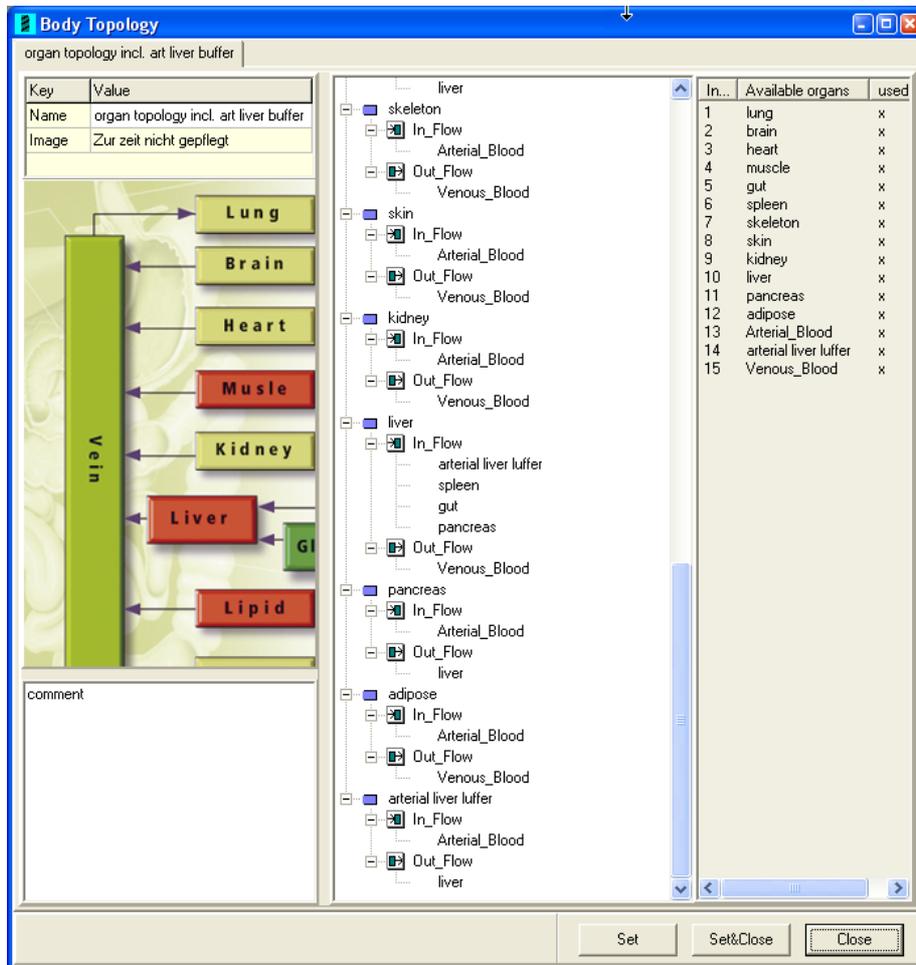


Abbildung 19: Im Dialog zur Definition einer Topologie kann eine Topologie abhängig von einer beliebigen Auswahl an Kompartimenten gebildet werden

In Abbildung 19 sind rechts die definierten Kompartimente/Organe aufgelistet. Sie bzw. ein Teil von ihnen bilden die Topologie (Mitte).

12.1.4 Modell

Ein Modell ist die benutzerdefinierte Darstellung eines Elementarprozesses. Es enthält im Wesentlichen die rechte Seite einer/mehrerer Differentialgleichung(en) in Form eines Skripts. Der/Die Rückgabewert(e) `VdComp1/2/3` gibt den Beitrag an, den der elementare Prozess, der durch dieses Modell modelliert wird, an der Gesamtdifferentialgleichung für einen/die Wirkstoff(e) hat. Das Skript eines Modells kann in Abhängigkeit von Parameterbezeichnern und Konzentrationen von Wirkstoffen ausgedrückt werden.

Dieser Ausdruck wird zur Laufzeit des Programms ausgewertet (siehe dazu auch Interpretationmechanismus, 12.4.1). Damit ein Modell in identischer Form an verschiedenen Stellen (eines Organismus) interpretiert werden kann, darf es nicht von absoluten Bezeichnern abhängig sein, sondern wird in Abhängigkeit von Bezeichnern `Comp1/2/3` und Parameterbezeichnern formuliert. Bei der Auswertung ist intern immer klar, um welche Konzentrationen und Parameterwerte es sich handelt (siehe dazu auch 12.2.3).

Jedes Modell hat bestimmte Eigenschaften: Name, Typ (Transfer, Bindung, Metabolismus, Dosierung), Anzahl zu behandelnder Wirkstoffe, Systembiol. Größen und ihre Differentialgleichungen. Im unteren Beispiel (Abbildung 20) ist ein 1-Wirkstoff-Modell gezeigt, d.h. dieses Modell beschreibt die Massenänderung pro Zeit für einen Wirkstoff, der unabhängig von anderen Wirkstoffen reagiert. Das Modell zeigt aber auch eine systembiologische Größe (mit `Comp1_bound` vom Benutzer eingeführt), die hier als Puffer für den Wirkstoff eingesetzt wird:

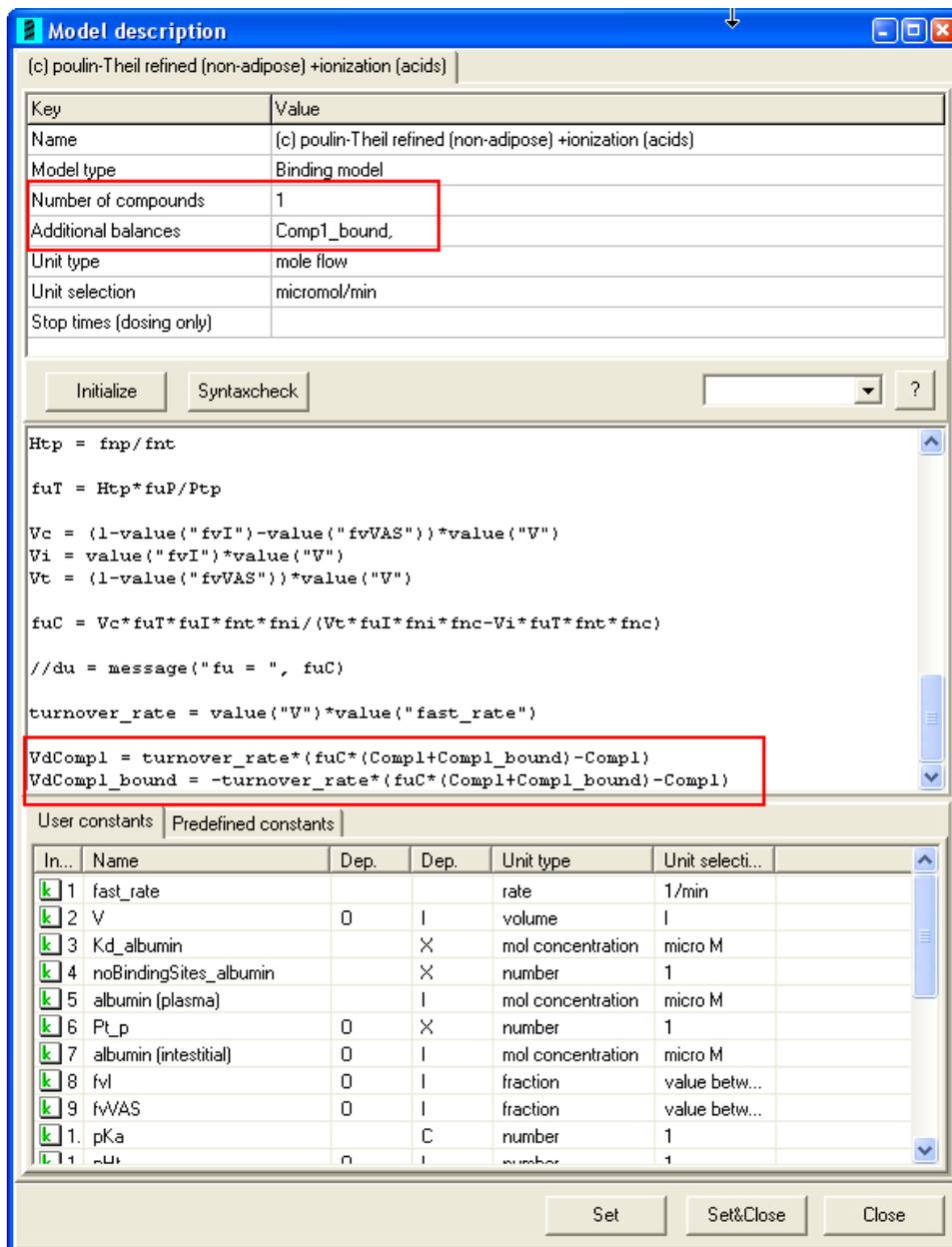


Abbildung 20: Im Dialog zur Definition eines Modells findet die Eingabe der rechten Seite der Differentialgleichung in unabhängiger Form statt

Die Parameter, welche im Modell verwendet werden, werden aus Implementationsgründen aus der Parameterbasis geholt und angemeldet (in Abbildung 20 die untere Liste). Für jeden Parameter ist der Einheitentyp durch die Deklaration in der Parameterbasis gegeben, es muss aber angegeben werden, in welcher konkreten Einheit er innerhalb der Modellbeschreibung benutzt wird. Der Benutzer ist für die Konsistenz seines Skriptes bzgl. der Einheiten eigenverantwortlich.

Ein Beispiel für die kompartmentunabhängige Formulierung eines Modells ist schematisch in Abbildung 21 gezeigt.

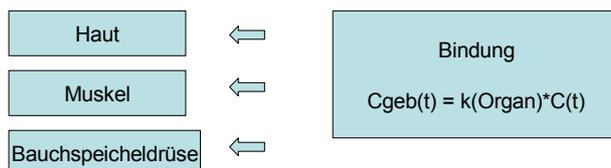


Abbildung 21: Beispiel für abstrakte Modellierung: das Bindungsmodell kann in dieser Form sowohl für Haut als auch Muskel, Bauchspeicheldrüse usw. verwendet werden

Angenommen, im interstitiellen Raum von Haut, Muskel und Bauchspeicheldrüse wird die Bindung eines Wirkstoffes an Proteine in der Form $C_{geb}(t) = k(\text{Organ}) \cdot C(t)$ gegeben mit einem physiologisch erkannten organabhängigen Parameter k , der in der Haut etwa den Wert 0,2, im Muskel den Wert 0,3 und in der Bauchspeicheldrüse den Wert 0,23 hat. Dann ist es möglich, diese Beschreibung über eine Interpreterfunktion anzugeben. Der Wert von $k(\text{Organ})$ muss dabei abhängig vom Ort der Auswertung bestimmt werden.

12.1.5 Modellbasis

In der Modellbasis werden alle Modelle gesammelt. Sie existiert außerhalb von sogenannten MEDICI-PK-Projektdateien, die jeweils in sich abgeschlossene Projekte darstellen. Innerhalb eines FBT werden die hier angelegten Modelle zur Darstellung der Elementarprozesse Transfer, Bindung, Metabolismus Dosierung angeboten.

12.1.6 Parameterbasis

In der Parameterbasis werden Parameter gesammelt. Die Parameterbasis existiert wie die Modellbasis außerhalb von sogenannten MEDICI-PK-Projektdateien. Jeder Parameter definiert sich über seinen Namen; innerhalb der Parameterbasis hat er keinen Wert. Die Modelle werden von Parametern (sowie Konzentrationen und Zeit) abhängen. Je besser man diese Parameter als physiologische oder wirkstoffspezifische Parameter identifizieren kann, um so sinnvoller werden die Modelle sein. Die Parameter haben demnach eine Bedeutung und können eingeordnet werden als unabhängig, organabhängig, wirkstoffabhängig, individuumabhängig, oder auch als Kombination aus diesen Attributen. Die Software muss diese Eigenschaften abfragen und die Bearbeitung der Parameter von diesen Eigenschaften abhängig gestalten, d.h. z.Bsp., dass

ein Parameter, welcher organabhängig ist, auch tatsächlich für jedes Organ einen eigenen Wert (und eine eigene Einheit) verwaltet, während ein organunabhängiger Parameter nur einen Wert (mit Einheit) trägt.

12.1.7 Vollständiges Körpermodell (FullBodyTemplate)

Das FullBodyTemplate ist ein Muster oder Formular, welches auf Basis einer ausgewählten Topologie die Modelle für jeden Elementarprozess in jedem Subkompartiment verwaltet. Hierzu werden dem Modellierer die 4 (zur Zeit vorgegebenen) Subkompartimente angeboten (siehe Abbildung 22). Jedem Subkompartiment kann er seine Modelle für die Prozesse Transfer, Bindung und Metabolismus zuordnen. Dies muss für alle Kompartimente einer zuvor definierten Topologie durchgeführt werden. Als Unterstützung der Eingabe wird die Zuordnung Kompartiment - Modelle für ein allgemeines Kompartiment („Do for all“) durchgeführt; nach der Auswahl einer Topologie wird diese Zuordnung entsprechend vervielfältigt, d.h. jedes Kompartiment hat dieselbe Modellzuordnung. Nun kann für speziell ausgezeichnete Kompartimente die Modellierung abweichend von der allgemeinen Auswahl gewählt werden.

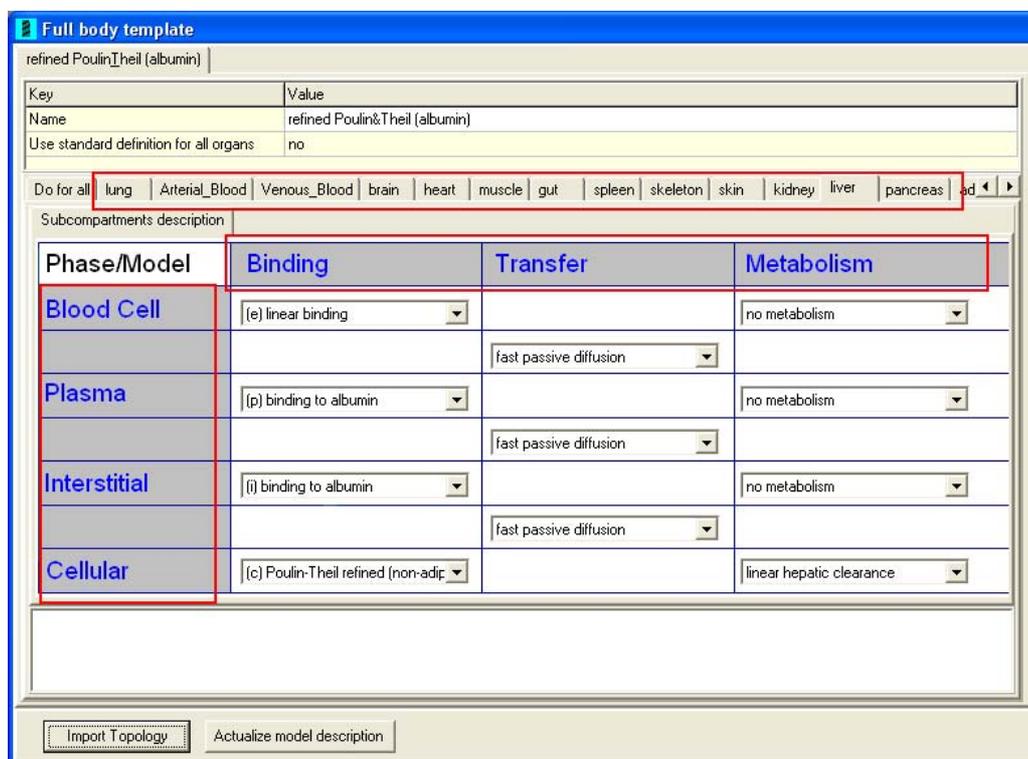


Abbildung 22: Dialog zur Definition eines FBT (FullBodyTemplate): für jedes Organ können pro Subkompartiment Modelle zur Modellierung der physiologischen Effekte zugeordnet werden

Ein FullBodyTemplate hat damit die vollständige Information, wie die Konzentrationsverläufe von Wirkstoffen modelliert werden sollen, abhängig von Konzentrationen und von Parametern. Zur Simulation fehlt ihm die Dosierung sowie die Zuordnung von Parameterwerten! Ein FullBodyTemplate kann anhand der ausgewählten Modelle entscheiden, wieviele Wirkstoffe an seinen Modellen beteiligt sein müssen.

12.1.8 Dosierung

Eine Dosierung ist die Beschreibung der Zugabe eines Wirkstoffs. Dazu gehören Informationen über die Menge an Wirkstoff sowie die Startzeit, der Verlauf der Zugabe und die Wiederholung. Eine Dosierung wird als Modell angelegt und kann daher auch funktionelle Abhängigkeiten und Verläufe darstellen. Jeder Wirkstoff kann dem Organismus über mehrere verschiedene Dosierungen zugeführt werden.

12.1.9 Wirkstoff(objekt), Compound

Ein Compound ist eine Sammlung von Parametern, denen bestimmte Werte zugewiesen werden (Wertelieferant). Innerhalb eines MEDICI-PK-Projektes können beliebig viele Compounds angelegt werden; jedes kann einen anderen Wirkstoff repräsentieren, indem es andere Werte für die Parameter enthält. Innerhalb eines Simulationsobjektes (siehe (12.2)) können nur vollständige Compounds verwendet werden. Daher gibt es die Möglichkeit, einen neuen Compound auf Basis eines bereits definierten FBTs anzulegen: es werden alle benötigten (d.h. alle in den Modellen referenzierten), als wirkstoffabhängig gekennzeichneten Parameter in das neue Compound eingetragen, es müssen dann nur noch die Werte zugefügt werden.

12.1.10 Individuum/Individual

Ein Individual ist eine Sammlung von Parametern, denen bestimmte Werte zugewiesen werden (Wertelieferant). Innerhalb eines Projektes können beliebig viele Individuals angelegt werden; jedes kann ein anderes Individuum repräsentieren, indem es andere Werte für die Parameter enthält. Innerhalb eines Simulationsobjektes (siehe (12.2)) können nur vollständige Individuals verwendet werden. Daher gibt es die Möglichkeit, ein neues Individual auf Basis eines bereits definierten FBTs anzulegen: es werden alle benötigten (d.h. alle in den Modellen referenzierten), als individuumabhängig gekennzeichneten Parameter in das neue Individual eingetragen, es müssen dann nur noch die Werte zugefügt werden.

12.1.11 Mix parameter set und General parameter set

Diese werden angelegt wie Compound und Individual; es werden die Parameter gesammelt, die als mix parameter oder als general parameter klassifiziert wurden.

12.2 Simulationsobjekt

12.2.1 Definition

Das Simulationsobjekt ist die Zusammenführung von allen bisher definierten orthogonalen Informationen mit dem Ziel, ein rechenbares Differentialgleichungssystem aufzustellen! Anhand der Abbildung 23 wird deutlich, in welcher Form dies geschieht.

The screenshot shows the 'Simulation' dialog box with the following components:

- Selection Table:**

Name	Selection
Maki Keizer 1995 model in pancreas interstitial	Maki Keizer 1995 model in pancreas interstitial
Full Body template	Maki Keizer 1995 model (copy)
General parameter set	insulin glucose model (copy) 1
Individual selection	healthy male human (copy)
Compound selection 1	tolbutamide_1 (copy)
Compound selection 2	insulin (copy)
Compound selection 3	
- Dependencies C-I:**

Dependencies C-I	Selection
	Tolbutamide-rat 2
- Requirements by Volumes and Flow:**

No.	m...	Name	Value	Unit	Individual	Unit	Compound	Unit
1		Vmax1	1.200e+02	mM/min			-	
2		K1	1.400e+00	mM			-	
3		Vmax2	3.200e+01	mM/min			-	
4		K2	1.700e+01	mM			-	
5		K _j	4.000e-05	mM			-	
6		V _m	2.400e-01	mM/min			-	
7		V _s	3.400e-02	mM/min			-	
8		K _s	1.300e-01	mM/min			-	
9		L	1.000e-02	mM/min			-	
10		tau	2.000e+01	min			-	
11		K _m	9.800e+00	mM			-	
12		K _{inh}	1.000e-06	mM			-	

Abbildung 23: Dialog zur Definition eines Simulationsobjektes: Zusammenführung von FBT (Modellierung) und Parameterinstanzen aus Individuum- und Wirkstoff-Datenbanken

Aus der Liste der angelegten FullBodyTemplates wird eines ausgewählt, dazu werden Parameterobjekte General parameter und Individual ausgewählt (Abbildung 23, Rechteck 1). Es können bis zu drei Wirkstoffe parallel gerechnet werden (Abbildung 23, Rechteck 2); „ein Wirkstoff wird gerechnet“ bedeutet, dass das DGL-System für diesen Wirkstoff aufgestellt wird. Dies wird erkannt, sobald ein entsprechender Compound, also das Parameterobjekt, welches einen Wirkstoff repräsentiert, ausgewählt wird. Der Startwert einer Wirkstoffkonzentration wird i.A. über die Dosierung gegeben. Eine Dosierung kann zu jedem der zu rechnenden Wirkstoffe angegeben werden (Abbildung 23, Rechteck 2). Mixparameter müssen ausgewählt werden, falls diese erforderlich sind. Erforderlich bedeutet, dass in mindestens einem der Modelle des FullBodyTemplates ein Parameter benutzt wird, der als Mischparameter klassifiziert wurde.

Anhand des FullBodyTemplates können alle Parameter, die benutzt werden, aufgeführt werden (Abbildung 23, Rechteck 3). Sie werden nach ihrer Abhängigkeit zusammengefasst. Nach Auswahl der Parameterobjekte (Compound, Individual, General, Mix) können die aktuellen Werte dieser Parameter angezeigt werden. Es ist in diesem Programmpaket möglich, die Werte dieser Parameter hier noch zu ändern!

12.2.2 Weitere Ansichten zur Verifizierung und Änderung

Der Zugriff auf Modellbeschreibungen und Parameterwerte mit dem Ziel der Verifizierung und Änderung ist aus verschiedenen Ansichten möglich, siehe die folgenden Abbildungen 24 und 25. In der Baumdarstellung (Abbildung 24) werden die Informationen kompartimentweise dargestellt. Über die Subkompartimente sind die dort verwendeten Modelle zu den Prozessen Transfer, Bindung, Metabolismus, Dosierung aufgeführt; deren Beschreibung und auch die Parameterwerte können dort geändert werden.

In der Modellliste (Abbildung 25) werden alle Modelle, sortierbar nach Kompartimenten und Subkompartimenten, nach Typ oder nach Namen, aufgeführt und sind leicht wiederzufinden. Die zugehörigen Parameter werden getrennt aufgeführt. Auch sie sind für Änderungen zugänglich.

Alle Informationen sind daher immer im Zusammenhang ihrer Umgebung und ihrer Bedeutung sichtbar und erkennbar! Die gesamte Darstellung ist transparent.

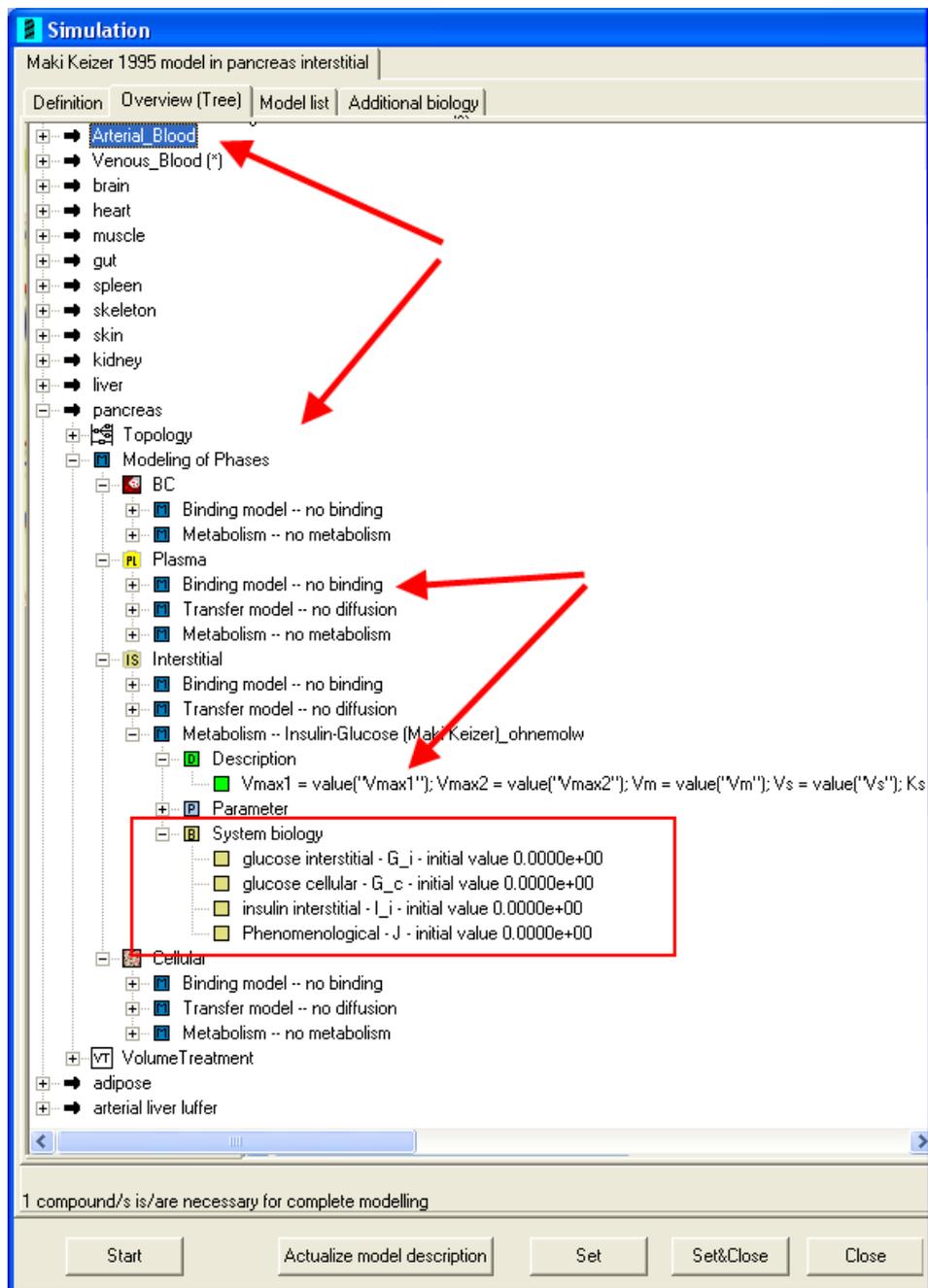


Abbildung 24: Informationen in Baumdarstellung, ausgehend von Organtopologie; Modelle und Parameter sind im Zusammenhang sichtbar

The screenshot displays the 'Simulation' window for the model 'exp. -tolbutamide (refined+ion)'. The interface is divided into several sections:

- Model List:** A table listing 35 models (No. 22-56) across various organs (brain, gut, heart, kidney) and phases (Plasma, Interstitial, Cellular, BC). Each model is associated with a specific model type (Binding, Transfer, Metabolism) and a description. A red box highlights models 22 through 32.
- Parameter Table:** Located on the right, it lists parameters such as 'fast_rate(tolbutamide_1)', 'V(tolbutamide_1)', 'noBindingSites_album...', 'albumin (plasma)(tolbu...', and 'Kd_albumin(tolbutami...'. Each parameter has a corresponding numerical value and unit.
- Add. Compounds:** A section below the parameter table with a table for 'Compound' and 'Value', currently showing 'bound concentration ...' with a value of '0.0000e+00'.
- Mapping:** A section at the bottom right with a table for 'No.' and 'Compound', currently empty.

Abbildung 25: Die Modelliste gibt einen Überblick über alle verwendeten Modelle in Vielfachheit pro Subkompartiment. Die Modelle können hier deaktiviert werden, d.h. sie werden nicht ausgewertet und tragen nicht zum DGL-System bei. Der Zugriff auf die Parameter jeden Modells in Abhängigkeit vom Organ ist möglich.

12.2.3 Mapping der Modelle

Sind nur 1-Compound-Modelle innerhalb des FullBodyTemplates benutzt, so ist die Zuordnung von einer Wirkstoffkonzentration zu dem Bezeichner „Comp1“, der im Modell die Wirkstoffkonzentration bezeichnet, trivial. Werden aber n-Compound-Modelle benutzt, so bedeutet das zum einen, dass im Simulationsobjekt n Wirkstoffe angegeben werden müssen, und zum anderen, dass die Zuordnung von Wirkstoffen zu den Bezeichnern Comp1, Comp2,.. nicht klar ist. Dies muss über ein Mapping erfolgen, sobald die Wirkstoffe ausgewählt sind (siehe Abbildung 26).

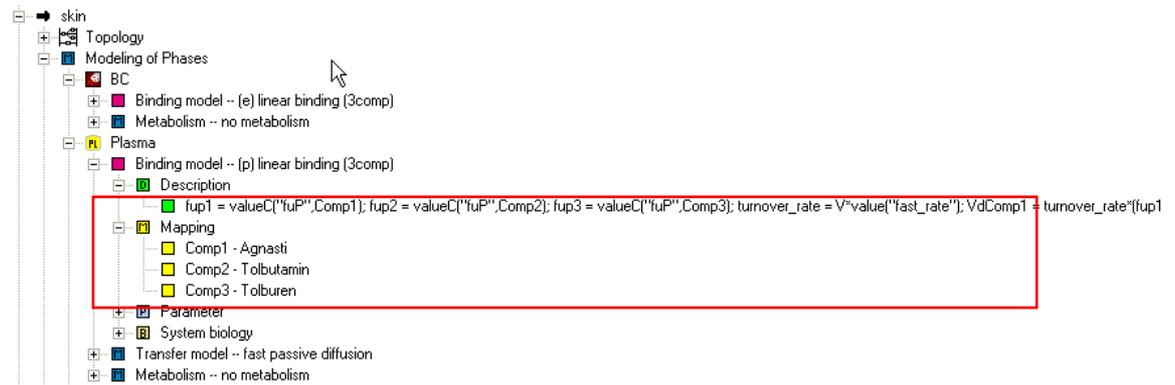


Abbildung 26: Mapping zwischen Bezeichner und tatsächlich benutzten Wirkstoffen

12.3 Auswertung und Ergebnisse

Das Simulationsobjekt hat alle Informationen und kann gemäß des Kalküls das DGL-System für alle zu bestimmenden Konzentrationsverläufe von Wirkstoffen aufstellen. Zusätzliche Angaben wie Endzeit oder Einstellungen die Numerik betreffend werden innerhalb des Simulationsobjektes gesichert.

Das DGL-System wird mit einem semi-impliziten Euler-Verfahren gelöst (die Beschreibung dazu findet sich im Abschnitt (12.7) sowie im Anhang). Es ergeben sich die gesuchten Konzentrationsverläufe; sie werden direkt online (Abbildung 27) angezeigt.

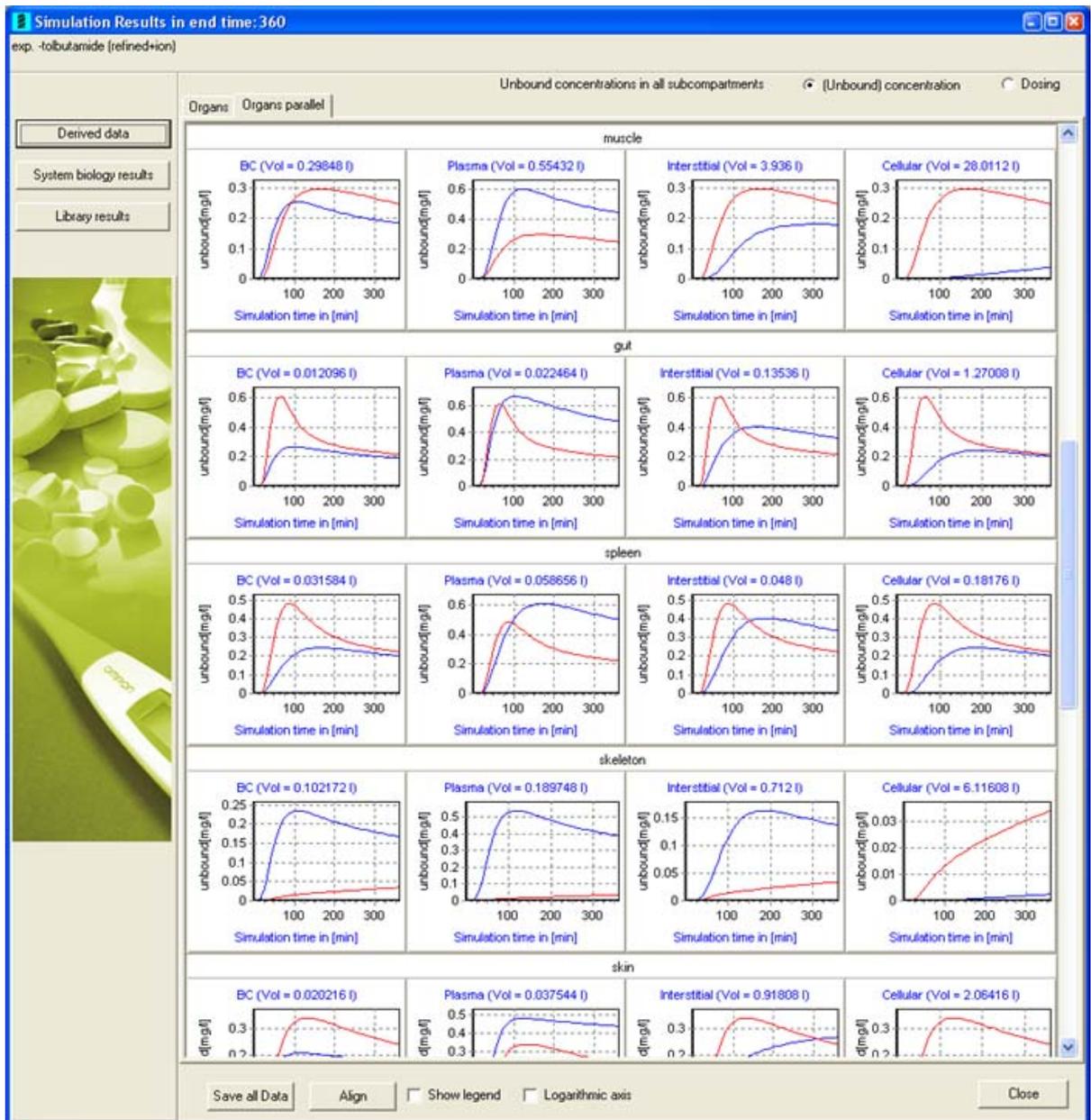


Abbildung 27: Simulationsergebnisse mit zugeladenen Referenzkurven pro Organ, pro Subkompartiment pro Wirkstoff sind direkt nach der Simulation verfügbar

In MEDICI-PK werden bestimmte Größen automatisch (fest implementiert) mitgerechnet und zur Verfügung gestellt. Dazu gehören der AUC (area under the curve) und der max-Wert (größter erreichter Konzentrationswert). Es wird eine Massenbilanz mitgeführt, um den Verbleib einer Wirkstoffgabe verfolgen zu können.

Darüberhinaus können eigene benutzerdefinierte Ausgabegrößen angelegt werden, die ebenso ausgewertet werden. Auch deren Verläufe (Abbildung 28) werden online angezeigt.

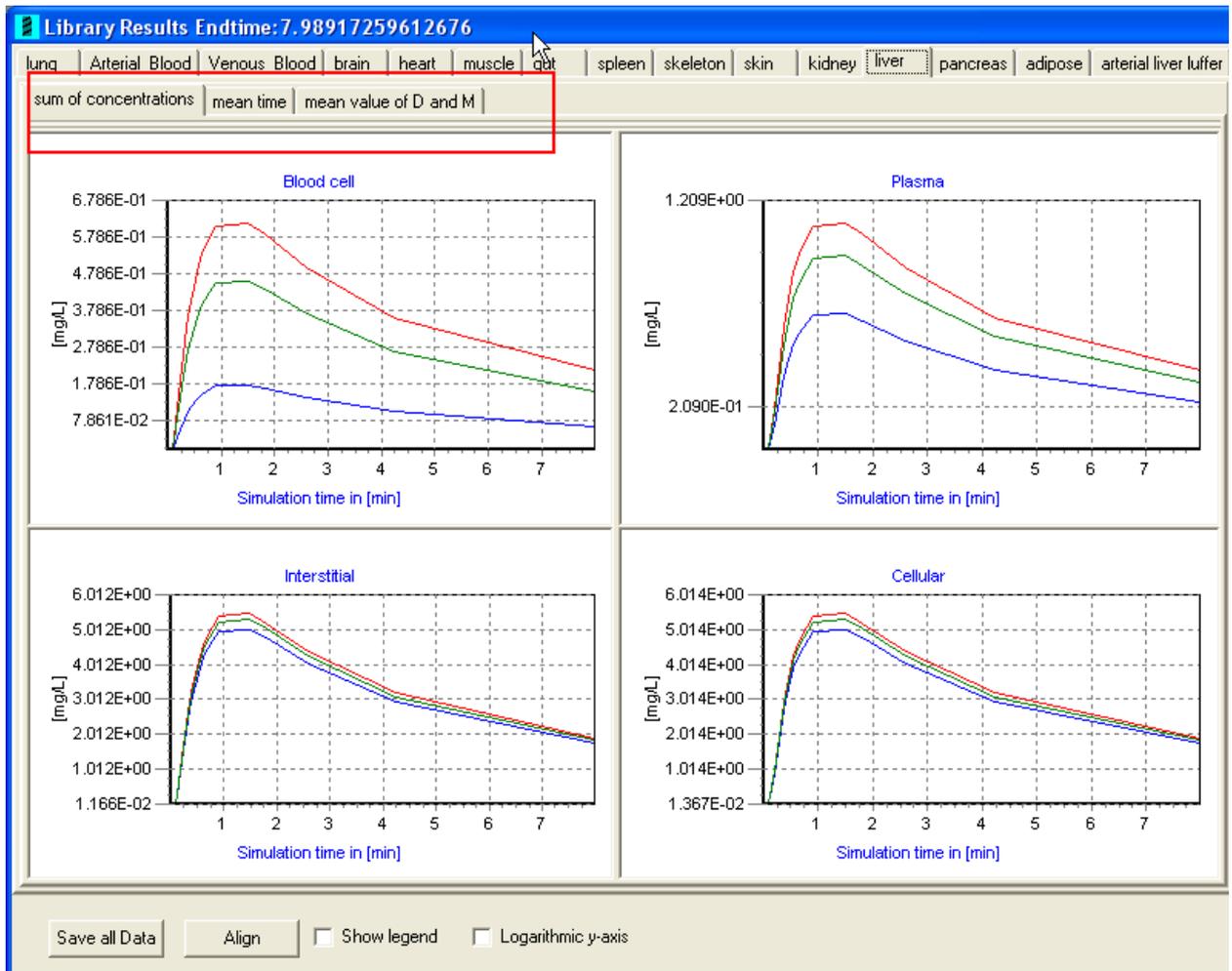


Abbildung 28: Benutzerdefinierte Ausgaben, abhängig von beliebigen Größen des Systems (sämtliche Konzentrationswerte + Parameter), können in beliebiger Anzahl definiert werden. Sie können pro Organ, pro Subkompartiment, pro Wirkstoff ausgewertet werden

Alle Resultate sind online verfügbar und exportierbar.

12.4 Intelligenz in der Software

Die Elementaren Prozesse bzw. die Modelle, die diese beschreiben, müssen vom Modellierer frei vorgegeben werden können. In früherer Zeit gab es Programme, die eine Vorkompilation vornahmen, um benutzerdefinierten Code einem Programm zuzulinken. Dies ist nicht mehr state-of-the-art. Die Modelle müssen zur Laufzeit des Programms ausgewertet werden.

12.4.1 Auswertung von Skripten, Übergabeparameter

In Presto-Kinetics ist es möglich, Reaktionsgeschwindigkeiten als Funktion anzugeben, die u.a. von Parametern abhängen, die im gleichen Projekt definiert sind, die aber durchaus darüberhinaus noch zeitlich oder temperaturabhängig variabel sind. Dafür wurde ein Compile-Mechanismus („Interpretermechanismus“) implementiert, der eine solche Funktion in jedem Zeitschritt der Simulation auswertet, d.h. er findet die genannten Größen im Projekt, ermittelt deren aktuellen Wert und führt bestimmte arithmetische Operationen aus. Für die Elementarprozesse, die in unserem Zusammenhang zu definieren sind, bietet sich dieser Mechanismus genauso an. Er erlaubt die nahezu beliebige Formulierung von Kinetiken; es sind natürlich Schnittstellen zu definieren, so dass die Eingabeparameter eines solchen Skriptes sowie der/die Rückgabewert(e) abgesprochen implementiert werden.

Grundlegend ist die Festlegung, wieviele Wirkstoffe in einer Modellbeschreibung behandelt werden sollen (1-Compound-Modell oder n-Compound-Modell). Immer verfügbar sind in einem Modell die aktuellen Konzentrationen eines Wirkstoffs (bzw. der Wirkstoffe) in einem Subkompartiment sowie das Volumen des Subkompartiments, in welchem die Auswertung aktuell stattfindet. Sie sind über die festen Bezeichnungen Comp1 und V zugreifbar. In n-Wirkstoffmodellen werden die Konzentrationen der weiteren Wirkstoffe über Comp2, Comp3 übergeben.

Als Rückgabewerte werden nahezu immer Ströme erwartet (kenntlichgemacht durch den Rückgabewert *VdComp*; dabei handelt es sich wählbar (!) um einen Masse- oder Molenstrom.

Im folgenden werden die implementierten Modelltypen beschrieben. In den Beispielen wird eine Syntax verwendet, die etwas anschaulicher ist als die tatsächliche Syntax des Interpreters (dazu siehe [48]). Für den elementaren Prozess Flow ist ein festes Modell implementiert, das keine Benutzeränderungen zulässt und daher auch nicht aufgeführt wird. Dosierungen sind dagegen als abstrakte Beschreibungen zugelassen und werden auch hier aufgeführt.

Transfermodelle Als weitere Übergabeparameter werden die Konzentrationen der Wirkstoffe des darüberliegenden Subkompartiments übergeben, die über feste Bezeichner Comp1Top, Comp2Top, ect. angesprochen werden. Rückgabewert ist der

Strom für jeden Wirkstoff, der durch den Transfer erzeugt wird. Der Transfer ist gerichtet; ein positives Vorzeichen bedeutet Transfer aus einem Subkompartiment in ein darüberliegendes Subkompartiment. Beispiel:

$$VdComp1 = value(„D_fast“) \cdot (Comp1Top - Comp1)$$

Metabolismen Es werden keine weiteren Übergabeparameter übergeben. Für jeden Wirkstoff muss die Masseänderung (Molenänderung) pro Zeit angegeben werden, wie gewohnt als Funktion anderer Größen. Beispiel:

$$\begin{aligned} VdComp1 &= -value(„myHepa“) \cdot Comp1 \\ VdComp2 &= value(„d_org“) \cdot Comp1 - value(„P“) \cdot Comp2 \end{aligned}$$

Bindungsmodelle Es werden keine weiteren Übergabeparameter übergeben. Rückgabewert ist die Masseänderung (Molenänderung) pro Zeit, die durch die Bindung erzeugt wird.

Dosierungsmodelle Übergabeparameter sind die Initialdosierungsmenge und die aktuelle Simulationszeit. Rückgabewert ist der aktuelle Dosierungswert. Beispiel: dosing = value(„h“) * initdosing

12.5 Annahmen und Möglichkeiten in Medici-PK

12.5.1 Annahmen

1. Die Volumina der Kompartimente bleiben während einer Rechnung konstant, es werden keine Volumeneffekte berücksichtigt
2. Ein Flow wird nur für die Subkompartimente BC und Plasma berechnet.
3. Der Flow zwischen zwei Subkompartimenten (verschiedener Kompartimente) wird automatisch gerechnet und kann vom Benutzer nicht eigenständig modelliert werden. Der Blutfluss in ein Subkompartiment hinein stimmt mit dem Blutfluss aus dem Subkompartiment heraus überein ($v_e = v_f$).
4. Ein Subkompartiment wird als gut-durchmischt angesehen, d.h. ein Wirkstoff, der einem Subkompartiment zugeführt wird, wird sich direkt gleichmäßig verteilen
5. Es können bis zu 3 Wirkstoffe parallel gerechnet werden. Die Liste der möglichen Wirkstoffe ist zwar grundsätzlich offen, es mussten aber Einschränkungen gemacht werden, um z.Bsp. Übergabewerte zu realisieren (siehe 12.4.1)

12.5.2 Möglichkeiten

Für den Benutzer sichtbar, erlaubt dieses Softwarepaket:

1. Eingabe beliebiger Organe, damit auch verschiedene Species modelliert werden können und auch die Zusammenfassung mehrerer Organe zu Clustern (Beispiel: Magen-Darm-Trakt, oder gut-durchblutete Organe) möglich ist
2. Eingabe einer beliebigen Topologie/Physiologie, damit sowohl feine Strukturen mit der Angabe aller zugänglichen Organe möglich sind, als auch Teilstrukturen (durch Auswahl nur weniger Organe) untersucht werden können, je nach Absicht und Datenlage, aber auch Studien über beliebig geartete Physiologien durchgeführt werden können
3. Verwaltung von Parametern
4. Verwaltung von Modellierungsfunktionen für verschiedene Zwecke: Dosierungen, Bindungsmodelle, Transfermodelle sowie Metabolismusmodelle.
5. Anzeige sämtlicher Konzentrationsprofile mit Möglichkeiten zur Weiterverarbeitung
6. Anzeige benutzerdefinierter Variablen, die in Abhängigkeit von allen internen Größen ausgewertet werden (Bibliotheksvariable)

12.5.3 Annahmen, Möglichkeiten und Einschränkungen in der Topologie und Kompartimentierung

1. Zugelassen sind beliebige und beliebig viele Kompartimente/Organe
2. Arterie und Vene werden standardmäßig immer angenommen, also in eine Topologie mit übernommen, können aber eliminiert werden
3. Kompartimente des Typs Arterie und Vene haben nur die Subkompartimente Blutzellen und Blutplasma
4. Flow findet nur in den Blutkompartimenten statt
5. Zugänglich sind bisher nur die Subkompartimente Blutzellen, Blutplasma, Interstitium und Gewebe (cellular), da diese für eine komplexe Modellierung bisher ausreichen
6. Die Subkompartimente sind volumenmäßig abhängig vom Gesamtvolumen des Kompartiments. Ebenso wie der Blutfluss werden die Volumina in Abhängigkeit von den Kompartimentwerten durch Parameter wie Hct bestimmt, siehe 9.2.6.

12.6 Sicherheitsaspekte

Modellierung in der Pharmakologie unterliegt besonderen Sicherheitsrichtlinien. Es ist unbedingt nötig, reproduzierbare Ergebnisse zu erhalten. So sehr die Informationstechnik auch Möglichkeiten der Datenaufbereitung schafft, so schwierig ist es dennoch, (wie in diesem Fall) Daten und Modelle zu verwalten. Eine der Fragen, die beim Entwurf von Software immer auftritt, ist die, ob Objekte referenziert oder kopiert werden. Referenz bedeutet automatische Adaptivität bei Änderungen, zugleich aber auch immer die Unsicherheit, welche Werte tatsächlich verwendet wurden.

In MEDICI-PK werden Objekte, die innerhalb anderer Objekte benutzt werden, kopiert. Änderungen, die innerhalb eines Simulationsobjektes an Modellen und/oder Parametern gemacht werden, kommen ausserhalb des Simulationsobjektes also nicht zum Tragen, können aber nachgeführt werden, wenn sich eine Änderung als sinnvoll erwiesen hat.

Ein wesentlicher Automatismus sorgt dafür, dass jede Simulation, die gestartet wird, als ein Simulationsobjekt inklusive der Ergebnisse in eine Liste gehängt wird und solange, zum Beispiel als Referenzlösung, zur Verfügung steht, wie das Programm läuft. Es ist auch möglich, diese Simulationsobjekte in eigene Dateien zu speichern. Sie enthalten immer vollständig den Stand bzgl. Modellen und Parametern, mit denen die Simulation gestartet wurde.

12.7 Numerische Lösung

12.7.1 Problemstellung

Die Modellierung in Medici-PK führt auf ein Anfangswertproblem der Form

$$x'(t) = f(x, t), x(0) = x_0 \in \mathbb{R}^n, f : \mathbb{R}^n \rightarrow \mathbb{R}^n, t \in [0, T] \quad (73)$$

Die Dimension n ergibt sich aus allen beteiligten Zustandsgrößen des Modells. Die rechte Seite f ergibt sich additiv aus verschiedenen Beiträgen $f_i = \sum_p f_i^p$, wie es sich durch Anwendung des Pharmakokinetischen Operators (69) auf den gesamten Organismus ergibt. In die rechte Seite $f(x, t)$ geht die Zeit direkt ein. Es handelt sich nicht um ein autonomes Problem, da wir Dosierungen betrachten, die nicht nur Startwerte x_0 der Zustände beschreiben, sondern sowohl über differenzierbare Funktionen, als auch unstetige Beiträge über die Zeit modelliert werden können (siehe (67)). Daher bietet es sich an, das Gesamtzeitintervall $[0, T]$ gemäß der Dosierungszeitpunkte zu zerlegen in ein Gitter

$$\Delta = \{t_0 = 0 < t_1 < \dots < t_d < t_{d+1} = T\} \quad (74)$$

wobei die $t_i, i = 1, \dots, d$ die möglichen Dosierzeitpunkte bezeichnen.

Wir nehmen nun an, dass auf jedem Teilintervall $[t_i, t_{i+1}], i = 0, \dots, d$ die gesamte rechte Seite so oft wie nötig stetig differenzierbar ist, so dass sowohl Existenz und Eindeutigkeit der Lösung wie auch die grundsätzliche numerische Lösung gesichert sind.

Wir gehen davon aus, dass die betrachteten Systeme *steif* in numerischen Sinne sind. Daher berechnen wir die numerische Lösung mit dem Differentialgleichungssolver EULSIM ([7]), welches für steife Differentialgleichungen für technische (10^{-3}) bis mittelhohe Genauigkeiten (10^{-6}) sehr bewährt ist. EULSIM basiert auf einer linear-impliziten Euler-Diskretisierung des AWP's ([8]). Für einen gegebenen Zustand x_0 und eine gegebene Schrittweite τ ergibt sich die Näherung x_1 dabei durch (autonome Formulierung)

$$(I - \tau A)\Delta x = \tau f(x_0), x_1 = x_0 + \Delta x \quad (75)$$

mit der Jacobimatrix $A = (a_{ij}) = \left(\frac{\partial f_i}{\partial x_j}\right)$.

Im Rahmen von EULSIM wird ein solches System (75) für verschiedene Schrittweiten $\frac{\tau}{k}$, k gemäß einer Folge, berechnet und dann extrapoliert. Details finden sich in den genannten Arbeiten. Eine Ordnungs- und Schrittweitensteuerung sorgt dafür, dass das Problem (73) mit gegebener Genauigkeit und möglichst geringem Rechenaufwand gelöst wird. Auf Basis von Fehlerschätzungen wird eine Schrittweite τ_{new} für den nächsten Schritt bereitgestellt, die wir im Sinne des Dosierungsgitters (74) gegebenenfalls einschränken ($\tau_{new} = \min(\tau_{new}, t_{next}), t_{next}$ der als nächstes zu erreichende Dosierungspunkt). In MEDICI-PK wird EULSIM in einer C++-Implementierung basierend auf der TIDE-Bibliothek ([19]) verwendet.

12.7.2 Jacobi-Matrix

Aus (75) ist ersichtlich, dass die Berechnung der Jacobimatrix der rechten Seite $f(x, t)$ notwendig ist. Da große Teile von f sich erst durch Spezialmodellierung ergeben, kann nicht die gesamte rechte Seite vorab analytisch abgeleitet werden. Ebenso läßt sich hier aufgrund der modularen Verarbeitung, der Offenheit der Funktionen und der möglichen späteren Anbindung von verkapselten Bibliotheken keine automatische Differentiation ([17]) verwenden.

Man könnte nun einen numerischen Differenzquotienten der gesamten rechten Seiten berechnen:

$$\frac{\partial f_i}{\partial x_j} \approx \frac{f_i(x + \Delta x_j e_j) - f_i(x)}{\Delta x_j},$$

wobei die Störung Δx_j wie folgt gegeben ist (ρ ein Sicherheitsfaktor $\succeq 1$, eps_{mach} Maschinengenauigkeit seien gegeben) :

$$\Delta x_j = \begin{cases} \rho * eps_{mach} |x_j|, & |x_j| > 0 \\ \rho * eps_{mach}, & |x_j| = 0 \end{cases}$$

Dieses Vorgehen ist sehr aufwändig und nimmt keinerlei Rücksicht auf die besondere Struktur der hier behandelten Systeme. In unserem Fall lohnt es sich, anders vorzugehen: für jeden Teilausdruck f_i^p von $f_i = \sum_p f_i^p$ wird angenommen, dass er sich als Verkettung schreiben lässt (p durchläuft die physiologischen Effekte Flow, Transfer, Bindung, Metabolismus):

$$f_i^p(x, t) = g_i^p(m_i^p(x, t))$$

Die Funktion g_i^p beschreibt dabei den strukturell vorab bekannten Teil von f_i^p , während $m_i^p(x, t)$ sich aus den speziellen Modellen ergibt. In Kapitel 10.1 ist jeder Prozessstyp p ausführlich beschrieben; die $m_i(x, t)$ entsprechen den dort angegebenen

Funktionen $f_{transfer}$, $f_{binding}$, $f_{metabol}$. Für die Ableitung betrachten wir dann

$$\frac{\partial f_i^p(x, t)}{\partial x_j} = \frac{\partial g_i^p}{\partial m_i^p} \frac{\partial m_i^p(x, t)}{\partial x_j}$$

Die Ableitung $\frac{\partial g_i^p}{\partial m_i^p}$ ist dann geschlossen bekannt, für $\frac{\partial m_i^p(x, t)}{\partial x_j}$ wird ein numerischer Differenzenquotient benutzt. Nur an den Stellen i , an denen $m_i^p(x, t)$ auch tatsächlich ausgewertet wird, muss sowohl der Differenzenquotient gebildet als auch die Ableitung bestimmt werden. Dies bedeutet, dass nur die wirklich benutzten Terme für die Ableitungsberechnung herangezogen werden, und dass diese Terme auch nur bezüglich der dort tatsächlich verwendeten Größen (numerisch) abgeleitet werden. Damit ist größtmögliche Effizienz und Genauigkeit gesichert. Wie sich die Funktionen g_i^p und m_i^p in den betrachteten Spezialfällen ergeben, ist im Anhang Kapitel 17 gezeigt.