

Teil II

Modellierung in der Pharmakokinetik

8 Modularität und Orthogonalität als Konstruktionsprinzip der computergestützten Modellierung

Bei der mathematischen Modellierung chemischer und biologischer Prozesse spielen kinetische Betrachtungen zu den Konzentrationen von Reaktanten/Substanzen eine wichtige Rolle. Das haben wir bereits im ersten Teil dieser Arbeit an Beispielen der Technischen Chemie gesehen, in der Pharmakokinetik gilt dies ebenso. Daher werden wir im folgenden im Zusammenhang mit dem Begriff *Modellierung* die Beschreibung der Kinetik verstehen und dementsprechend immer auch die Bildung von Differentialgleichungen zur Beschreibung der Konzentrationsveränderung von Reaktanten als Hintergrund beachten.

Ein Softwarepaket kann *diesen* Aspekt der Modellierung unterstützen. Was muss es dazu können, und wie muss es dafür vorbereitet werden?

Um dies beantworten zu können, muss man sich zunächst den Gesamtaufwand der Modellierung klarmachen:

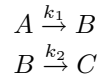
Modellieren besteht nicht (alleine) darin, ein Differentialgleichungssystem aufzustellen, welches Messdaten eines Systems möglichst genau nachbildet. Dieses ist ganz am Ende einer langen (Modellier-)Entwicklungskette ein Kriterium, um die Güte eines Modells abschätzen zu können. Innerhalb dieser Entwicklungskette gibt es ein Austesten, Austausch, Diskutieren und Verifizieren von verschiedenen Ansätzen für die Elementaren Prozesse, bis man zu dem Modell gelangt, welches den Erwartungen entspricht.

Läge die Aufgabe nur darin, ein DGL-System aufzustellen und dann zu lösen, könnte man dies mit Hilfe sogenannter Gleichungsbasierter Tools wie z.B. MATLAB erledigen. Wer solche Tools kennt, kennt aber auch die Krux dieser Instrumente, die verhindert, dass solche Tools zur Modellentwicklung eingesetzt werden können:

ein großes DGL-System ist nicht rückwärts-abbildbar, d.h. man kann zwar ein komplexes Modell eines Systems in DGLs abbilden, aber man kann aus den DGLs nicht eindeutig die elementaren Prozesse und Zusammenhänge ablesen, aus denen es entstanden ist. Das Wissen um die Gesamtstruktur bleibt im Kopf des Modellierers. Die simple Denkaufgabe, eine strukturelle Information der Gesamtstruktur auszutauschen oder aber Werte von Parametern zu ändern wird schon für den Entwickler des DGL-Systems zu einer mühsamen und fehlerträchtigen Aufgabe, denn dies muss an allen Stellen des DGL-Systems nachgeführt werden, an denen diese Information genutzt wird. Dazu kann er auf sein Wissen zurückgreifen. Zu einer praktisch unmöglichen Aufgabe wird dies für jemanden, der nur das DGL-System erhält. Er kann nicht

entscheiden, wo die strukturelle Information in derselben Bedeutung genutzt wurde.
Wissenstransfer findet über ein reines DGL-System nicht statt.

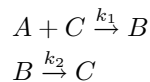
Ein simples Beispiel (wir vernachlässigen hier mögliche weitere Reaktionen, die das System zu einem tatsächlich anwendungstechnisch sinnvollen System machen):



entspricht (nach Potentialansatz) dem DGL-System

$$\begin{aligned} \frac{dA}{dt} &= -k_1 A \\ \frac{dB}{dt} &= +k_1 A - k_2 B \\ \frac{dC}{dt} &= k_2 B \end{aligned}$$

Eine Änderung dieses Teil-Modells durch Austausch der Teilstruktur Elementarprozess 1 gegen eine andere betrifft nur diesen Elementarprozess und lässt sich leicht hinschreiben:



Die entsprechende Änderung im DGL-System betrifft aber das DGL-System als Ganzes, ist nicht sofort einsichtig und nicht transparent.

$$\begin{aligned} \frac{dA}{dt} &= -k_1 AC \\ \frac{dB}{dt} &= +k_1 AC - k_2 B \\ \frac{dC}{dt} &= -k_1 AC + k_2 B \end{aligned}$$

Händisch durchgeführte Änderungen in einem großen und komplexen DGL-System sind immer problematisch. Beachtet man jetzt, dass Modellierung kein singuläres Ereignis ist, sondern dass neben dem initialen Aufstellen eines (Start-)Modells (mit Bildung eines DGL-Systems) Variationen der Strukturen und auch der Eingangsparameter notwendig sind, so ist klar, dass effektive und im Ergebnis langlebige Modellentwicklung nicht mit Hilfe reiner Gleichungsbasierter Tools betrieben werden kann.

Wie kann Software den Prozess des Modellierens unterstützen? Wie man an dem obigen Beispiel schön sieht, ist es unabdingbar, die Strukturen des Systems, welches man betrachtet, zu definieren, Hierarchien zu beschreiben und in Hinblick auf

die möglichen Änderungsanforderungen und Fragestellungen, die die Modellierungskette beantworten soll, zu optimieren. Diese Strukturen sind abhängig vom Anwendungsgebiet, mit dem man sich beschäftigt. Daher ist nicht jede Software für jedes Anwendungsgebiet optimal geeignet. Hat man die grundlegenden Strukturen identifiziert, kann ein Softwarepaket die Verwaltung, Verschaltung, Referenzierung aller Bausteine in Vielfachheiten oder Einfachheiten übernehmen und *zuverlässig* bearbeiten. Das Vorhandensein eines Kalküls, nach dessen Regeln ein Softwarepaket in die Lage versetzt wird, aus allen Strukturen das resultierende Differentialgleichungssystem zu erstellen, führt dann dazu, dass, wie es auch sein sollte, das Aufstellen und Lösen der Differentialgleichungen zu einer Nebenaufgabe wird. Das Wissen um die Strukturen kann auf diese Weise **transparent** gemacht und **transportiert** werden!

Eine wesentliche Anforderung an die gefundenen Strukturen ist die **Orthogonalität** zueinander. In erster augenscheinlicher Konsequenz bedeutet das, dass die Informationen, die die Strukturen tragen, eindeutig vorhanden sind. Orthogonalität geht aber noch weiter: der Austausch einer Teilstruktur muss unabhängig von den weiteren Teilstrukturen möglich sein; wird eine Struktur ausgetauscht, so müssen alle anderen davon unberührt verbleiben! Nur die Unberührtheit der anderen Strukturen sichert die **Zuordnung von Erkenntnisgewinn** zu einer bestimmten Struktur.

Die Forderung nach Orthogonalität ist im Sinne nicht nur der Informationsverarbeitung im Grunde trivial. Neben der Vermeidung von Redundanz wird das strikte **Durchhalten von Bedeutungszuweisung** erleichtert. Damit ist gemeint, dass elementare Bausteine wie auch Parameter mit einer gewissen Bedeutung in das Gesamtmodell eingesetzt werden; diese Bedeutung sollten sie behalten, wo auch immer sie verwendet (oder referenziert) werden. Wenn sie dieses aber tun, so können sie einer bestimmten Struktur zugewiesen werden. Wir werden in den folgenden Kapiteln diese Strukturbestimmung für das Gebiet der Pharmakokinetik vorführen.

Was hat die Autorin erreicht? Das Definieren der Strukturen und die Einhaltung der Orthogonalität der Information bei der Festlegung der Strukturen ist keineswegs trivial. Für die Pharmakokinetik ist es der Autorin gelungen, tatsächlich modulare Bausteine (Strukturen) festzulegen und in Softwareobjekte zu implementieren, die eine vollständig orthogonale und hierarchische Informationshaltung gewährleistet. Wie in den weiteren Kapiteln dargestellt werden wird, handelt es sich dabei um

1. Subkompartimente (Verteilungsräume)
2. Kompartimente (Hierarchie von Subkompartimenten)
3. Topologien von Kompartimenten
4. Modellbasis für elementare Prozesse
5. FullBodyTemplates als Zuordnung von Elementarprozessen zu Subkompartimenten auf Basis einer Topologie
6. Parameter-Objekte, die nach Abhängigkeit getrennt sind (Wirkstoffobjekte, Individuenobjekte, Mischparameter-Objekte, Allgemeine Parameter)
7. Dosierungsstrategien (unabhängig von Wirkstoff)

8. Simulationsobjekte, die die Verknüpfung benötigter Bausteine beschreibt

Diese Strukturen und Hierarchien wurden in einem Softwarepaket mit Namen MEDICI-PK implementiert, welches bereits zur Modellierung eingesetzt werden konnte. MEDICI-PK ist in der Lage, anhand eines ebenfalls von der Autorin entwickelten **Kalküls** aus den Informationsbausteinen automatisch das Differentialgleichungssystem aufzustellen, welches nach beschriebener Modellierung die Konzentrationsverteilungen des/der Wirkstoffe im gewählten Organismus beschreibt. Dieses Kalkül basiert auf Vereinbarungen, die aufgrund physiologischer Gegebenheiten getroffen wurden (wie zum Beispiel die Behandlung einer Topologie als Kreislauf), aber auch Vereinbarungen, die den Zusammenbau des Differentialgleichungssystems betreffen (wie zum Beispiel das Berücksichtigen eines Blutflusses als Reaktantenaustausch zwischen zwei Organen). Dies werden wir in Ausführlichkeit beschreiben.

Einsicht und Informationsgewinn als Konsequenz Die definierten Strukturen werden innerhalb des Softwarepaketes benutzt, um die Verteilung von Wirkstoffen in einem (je nach Erkenntnisstand vollständig modellierten) Organismus zu beschreiben. Da sämtliche Strukturen innerhalb eines sogenannten MEDICI-Projektes wiedererkennbar sind, ist die Transparenz des Modells gegeben. Wird ein solches Projekt an andere Benutzer übergeben, so können diese leicht Informationen über die Gesamtstruktur und die elementaren Teilstrukturen daraus ziehen. Damit ist vollständiger Wissenstransfer gewährleistet.

Durch die Modularität der Strukturen ist es möglich, je nach Wissensstand und nach Erfordernissen **beliebig einfache** oder **beliebig komplexe** Oberstrukturen (Hierarchien) aufzubauen, die weiterhin transparent bleiben. Dadurch ist es möglich, verschiedene Ziele der Modellierung effektiv zu verfolgen: Nachbau eines gegebenen Modells (Literatur), aber im besonderen auch Modellierung nach vorne, d.h. Ausprobieren von Ideen, abhängig oder auch gewollt unabhängig vom Forschungsstand des Fachgebietes.

Durch die Orthogonalität ist es möglich, gezielt Teilstrukturen auszutauschen: z.Bsp. können bestimmte Elementarprozesse in ihren Ausformungen oder Parameter (eines gewissen, gewünschten Typs) in ihren Werten geändert werden. Ebenso können Topologien ausgetauscht werden (Wechsel der Species) unter Beibehaltung der Elementarprozessauswahl. Wichtig ist, dass die Bedeutung der ausgetauschten Strukturen dennoch dieselbe bleibt, damit eine Konsistenz des Gesamtmodells erhalten bleibt. All diese Aufgaben kann ein Softwarepaket, welches diese Strukturen kennt, mit wenigen Aktionen seitens des Benutzers **verlässlich** erledigen. Durch diese Möglichkeit der schnellen Definition und Variation eines Gesamtmodells ist Erkenntnisgewinn über Richtigkeit, Adaptivität und Übertragbarkeit des Modells in kurzer Zeit, also sehr effektiv, gesichert.

Notwendigkeit eines neuen Ansatzes Der aktuelle Stand an verfügbaren Softwaretools zur Modellierung und Simulation in der Pharmakokinetik ist in [37] zusammengefasst. Im Groben kann man die Werkzeuge unterteilen in wissenschaftliche Software wie Berkeley Madonna, Matlab, MLAB, Octave, bio-mathematische Modellersoftware wie ADAPT II, ModelMaker, NONMEM, WinNonlin, oder physiologisch basierte Software wie GastroPlus, Pathway Prism, PK-Sim, Physiolab. Wie in [37]

festgestellt, scheint es eine inverse Beziehung zwischen Benutzerfreundlichkeit und Flexibilität in der Modellierung in der zur Verfügung stehenden Software zu geben. Danach gibt es bisher keine Software, die sowohl freie Modellierung zulässt als auch gleichzeitig den nötigen Rahmen im Sinne des Anwendungsgebietes Pharmakokinetik zur Verfügung stellt, innerhalb dessen ein hohes Maß an Benutzerfreundlichkeit erreicht werden kann. Dies ist, so ist die Autorin überzeugt, mit MEDICI-PK nun erreicht.