

## V11 metabolische Netzwerke

- Metabolisches Netzwerk: Gesamtheit aller metabolischen Pfade einer Zelle / eines Organismus
- Methoden der Systembiologie ermöglichen die integrierte, simultane Betrachtung von kompletten metabolischen Netzwerken.
- **wichtige Fragen:**
  - (1) wie soll man die Gesamtheit der metabolischen Flüsse im Netzwerk beschreiben?
  - (2) wie können mathematische Techniken die gezielte Manipulation von Mikroorganismen im Hinblick auf die Produktion bestimmter Substanzen unterstützen?

Softwarewerkzeuge WS 20/21 – V11

In V11 behandeln wir zunächst kurz metabolische Netzwerke und die Flux Balance Methode. Danach folgt ein zweiter Teil zu dynamischen Simulationen und zu SBML.

## Ecocyc : Datenbank mit Reaktionen von *E. coli*

Datentyp	Anzahl (release 20.1)
Gene	4505
Genprodukte, die in Mini-Review behandelt werden	3884
Genprodukte mit exp. validierten GO-Termen	3350
Enzyme	1567
Metabolische Reaktionen	1913
Substanzen	2699
Transporter	282
Transportreaktionen	485
Transportierte Substrate	338
Transkriptionsfaktoren	204
Regulatorische Interaktionen	6399
Transkriptionsinitiation	3457
Transkription – Elongation	23
Regulation der Translation	212
Enzym-Modulierung	2675
andere	32

Keseler et al. *Nucl. Acids Res.* (2017) 45: D543-D550

Softwarewerkzeuge WS 20/21 – V11

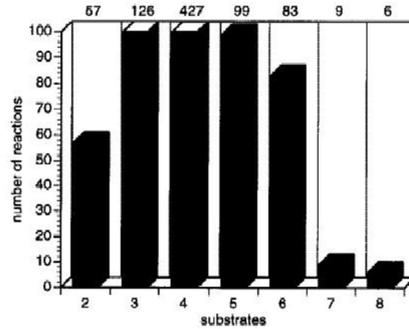
Das am besten charakterisierte metabolische Netzwerk existiert für *E. coli* in der Datenbank Ecocyc (<http://ecocyc.org>). Sie enthält außerdem regulatorische Informationen über Transkriptionsfaktoren und ihre Zielgene sowie zu Ko-faktoren der Enzyme.

## Beispiel: Stoffwechsel von *E. coli*

**Table 1.** Most Frequently Used Metabolites in *E. coli* Central Metabolism

Occurrence	Name of metabolite
205	H <sub>2</sub> O
152	ATP
101	ADP
100	phosphate
89	pyrophosphate
66	NAD
60	NADH
54	CO <sub>2</sub>
53	H <sup>+</sup>
49	AMP
48	NH <sub>3</sub>
48	NADP
45	NADPH
44	Coenzyme A
43	L-glutamate
41	pyruvate
29	acetyl-CoA
26	O <sub>2</sub>
24	2-oxoglutarate
23	S-adenosyl-L-methionine
18	S-adenosyl-homocysteine
16	L-aspartate
16	L-glutamine
15	H <sub>2</sub> O <sub>2</sub>

Im Mittel enthält jede Reaktion 4 Substrate.



**Figure 4** Diagram showing the number of reactions containing varying numbers of substrates (reactants plus products).

Ouzonis, Karp, Genome Research 10, 568 (2000)

Softwarewerkzeuge WS 20/21 – V11

Dies ist eine relativ alte Publikation über Ecocyc. Seither habe ich aber diese nützliche Analysen in keinem der Update-Papers über Ecocyc gefunden. (linke Tabelle): die „hub“-Metabolite, die an sehr vielen Reaktionen beteiligt sind, sind Wasser, ATP, ADP etc. Dies sind gewissermaßen „Kreuzungen“ der metabolischen Pfade.

(rechts) die meisten Reaktionen haben 4 Substrate:  $A + B \rightarrow C + D$

## Beispiel: Stoffwechsel von *E. coli*

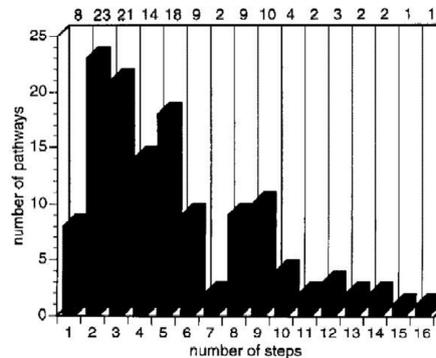
Eine ältere Version von EcoCyc enthielt 131 Stoffwechsel-Pfade.

Die Länge der Pfade variiert von 1 bis 16. Der Mittelwert ist 5.4.

Von den 607 Enzymen sind 100 multifunktional.

Purin-Nukleosid-Phosphorylase und Nukleosid-Diphosphatkinase katalysieren 7 bzw. 9 Reaktionen.

483 Reaktionen gehören zu einem Pfad, 99 Reaktionen gehören zu mehreren Pfaden.



**Figure 5** Length distribution of EcoCyc pathways; two pathways are not included because the number of steps is not known.

Ouzonis, Karp, Genome Research 10, 568 (2000)

Softwarewerkzeuge WS 20/21 – V11

Diese Graphik zeigt die Länge der Stoffwechselffade. Die meisten Pfade sind (bzw. waren 2000) nicht sehr lang.

Es gibt etliche Enzyme, die mehrere Reaktionen katalysieren. Allerdings hängt das auch davon ab, wie man zählt.

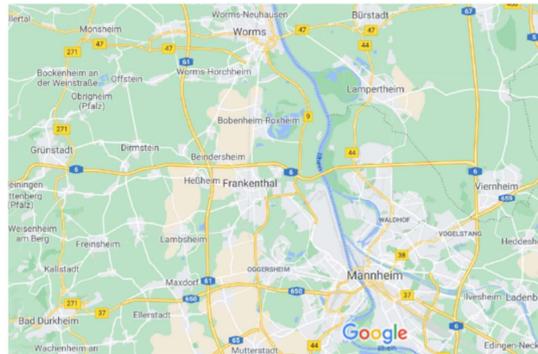
Nukleosid-Diphosphatkinase (NDPK) kann z.B. ADP, CDP, GDP, UDP und TDP phosphorylieren.

## Fazit

Stoffwechsel-Netzwerke von einfachen Organismen sind mittlerweile fast vollständig bekannt.

Ist die Beschreibung mit einzelnen Stoffwechsel-Wegen adäquat?

- Reaktionen, Enzyme und Substrate gehören oft zu mehreren Pfaden.
- Die Einteilung in einzelne Stoffwechsel-Pfade ist nicht immer eindeutig.



Softwarewerkzeuge WS 20/21 – V11

maps.google.de

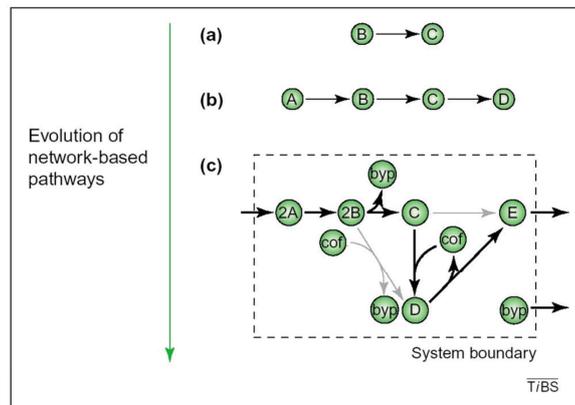
Man kann das metabolische Netzwerk eines Organismus z.B. mit dem Straßennetzwerk von Deutschland vergleichen. Es gibt manche Pfade wie Glykolyse, die man mit einer viel befahrenen Autobahn vergleichen könnte, z.B. die A8 von Stuttgart nach München oder die A1 durchs Ruhrgebiet. Manchmal ist es aber recht willkürlich, wo eine Autobahn beginnt und wo sie endet. Warum biegt die A6 bei Viernheim nach Süden ab und geht nicht nach Norden (die Richtung heißt A67)?

## Metabolische Pfade in der post-genomischen Ära

(a) klassische Biochemie bestimmt Stöchiometrien einzelner **Reaktionen**

(b) Katalogisierung vieler Reaktionen, Gruppierung nach gemeinsamen Metaboliten führt zu traditionellen **Pfaden** wie Glykolyse, Pentose-Phosphat-Pfad

(c) Durch komplette Information können nun die kompletten metabolischen Pfade zu einem **Netzwerk** kombiniert werden.



Im Laufe der Zeit entwickelte sich die Biochemie von einem Fach, das einzelne enzymatische Reaktionen aufgeklärt über die Katalogisierung in aufeinanderfolgende biochemische Pfade zum Fach der Systembiologie, bei dem man nun das gesamte Netzwerk aller biochemischen Pfade und enzymatischen Reaktionen gemeinsam analysiert.

## **Metabolische Pfade in der post-genomischen Ära**

Traditionelle metabolische Pfade dienen als konzeptioneller Rahmen für Forschung und Lehre.

Man kann dadurch Metabolismen verschiedener Organismen vergleichen.

Jedoch sind sie nicht für quantitative, systemische Bewertungen biologischer Reaktionsnetzwerke geeignet, da sie nur Teile der Netzwerke darstellen.

Sie wurden oft in Zelltypen entdeckt, in denen sie wichtige metabolische Funktionen übernehmen (z.G. Glykolyse in Hefe).

Man kann diese Pfade jedoch nicht einfach auf andere Zelltypen mit anderen Enzym- und Metabolit-Konzentrationen übertragen.

Kein Kommentar.

## Flux Balance Analyse

Jede chemische Reaktion erfüllt die **Erhaltung der Massen**.

Mit dieser grundlegenden Bedingung können metabolische Systeme untersucht werden.

Man muss lediglich die Stöchiometrie aller metabolischen Pfade und die metabolischen Anforderungen kennen. Für die zeitliche Änderung der Konzentration  $c$  jedes Metaboliten  $i$  gilt:

$$\frac{dc_i}{dt} = V_{\text{synthetisiert}} - V_{\text{abgebaut}} + V_{\text{transport\_in}} - V_{\text{transport\_out}}$$

Im Gleichgewicht kann man die Massenerhaltung in einem metabolischen Netzwerk mathematisch durch eine Matrixgleichung darstellen:

$$S \cdot v = 0$$

Hierbei ist die Matrix  $S$  wiederum die  $m \times n$  **stöchiometrische Matrix**,  
 $m$  = Anzahl der Metabolite,  $n$  = Anzahl der Reaktionen im Netzwerk.

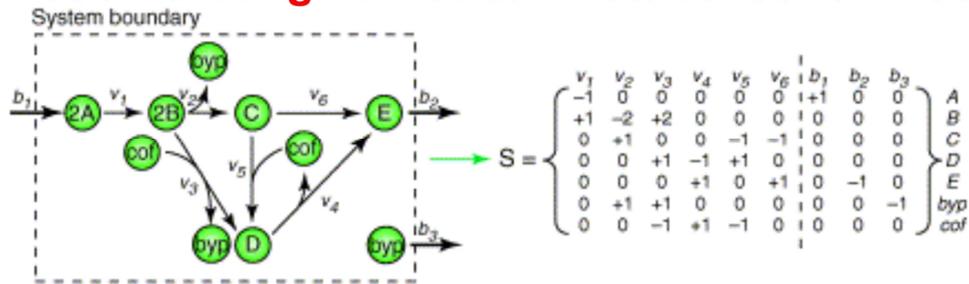
Der Vektor  $v$  beinhaltet alle Flüsse des metabolischen Netzwerks,  
nämlich die internen Flüsse, die Transportflüsse, und das Zellwachstum.

Softwarewerkzeuge WS 20/21 – V11

Wir steuern nun auf die wichtige Methode namens Flux Balance Analysis zu. Zunächst einmal erinnern wir an das wichtige Prinzip der Massenerhaltung bei chemischen Reaktionen. Substanzen verschwinden nicht oder kommen nicht aus dem Nirgendwo.

Die obere Gleichung gilt für die Konzentration eines Metaboliten innerhalb des gerade betrachteten Systems. Dieser kann zum einen aus anderen Substanzen hergestellt werden (1. Term rechte Seite) oder in andere Substanzen umgewandelt werden (2. Term), oder aus dem System hinaus transportiert werden (3. Term) oder hinein transportiert werden (4. Term). Wenn man alle Flüsse des Systems in einen großen **Flussvektor  $v$**  packt und die Beteiligung aller Metabolite an den Reaktionen des Systems in eine **stöchiometrische Matrix  $S$** , dann beschreibt die untere Gleichung die allgemeine Massenerhaltung. Wir schauen uns auf der nächsten Folie dazu gleich ein Beispiel an.

## Beschreibung vernetzter metabolischer Pfade



(a) aus genomischen, biochemischen, physiologischen Daten wird ein Reaktionsnetzwerk aufgestellt. Es gibt **interne Flüsse** innerhalb der Systemgrenzen und **externe Flüsse** zum Austausch mit der Umgebung.

(b) Dieses Netzwerk wird durch eine  $m \times n$  **stöchiometrische Matrix** dargestellt, in der  $m$  Metabolite durch  $n$  Reaktionen miteinander verbunden werden..

Papin et al. TIBS 28, 250 (2003)

Die Abbildung oben links zeigt ein einfaches metabolisches Netzwerk. Die gestrichelte Linie grenzt das betrachtete System von der Umgebung ab. Es gibt 5 Metabolite A bis E im System, ein Nebenprodukt (byproduct) byp, das bei 2 Reaktionen  $v_2$  und  $v_4$  erzeugt wird und einen Kofaktor cof, der bei den Reaktionen  $v_3$  und  $v_5$  konsumiert wird und in  $v_4$  erzeugt wird. Es gibt 6 interne Reaktionen (Flüsse)  $v_1$  bis  $v_6$  und drei Transportflüsse  $b_1$  bis  $b_3$  zwischen System und Umgebung.

Beispielsweise wird bei Reaktion  $v_1$  der Metabolit A verbraucht (deswegen steht dort -1 in der Matrix) und der Metabolit B erzeugt (+1). Eigentlich konvertiert  $v_1$  zwei Einheiten von A in zwei Einheiten von B. Man könnte also auch -2 und +2 in die Matrix eintragen. Es kommt jedoch lediglich auf das Verhältnis von A und B an, das durch -1/+1 auf gleiche Weise ausgedrückt werden kann. Da  $v_2$  und  $v_3$  byp erzeugen, muss es den Transportfluss  $b_3$  geben, der byp hinaustransportiert.

## Lösungsraum der metabolischen Flüsse

Da die Zahl an Metaboliten gewöhnlich kleiner ist als die Zahl an Reaktionen ( $m < n$ ) ist die Flussgleichung üblicherweise **unterbestimmt**.

Daher gibt es im Allgemeinen eine Vielzahl an erlaubten Flussverteilungen, die die Massenerhaltung erfüllen. Dieser Satz von Lösungen entspricht dem **Nullraum** der Matrix  $S$ .

Consider

$$\begin{pmatrix} 0 & 2 & 1 \\ 3 & -1 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

$$\begin{aligned} \text{Corresponds to} \quad 2x_2 + x_3 &= 0 \\ 3x_1 - x_2 + x_3 &= 0 \end{aligned} \Leftrightarrow \begin{aligned} 2x_2 &= -x_3 \\ 2x_1 &= -x_3 \end{aligned}$$

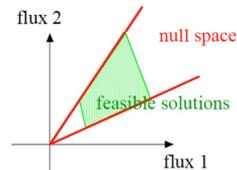
=> only one free parameter:  $x_3$

$$\text{null space: } \vec{x} = \begin{pmatrix} -a \\ -a \\ 2a \end{pmatrix}$$

Add inequalities for external fluxes  
(here, e.g.:  $x_3 \geq 0$ )

=> **feasible** solutions for  $a \geq 0$

Generally: null space is a cone,  
constraints select part of it



Softwarewerkzeuge WS 20/21 – V11

Was suchen wir eigentlich? Wir möchten alle Flussvektoren  $v$  bestimmen, die die Gleichung  $S \cdot v = 0$  erfüllen, d.h. eine gültige Lösung im Gleichgewicht sind.

Wenn  $S$  die Form einer quadratischen Matrix hätte ( $m = n$ ) und alle Reihen linear unabhängig sind (d.h. nicht der Fall eintritt, dass eine Reihe durch Linearkombination anderer Reihen dargestellt werden kann), dann besitzt die Gleichung  $S \cdot v = 0$  genau eine einzige Lösung.

Oft hat  $S$  jedoch die Form eines länglichen Rechtecks, es gibt also mehr Variablen (Komponenten des Flussvektors) als Gleichungen. Das Gleichungssystem ist daher unterbestimmt. Es gibt unendlich viele Flussvektoren  $v$  bzgl. alle Punkte in einem Lösungsraum zwischen begrenzenden Geraden (Flächen) sind gültige Lösungen.

## ***E.coli in silico***

Das am besten charakterisierte zelluläre System ist *E. coli*.

Im Jahr 2000 konstruierten Edwards & Palsson eine *in silico* Darstellung des *E.coli*- Metabolismus.

Dies erforderte eine Menge Handarbeit um die notwendigen Informationen

- aus der biochemischen Literatur,
- aus Genomannotationen und
- aus metabolischen Datenbanken wie EcoCyc und KEGG zusammenzustellen.



Bernhard Palsson,  
UC San Diego

Edwards & Palsson  
PNAS 97, 5528 (2000)

Softwarewerkzeuge WS 20/21 – V11

Wir betrachten nun ein bereits klassisches Paper von Edwards und Palsson aus dem Jahr 2000. Es wurde seither mehr als 1100 mal zitiert und behandelte natürlich *E. coli*. Dies ist ein Link auf dieses Paper:  
<https://www.pnas.org/content/97/10/5528>



## ***E.coli in silico***

Begrenze  $\alpha_i = 0$  für irreversible interne Flüsse,

$\alpha_i = -\infty$  für reversible interne Flüsse (aufgrund der biochemischen Literatur)

Transportflüsse für  $\text{PO}_4^{2-}$ ,  $\text{NH}_3$ ,  $\text{CO}_2$ ,  $\text{SO}_4^{2-}$ ,  $\text{K}^+$ ,  $\text{Na}^+$  blieben unbeschränkt.

Für die anderen Metabolite wurden Obergrenzen verwendet  $0 < v_i < v_i^{max}$   
außer für die Metabolite, die das metabolische Netzwerk verlassen können  
(d.h. Acetat, Äthanol, Laktat, Succinat, Formiat, Pyruvat etc.)

Aus dem Satz der möglichen Flussverteilungen wird mit **linearer Programmierung** eine bestimmte Lösung bestimmt, die eine bestimmte metabolische Zielfunktion  $Z$  maximiert (bzw.  $-Z$  minimiert).

$$Z = \sum c_i \cdot v_i = \langle \mathbf{c} \cdot \mathbf{v} \rangle$$

Als Zielfunktion  $Z$  wird hier die Summe aller Einzelflüsse definiert, was der Bildung maximaler Biomasse entspricht.

Edwards & Palsson, PNAS 97, 5528 (2000)

Softwarewerkzeuge WS 20/21 – V11

Aufgrund von praktischen Überlegungen gibt es für manche Flüsse (Reaktionen) Obergrenzen.

Man möchte nun natürlich nicht unendlich viele Lösungen betrachten, sondern lieber „die beste“. Dazu verwenden wir die Annahme, dass Bakterien so einfach „gestrickt“ sind, dass sie einfach so schnell wie möglich wachsen möchten. Bakterien können natürlich nicht wie wir nachdenken, ob das wirklich das Beste für sie ist. Sie bestehen ja nur aus einer Zelle und haben kein Gehirn wie wir 😊 Das Prinzip des maximalen Wachstums übersetzt man in maximale Biomasseproduktion bzw. die maximale Summe aller Einzelflüsse mit entsprechenden Koeffizienten  $c_i$ .

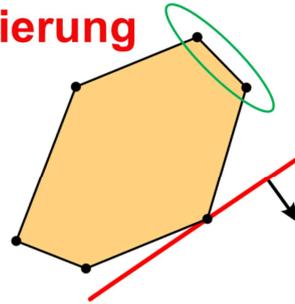
## Lineare Programmierung

Lineare Programmierung ist eine Technik um optimale Werte für eine lineare objektive Funktion zu finden, wobei lineare Bedingungen für die Gleichheit und Ungleichheit für einzelne Variablen gelten.

Der Lösungsraum dieser Bedingungen ist ein konvexes Polytop. Jede Bedingung definiert eine begrenzende Fläche.

Die objektive Funktion ist eine lineare Funktion, die auf diesem Polyeder definiert ist.

Ein Algorithmus für lineare Programmierung findet einen Punkt in diesem Polyeder, wo diese Funktion den kleinsten (oder größten) Wert annimmt.



Dargestellt ist ein einfaches lineares Programm mit 2 Variablen ( $x$  und  $y$ ) und 6 Ungleichungen.

Der Lösungsraum ist gelb gefärbt und bildet ein 2-dimensionales Polygon.

Die rote Linie repräsentiert die lineare Kostenfunktion.

Der Pfeil zeigt in die Richtung, in die wir optimieren.

[www.wikipedia.org](http://www.wikipedia.org)

Softwarewerkzeuge WS 20/21 – V11

14

Die Abbildung illustriert das Prinzip der linearen Programmierung. In diesem einfachen Fall gibt es nur 2 Variablen  $x$  und  $y$ , so dass wir das Ganze einfach in 2 Dimensionen darstellen können. Der Lösungsraum ist gelb gefärbt und wird durch 6 Geraden begrenzt. Diese 6 Geraden werden durch 6 Ungleichungen festgelegt.

Eine solche Ungleichung könnte z.B.  $x + y \leq 10$  lauten. Diese Ungleichung könnte z.B. die grün umkreiste Gerade oben rechts beschreiben. Gültige Lösungen wären  $(9,1)$ ,  $(9,0)$ ,  $(8,2)$ ,  $(8,1)$ , etc. Durch eine Zielfunktion wird innerhalb des Lösungsraum eine Optimierungsrichtung festgelegt, hier symbolisiert durch den Pfeil. Alle Punkte auf der roten Geraden würden diese Zielfunktion optimieren. Nur einer davon liegt jedoch im zulässigen Lösungsraum. Dieser ist die optimale Lösung.

## Lineare Programmierung

Lineare Programme können in folgender kanonischer Form ausgedrückt werden:

$$\begin{aligned} &\text{maximize} && \mathbf{c}^T \mathbf{x} \\ &\text{subject to} && \mathbf{Ax} \leq \mathbf{b} \\ &\text{and} && \mathbf{x} \geq \mathbf{0} \end{aligned}$$

Dabei ist  $\mathbf{x}$  der zu bestimmende Vektor der Variablen,  $\mathbf{c}$  und  $\mathbf{b}$  sind Vektoren aus (bekannten) Koeffizienten,  $A$  ist eine (bekannte) Matrix mit Koeffizienten, und  $(\cdot)^T$  ist die transponierte Matrix.

$\mathbf{c}^T \mathbf{x}$  ist die zu optimierende objektive Funktion.

$$\mathbf{c}^T = (c_1, c_2, \dots, c_k); \mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \\ \dots \end{pmatrix}; \mathbf{c}^T \mathbf{x} = c_1 x_1 + c_2 x_2 + \dots + c_k x_k$$

Die Ungleichungen  $\mathbf{Ax} \leq \mathbf{b}$  and  $\mathbf{x} \geq \mathbf{0}$  sind die Bedingungen, die den Lösungsraum definieren.

Wenn alle unbekanntem Variablen ganzzahlige Werte annehmen müssen, nennt man das Problem ein **integer lineares Programmierungs** (ILP) problem.

[www.wikipedia.org](http://www.wikipedia.org)

Allgemein kann es natürlich mehr als 2 Variablen geben. In diesem Fall packt man sie einfachheitshalber in einen Vektor  $\mathbf{x}$ .

## ***E.coli in silico***

Um die Korrektheit dieses Vorgehens zu testen, wurden Änderungen des metabolischen Netzwerkes simuliert, die sich durch hypothetische Gendeletionen ergeben.

Dafür wird einfach der Fluss durch die entsprechende enzymatische Reaktion auf 0 gesetzt.

Dann wird der optimale Wert der Mutante ( $Z_{mutant}$ ) mit dem des Wild-types verglichen ( $Z$ )

$$\frac{Z_{mutant}}{Z}$$

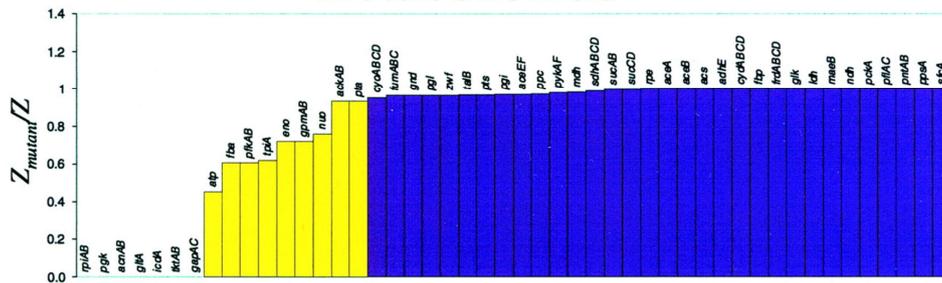
Edwards & Palsson  
PNAS 97, 5528 (2000)

Softwarewerkzeuge WS 20/21 – V11

Wie bereits erwähnt, suchen wir mit linearer Programmierung die Lösung (Kombinationen aus Flussraten für alle betrachteten Reaktionen des Systems), die zu einer möglichst großen Biomasse führt. Diese Lösung heißt  $Z$ .

In dem numerischen Reaktionssystem können wir sehr leicht den Effekt einer Gendeletion nachahmen. Wenn ein Gen deletiert wird, gibt es das davon kodierte Enzym nicht, so dass die von ihm katalysierte Reaktion nicht stattfinden kann. Wir setzen daher einfach den Fluss durch diese Reaktion auf 0.

## Gendeletionen im zentralen *E. coli* MG1655 Metabolismus



## Interpretation der Ergebnisse

Die 7 essentiellen Gene gehören zur 3-Kohlenstoff-Stufe der Glykolyse, zu 3 Reaktionen des Zitratzyklus und zu mehreren Mitgliedern des PPP.

Alle anderen Gene des zentralen Metabolismus konnten entfernt werden ohne nennenswerten Effekt auf das *in silico*-Wachstum von *E.coli*.

Edwards & Palsson PNAS 97, 5528 (2000)

Softwarewerkzeuge WS 20/21 – V11

Essentiell wichtige Pfade (zumindest in diesem Modell) waren daher die Glykolyse, der Zitratzyklus und der Pentose-Phosphat-Pfad.

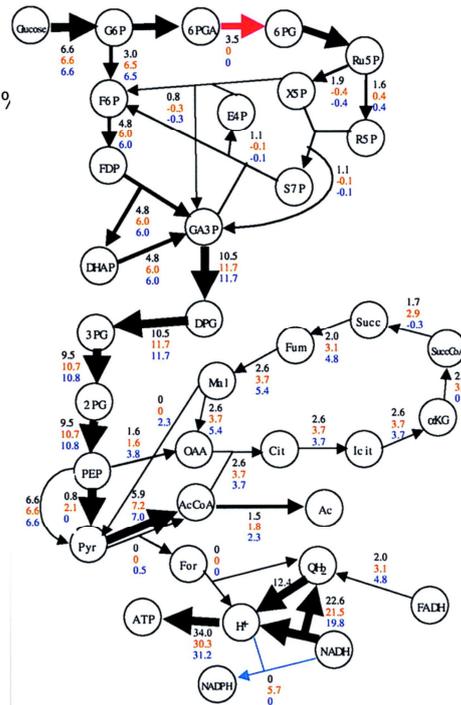
## Umlenken der metabolischen Flüsse

(Schwarz) Flussverteilung für Wildtyp.

(Rot) *zwf*- Mutante. Die Biomasse-Erzeugung ist 99% des Wildtyps.

(Blau) *zwf-pnt*- Doppelmutante. Biomasse-Erzeugung ist 92% des Wildtyps.

*E. coli in silico* umgeht die Deletion einer kritischen Reaktion (roter Pfeil) durch Erhöhung des Flusses durch die alternative G6P → P6P Reaktion.



Edwards & Palsson PNAS 97, 5528 (2000)

Softwarewerkzeuge WS 20/21 – V11

Dieses Beispiel zeigt die optimalen Flüsse (die zu maximaler Biomasseproduktion führen) in einem Ausschnitt des Gesamtsystems. Der Fluss führt von Glukose zu Glukose-6-Phosphat. Dort verzweigt sich das System. Der Glykolyse-Pfad führt nach unten zu Fruktose 6-Phosphat, der Pentose-Phosphat-Pfad nach rechts. Im Wildtyp werden 3.5 Flusseinheiten in den PPP geschickt und 3 Flusseinheiten in die Glykolyse. Im PPP kodiert das Gen *zwf* für das Enzym Glucose-6-Phosphat 1-Dehydrogenase und katalysiert die durch den roten Pfeil angedeutete Reaktion von 6PGA nach 6PG (oder so ähnlich zumindest). Wenn man diesen Fluss durch die *zwf*-Deletion auf Null setzt (siehe rote Null für diese Reaktion), wird bei G6P der gesamte Fluss nach unten in die Glykolyse umgelenkt.

## ***E. coli in silico***

+ und – bedeuten Wachstum bzw. Nichtwachstum.  
± heisst, dass Suppressormutanten beobachtet wurden, die diesem Mutantentstamm Wachstum erlaubten.

glc: Glucose, gl: Glycerol,  
succ: Succinat, ac: Acetat.

In 68 von 79 Fällen stimmte die Vorhersage mit dem Experiment überein.

Roten und gelbe Kreise kennzeichnen die vorhergesagten Mutationen, die Wachstum eliminieren bzw. reduzieren.

Edwards & Palsson  
PNAS 97, 5528 (2000)

Softwarewerkzeuge WS 20/21 – V11

Table 2. Comparison of the predicted mutant growth characteristics from the gene deletion study to published experimental results with single mutants

Gene	glc	gl	succ	ac
aceA	+/+		+/+	-/-
aceB				-/-
aceEF*	-/+			
ackA				+/+
● acn	-/-			-/-
acs				+/+
cyd	+/+			
cyo	+/+			
● eno <sup>1</sup>	-/+	-/+	-/-	-/-
● fba <sup>i</sup>	-/+			
fbp	+/+	-/-	-/-	-/-
frd	+/+		+/+	+/+
● gap	-/-	-/-	-/-	-/-
glk	+/+			
● gltA	-/-			-/-
gnd	+/+			
idh	-/-			-/-
mdh <sup>11</sup>	+/+	+/+	+/+	
ndh	+/+	+/+		
● nuo	+/+	+/+		
● pfk <sup>1</sup>	-/+			
pgi <sup>2</sup>	+/+	+/+	+/+	
● pgk	-/-	-/-	-/-	-/-
pgl	+/+			
pntAB	+/+	+/+	+/+	
ppc <sup>5</sup>	±/+	-/+	+/+	
pta				+/+
pts	+/+			
pyk	+/+			
● rpi	-/-	-/-	-/-	-/-
sdhABCD	+/+		-/-	-/-
sucAB	+/+		-/+	-/+
● tktAB	-/-			
● tpi <sup>3*</sup>	-/+	-/-	-/-	-/-
unc	+/+		±/+	-/-
zwf	+/+	+/+	+/+	

Zur Validierung des Simulationssystems verglich man die Ergebnisse der synthetischen Gendelektionen mit den experimentell bekannten Effekten. Die durch rote Kreise gekennzeichneten essentiellen Reaktionen entsprechen den Ergebnissen auf Folie 17. Die Übereinstimmung mit dem Experiment war nicht perfekt, aber eigentlich überraschend gut.

## EcoCyc FBA Modell

FBA Modell für die EcoCyc-Daten kann mit der Software MetaFlux berechnet werden.

Es gibt eine SBML-Datei davon <http://biocyc.org/download.shtml>.

Das EcoCyc FBA Modell enthält 1888 Reaktionen.

58 Metabolite produziere Biomass.

Auf einem minimalen Medium (Glukose, Ammoniak) haben 370 Reaktionen einen Fluss ungleich 0.

Für 383 Wachstumsbedingungen gibt das Modell in 72.6% der Fälle eine korrekte Vorhersage von Wachstum/Nicht-Wachstum.

Für 4207 Deletionen einzelner Gene gibt das Modell in 91.2% der Fälle eine korrekte Vorhersage von Wachstum/Nicht-Wachstum.

Latendresse M, Kruppenacker M, Trupp M, Karp PD  
*Bioinformatics* 2012;28:388-396

Softwarewerkzeuge WS 20/21 – V11

Dies ist eine aktuellere Referenz, ebenfalls von den Entwicklern der EcoCyc-Datenbank um Peter Karp. Der Effekt von Gen-Deletionen konnte sehr zuverlässig (91.2%) vorhergesagt werden, der Effekt von Wachstumsmedien nicht ganz so zuverlässig (72.6%).

Dies ist ein Link zum Artikel:

<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3268246/>

## Teil 2 – DGL-Modelle / Copasi / SBML

Aufstellen von Bilanzgleichungen

Dynamische Simulationen: Was ist das? Wozu?

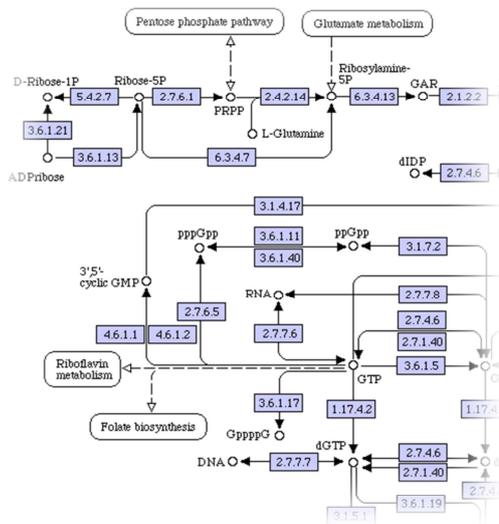
Simulations-Tool: Copasi

Vereinfachte Kinetiken: MM, Inhibierung, Hill

kinetische Daten: KEGG, SABIO-RK

Im zweiten Teil der heutigen Vorlesung beschäftigen wir uns mit Differentialgleichungsmodellen, mit denen man metabolische Prozesse dynamisch simulieren kann.

## Wdh: über die Formel zur Formel



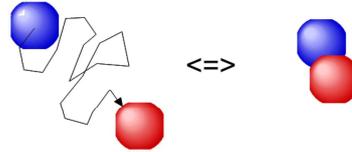
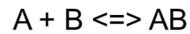
Ausschnitt aus [http://www.genome.jp/dbget-bin/show\\_pathway?ec00230+3.6.1.9](http://www.genome.jp/dbget-bin/show_pathway?ec00230+3.6.1.9)

- i) biologisches Netzwerk
- ii) Metabolite identifizieren (incl. Konzentrationen)
- iii) Einzelreaktionen aufstellen, Reaktionsraten?
- iv) Reaktionen in DGLs übersetzen
- v) Anfangswerte einsetzen und simulieren
- vi) Ergebnisse analysieren

Wir betrachten wiederum ein metabolisches System. Die Knoten sind die Metabolite, die durch enzymatische Reaktionen ineinander umgewandelt werden können.

## Massenwirkungsgesetz

Einfachste chemische Reaktion



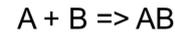
**Zeitliche Änderung von [A]:**

Gewinn: Dissoziation



$$\frac{d}{dt}[A] = G_A - L_A$$

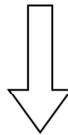
Verlust: Assoziation



AB zerfällt  
=>  $G_A$  proportional zu  $[AB]$

$$G_A = k_r [AB]$$

phänomenologischer  
Faktor



A und B müssen sich finden  
=>  $L_A$  abhängig von  $[A]$  und  $[B]$

$$L_A = k_f [A] [B]$$

$$\frac{d}{dt}[A] = k_r [AB] - k_f [A] [B]$$

In einer ersten Vorbetrachtung schauen wir uns eine einfache Assoziation zweier Moleküle A und B zu einem Komplex AB an. Die zeitliche Änderung der Konzentration von A können wir als Differenz von Gewinnen und Verlusten darstellen.

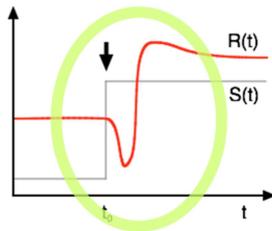
# Dynamische Simulationen

Zwei Anwendungsgebiete

zeitabhängiges Verhalten

Reaktionen des Systems auf  
Änderungen der äußeren  
Bedingungen (Randbedingungen)

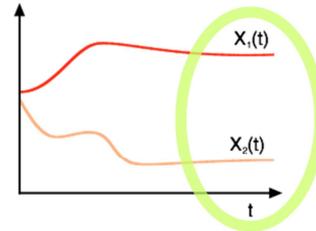
$$\frac{d\vec{X}(t)}{dt}$$



stationäre Zustände (steady state)

Suche nach Konzentrationen und  
Flüssen bei konstanten  
Randbedingungen

$$\frac{d\vec{X}(t)}{dt} = 0$$



Dynamische Simulationen sind in 2 Anwendungsfällen hilfreich. Im linken Fall möchten wir tatsächlich das zeitabhängige Verhalten analysieren, z.B. wie sich die Änderung von äußeren Einflüssen zum Zeitpunkt  $t_0$  auf das System auswirkt.

Im rechten Fall suchen wir stationäre Zustände, die sich nach einer Einschwingphase bzw. Relaxationszeit im System einstellen. Auch hierfür können dynamische Simulationen hilfreich sein, z.B. bei komplizierten Systemen, in denen man keine analytische Lösung bestimmen kann.

## Was lernt man aus dem Steady-State?

Konzentrationen = konst.

=> Zufluss = Abfluss

$$\frac{dA_2B(t)}{dt} = G_{A_2B} - L_{A_2B} = 0$$

Beispiel:  $2A + B \rightleftharpoons A_2B$

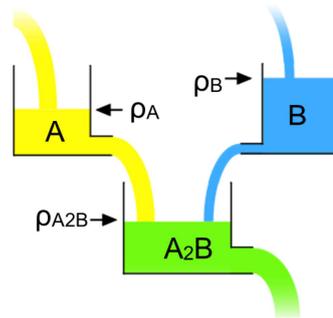
$$\frac{dA_2B(t)}{dt} = k_a A^2 B - k_d A_2B = 0$$

=> Gleichgewichts-Konzentrationen:  $A_2B = \frac{k_a}{k_d} A^2 B$

Steady state: + Bedingungen zwischen Konzentrationen und Raten

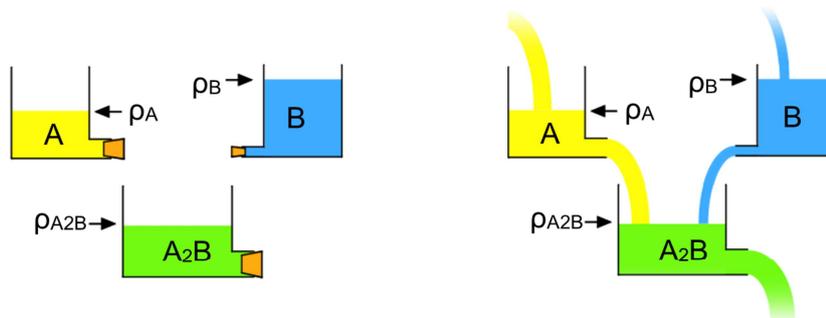
=> stationäre Betriebsmodi

- wie schnell wird der steady state erreicht?
- absolute Mengen / effektive Volumina



Im Gleichgewicht halten sich für alle Substanzen die Gewinne und Verluste die Waage, siehe die erste Gleichung. Andernfalls würde sich die Konzentration solange ändern, bis sich irgendwann doch ein Gleichgewicht einstellt.

## Statisches vs. dynamisches Gleichgewicht



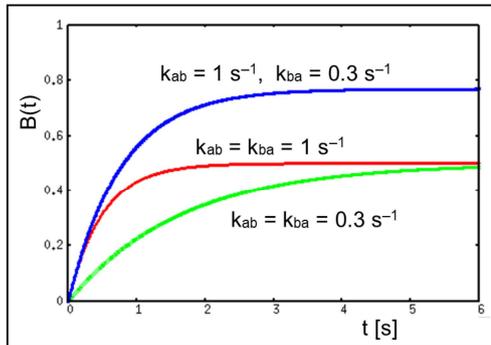
jeweils:  $\frac{d\vec{X}(t)}{dt} = 0$

Gleichgewicht bedeutet auf keinen Fall, dass kein Fluss stattfinden kann.

## Infos aus zeitabhängigen Simulationen

Ganz einfach:  $A \rightleftharpoons B$        $\frac{dA(t)}{dt} = k_{ba}B - k_{ab}A$   
 $A + B = \text{const.}$

Gleichgewicht:       $B = \frac{k_{ab}}{k_{ba}} A$



mit Anfangsbedingungen:

$$A(t=0) = A_0$$

$$B(t=0) = 0$$

$$B(t \rightarrow \infty) = \frac{k_{ab}}{k_{ab} + k_{ba}} A_0$$

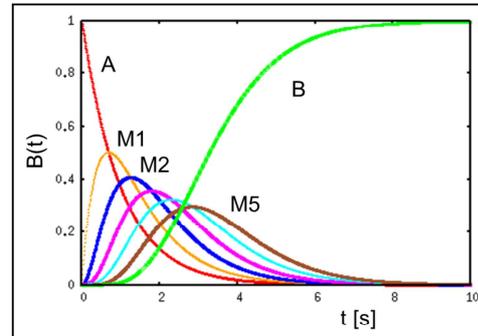
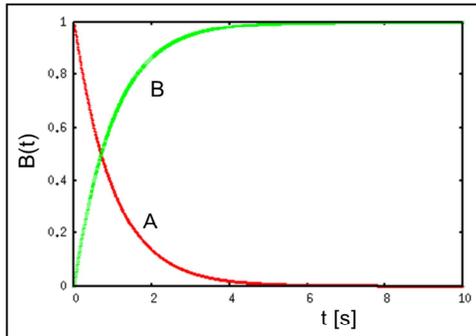
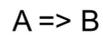
=> Gleichgewichtsverteilungen

=> wie schnell wird ein "Signal" weitergegeben?

Bei dynamischen Simulationen spielen die Raten natürlich eine wichtige Rolle. Wenn die Umwandlungsrate  $k_{ab}$  höher ist (blaue und rote Kurve), dann steigt die Konzentration von B schneller an. Die Konzentration im Gleichgewicht hängt jedoch vom Verhältnis aus Vorwärts- und Rückwärtsrate ab. Wenn  $k_{ab}$  und  $k_{ba}$  langsamer sind (grüne Kurve), dauert es länger bis sich das Gleichgewicht einstellt, erreicht aber denselben asymptotischen Wert von 0.5 wie in der roten Kurve.

## Länge von Reaktionspfaden

Vergleiche:



$$A_0 = 1, \quad k = 1 \text{ s}^{-1}$$

=> Zwischenprodukte verzögern die Antwort

=> Vorsicht beim Weglassen von Zwischenschritten

Links: bei einer einfachen Umwandlung von A nach B steigt B direkt proportional zum Verlust von A an.

Rechts: falls es Zwischenprodukte gibt, erfolgt der Anstieg von B zeitverzögert. So kann man beim Vergleich mit experimentellen Werten die Existenz von bisher unbekanntem Zwischenprodukten aufspüren.

## Complex Pathway Simulator



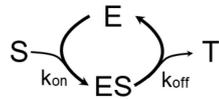
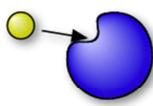
Entwickelt in den Gruppen von  
Pedro Mendes (Virginia Bioinf. Inst.) und Ursula Kummer (EML  
HD)

"COPASI is a software application for  
simulation and analysis of biochemical  
networks."

<http://www.copasi.org/>

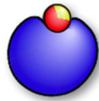
Copasi ist ein frei verfügbares Tool um die Dynamik von metabolischen Systemen zu simulieren.

## Enzyme: Michaelis-Menten-Kinetik



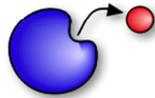
Reaktionsrate:

$$V = k_{\text{off}} ES$$



Steady state:  $k_{\text{on}} E \cdot S = k_{\text{off}} ES$

$$ES = \frac{k_{\text{on}} E \cdot S}{k_{\text{off}}} = \frac{E \cdot S}{K_M}$$



Gesamtmenge an Enzym ist konstant:

$$E_T = E + ES \quad \Rightarrow \quad ES = E_T \frac{S}{S + K_M}$$

Substratumsatz:  $V = V_{\text{max}} \frac{S}{S + K_M}$

Die Michaelis-Menten Kinetik ist ein sehr bekanntes Modell um den Substratumsatz von Enzymen zu beschreiben.

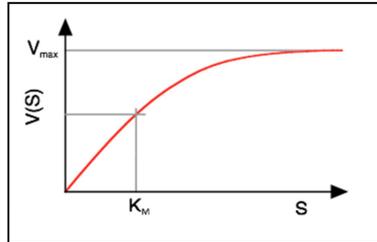
E ist die Enzymkonzentration, S die Substratkonzentration und ES die Konzentration an Enzym-Substrat-Komplex. Es gilt die einfache Annahme, dass sich im Gleichgewicht die Gewinne und Verluste an ES die Waage halten (2. Formel). Interessant ist, dass in der Formel für den Substratumsatz die Enzymkonzentration nicht auftaucht.

Eine Ableitung der Michaelis-Menten-Kinetik steht z.B. auf <https://www.chemieunterricht.de/dc2/rk/rk-mmgl.htm>

## Die Gleichung

Effektiver Umsatz nach MM:  $V = V_{max} \frac{S}{S + K_M}$

$$V_{max} = k_{off} E_T$$



$$K_M = \frac{k_{off}}{k_{on}}$$

- Vorteile:
- analytische Formel für den Umsatz
  - Interpretation der Kennlinie:  $V_{max}$ ,  $K_M$
  - Enzym kann ignoriert werden

Aber: weniger kinetische Informationen  
 $k_{on}$ ,  $k_{off}$ ,  $E_T \Rightarrow V_{max}$ ,  $K_M$

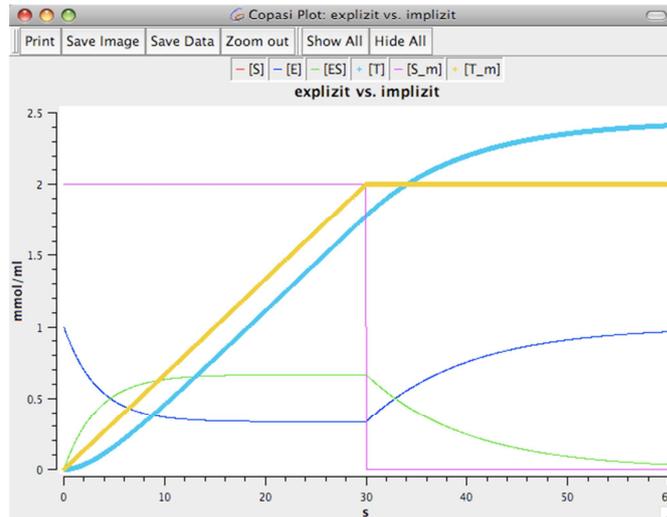
Die MM-Kinetik ergibt eine Kennlinie für die Reaktionsgeschwindigkeit (den Umsatz) als Funktion von S.

Dadurch dass sowohl  $V_{max}$  als auch  $K_M$  aus  $k_{on}$ ,  $k_{off}$  und  $E_T$  konstruiert werden, wird die kinetische Information reduziert.

## MM vs. explizite Modellierung

Wenn E verschiedene Substrate katalysiert  
=> MM geht nicht

Zeitverhalten:  
MM-Kinetik vs.  
explizite  
Modellierung  
=> Einschwingen  
=> anderer  
Gesamtumsatz



Nun betrachten wir eine dynamische Copasi-Simulation der Prozesse, die in der MM-Kinetik zusammengefasst sind.

Die lila Linie ist S, die dunkelblaue E, die grüne ES und die hellblaue das Produkt T.

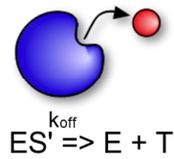
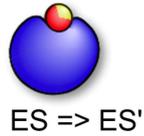
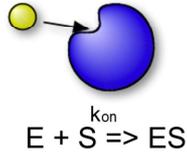
Man sieht, dass das Produkt in der dynamischen Simulation erst mit einer gewissen zeitlichen Verzögerung hergestellt wird. Zum Vergleich ergibt die analytische Formel der MM-Kinetik (gelbe Linie) einen unmittelbaren linearen Anstieg der Produktmenge.

Wenn S auf Null abgesenkt wird, stoppt die analytische MM-Kinetik sofort, wohingegen in der dynamischen Simulation ja noch ES vorliegt, das dann noch in T umgesetzt werden kann.

## Vereinfachte Kinetiken

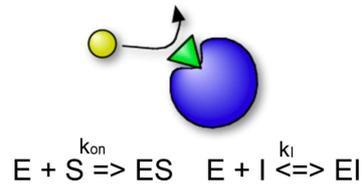
Enzymreaktion:

Michaelis-Menten

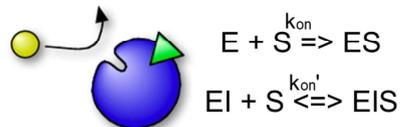


kompetitive Inhibition:

Inhibitor vs. Substrat

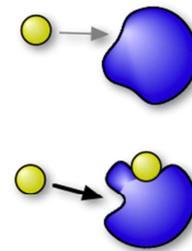


nicht-kompetitive Inhibition:  
Inhibitor verändert Enzym



Kooperative Bindung:

Hill-Kinetik

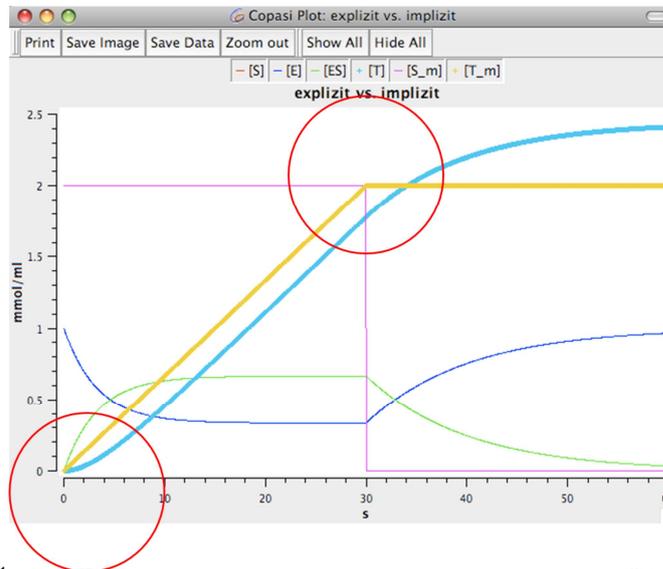


Es gibt auch andere kinetische Modelle, die hier nicht im Einzelnen diskutiert werden.

## MM vs. explizite Modellierung

Wenn E verschiedene Substrate katalysiert  
=> MM geht nicht

Zeitverhalten:  
MM-Kinetik vs.  
explizite  
Modellierung  
=> Einschwingen  
=> anderer  
Gesamtumsatz



Softwarewerkzeuge WS 20/21 – V11

35

In Fällen, in denen das Enzym E verschiedene Substrate umsetzen kann, lässt sich die MM-Kinetik nicht unmittelbar anwenden. Dann sind Simulationen, die alle Möglichkeiten parallel betrachten, vorteilhaft. Außerdem bieten Simulationen eine bessere Darstellung der zeitlichen Änderungen an Umschaltpunkten (rote Kreise).

## Wann effektive Kinetiken?

### Pro:

- weniger Aufwand  
(Modell, Simulation, Parameter)
- analytische Lösungen für  
einfache Systeme
- korrekter Steady state

### Contra:

- weniger Parameter  
=> weniger kinetische Informationen
- falsches dynamisches Verhalten

"Effektive Kinetiken sind brauchbar für langsame Signale"

"langsam" heißt dabei = Relaxationszeiten aller Zwischenschritte sind  
deutlich kürzer als Änderungen des Signals

Generell sind diese Überlegungen hilfreich um zu beurteilen, in welchen Fällen man effektive Kinetiken wie die MM-Gleichung einsetzen darf und wann nicht.

# Systems Biology Markup Language



XML-Dialekt für Speicherung und Austausch  
biochemischer Modelle

=> Archivierung

=> Transfer von Modellen in andere Softwaretools

## Acknowledgements

We are deeply indebted to the many funding agencies and organizations that have supported SBML over the years.

Since 2003, primary support for the continued development of SBML, as well as supporting software and activities, has come from the [National Institute of General Medical Sciences \(NIH/NIGMS\)](#) under grants R01 GM070923 and R01 GM077671. Additional support is provided by the [California Institute of Technology](#) (USA) and [Keio University](#) (Japan).

The development of SBML from its inception through 2003 was principally funded by the [Japan Science and Technology Agency](#) under the [ERATO Kitano Symbiotic Systems Project](#).

Additional support has in the past been provided by the following organizations and agencies: the [Systems Biology Institute](#) (Japan), the [University of Hertfordshire](#) (UK), the [Molecular Sciences Institute](#) (USA), the [National Human Genome Research Institute](#) (USA), the [International Joint Research Program of NEDO](#) (Japan), the [ERATO-SORST](#) Program of the Japan Science and Technology Agency (Japan), the [Ministry of Agriculture](#) (Japan), the [Ministry of Education, Culture, Sports, Science and Technology](#) (Japan), the [BBSRC e-Science Initiative](#) (UK), the [DARPA IPTO Bio-Computation Program](#) (USA), the Army Research Office's [Institute for Collaborative Biotechnologies](#) (USA), and the [Air Force Office of Scientific Research](#) (USA).

Nun stellen wir auf ein paar Folien noch Grundzüge der SBML-Sprache vor. Dieser Dialekt basiert auf der Webbrowser-Sprache XML und wird sehr verbreitet zur Definition und Simulation von systembiologischen Systemen verwendet.

## SBML <= XML

XML = eXtensible Markup Language

- hierarchische Baumstruktur:  
=> Schachtelung von <Object> ... </Object> oder <Objekt [Parameter...]/>
- genau ein Wurzelobjekt: <sbml...>

Aktuelle Dialekte: [siehe http://sbml.org/Documents/Specifications](http://sbml.org/Documents/Specifications)

SBML Level 1, Version 2

<http://www.sbml.org/specifications/sbml-level-1/version-2/sbml-level-1-v2.pdf>

SBML Level 2, Version 4, Release 1

<http://precedings.nature.com/documents/2715/version/1>



Öffnen Sie doch mal mit dem Chrome-Browser die Homepage der UdS, drücken Sie auf die rechte Maustaste und wählen Sie „Seitenquelltext anzeigen“ aus.

Dann sehen Sie anstelle der graphischen Darstellung die HTML-Steuerbefehle für die Webseite der Uni. Analoges gilt für SBML. Schön anzuschauen ist das nicht. Aber es funktioniert gut.

## Was ist enthalten?

beginning of model definition  
list of function definitions (optional)  
list of unit definitions (optional)  
list of compartment types (optional)  
list of species types (optional)  
list of compartments (optional)  
list of species (optional)  
list of parameters (optional)  
list of initial assignments (optional)  
list of rules (optional)  
list of constraints (optional)  
list of reactions (optional)  
list of events (optional)  
end of model definition

Was braucht man um ein systembiologisches System (model) zu definieren?  
Wie Sie sehen, sind eigentlich alle Inhalte optional, d.h. können eingefügt werden, wenn dies benötigt wird, oder weggelassen werden.

## Ein Beispiel



```

<?xml version="1.0" encoding="UTF-8"?>
<sbml level="2" version="3" xmlns="http://www.sbml.org/sbml/level2/version3">
  <model name="EnzymaticReaction">
    <listOfUnitDefinitions>
      <unitDefinition id="per_second">
        <listOfUnits>
          <unit kind="second" exponent="-1"/>
        </listOfUnits>
      </unitDefinition>
      <unitDefinition id="litre_per_mole_per_second">
        <listOfUnits>
          <unit kind="mole" exponent="-1"/>
          <unit kind="litre" exponent="1"/>
          <unit kind="second" exponent="-1"/>
        </listOfUnits>
      </unitDefinition>
    </listOfUnitDefinitions>
    <listOfCompartments>
      <compartment id="cytosol" size="1e-14"/>
    </listOfCompartments>
    <listOfSpecies>
      <species compartment="cytosol" id="ES" initialAmount="0" name="ES"/>
      <species compartment="cytosol" id="P" initialAmount="0" name="P"/>
      <species compartment="cytosol" id="S" initialAmount="1e-20" name="S"/>
      <species compartment="cytosol" id="E" initialAmount="5e-21" name="E"/>
    </listOfSpecies>
    <listOfReactions>
      <reaction id="veq">
        <listOfReactants>
          <speciesReference species="E"/>
          <speciesReference species="S"/>
        </listOfReactants>
        <listOfProducts>
          <speciesReference species="ES"/>
        </listOfProducts>
        <kineticLaw>
          <math xmlns="http://www.w3.org/1998/Math/MathML">
            <apply>
              <times/>
              <ci>cytosol</ci>
              <ci>kon</ci>
              <ci>E</ci>
              <ci>S</ci>
            </apply>
          </math>
          <listOfParameters>
            <parameter id="kon" value="1000000" units="litre_per_mole_per_second"/>
            <parameter id="koff" value="0.2" units="per_second"/>
          </listOfParameters>
        </kineticLaw>
      </reaction>
      <reaction id="vcat" reversible="false">
        <listOfReactants>
          <speciesReference species="ES"/>
        </listOfReactants>
        <listOfProducts>
          <speciesReference species="E"/>
          <speciesReference species="P"/>
        </listOfProducts>
        <kineticLaw>
          <math xmlns="http://www.w3.org/1998/Math/MathML">
            <apply>
              <times/>
              <ci>cytosol</ci>
              <ci>kcat</ci>
              <ci>ES</ci>
            </apply>
          </math>
          <listOfParameters>
            <parameter id="kcat" value="0.1" units="per_second"/>
          </listOfParameters>
        </kineticLaw>
      </reaction>
    </listOfReactions>
  </model>
</sbml>

```

Dies ist eine SBML-Version der Michaelis-Menten-Kinetik.

## Nochmal:



```
<?xml version="1.0" encoding="UTF-8"?>
<sbml level="2" version="3" xmlns="http://www.sbml.org/sbml/level2/version3">
  <model name="EnzymaticReaction">
    <listOfUnitDefinitions>
      :
    </listOfUnitDefinitions>
    <listOfCompartments>
      <compartment id="cytosol" size="1e-14"/>
    </listOfCompartments>
    <listOfSpecies>
      <species compartment="cytosol" id="ES" initialAmount="0" name="ES"/>
      <species compartment="cytosol" id="P" initialAmount="0" name="P"/>
      <species compartment="cytosol" id="S" initialAmount="1e-20" name="S"/>
      <species compartment="cytosol" id="E" initialAmount="5e-21" name="E"/>
    </listOfSpecies>
    <listOfReactions>
      :
    </listOfReactions>
  </model>
</sbml>
```

Das ist ein besser lesbarer Ausschnitt, wie das SBML-Dokument beginnt. Die Definition von Einheiten haben wir dabei übersprungen. Das Kompartiment Zytosol erhält ein Volumen, da dies für die Berechnung von Konzentrationen und Kinetiken benötigt wird. Die Menge an Metaboliten ES, Produkt P, S und E wird als absolute Teilchenzahl als Vielfaches von einem Mol ( $6 \times 10^{-23}$ ) definiert. Usw.

## Details: Einheiten

```
<listOfUnitDefinitions>
```

```
  <unitDefinition id="per_second">
```

```
    <listOfUnits>
```

```
      <unit kind="second" exponent="-1"/>
```

```
    </listOfUnits>
```

```
  </unitDefinition>
```

per\_seconds := s<sup>-1</sup>

```
  <unitDefinition id="litre_per_mole_per_second">
```

```
    <listOfUnits>
```

```
      <unit kind="mole" exponent="-1"/>
```

```
      <unit kind="litre" exponent="1"/>
```

```
      <unit kind="second" exponent="-1"/>
```

```
    </listOfUnits>
```

```
  </unitDefinition>
```

```
</listOfUnitDefinitions>
```

$\frac{\text{litre}}{\text{mol s}}$

Dies ist die Definition der verwendeten Einheiten. Der Computer weiß wirklich „fast gar nichts. Alles muss man dem sagen.“ Wie bei einem sehr kleinen Kind.

## Details: eine Reaktion

```
<listOfReactions> :
  <reaction id="vcat" reversible="false">
    <listOfReactants>
      <speciesReference species="ES"/>
    </listOfReactants>
    <listOfProducts>
      <speciesReference species="E"/>
      <speciesReference species="P"/>
    </listOfProducts>
    <kineticLaw>
      <math xmlns="http://www.w3.org/1998/Math/MathML">
        <apply>
          <times/>
          <ci>cytosol</ci>
          <ci>kcat</ci>
          <ci>ES</ci>
        </apply>
      </math>
      <listOfParameters>
        <parameter id="kcat" value="0.1" units="per_second"/>
      </listOfParameters>
    </kineticLaw>
  </reaction>
</listOfReactions>
```



$$\Rightarrow \frac{dN}{dt} = V_{\text{cytosol}} k_{\text{cat}} [ES]$$

lokaler Parameter!

Dies ist die eigentliche Mathematik, also die MM-Kinetik. Die Bezüge zwischen Reaktanden, Zwischenzustand ES und Produkt werden im braun unterlegten Teil definiert.

Die kinetische Gleichung steht im grün unterlegten Teil innerhalb von `<math>` und definiert, wie sich E + P zeitlich ändern. Nun wird also das Volumen des Zytosols benötigt.

# es gibt bereits sehr viele Modelle

BioModels is a repository of mathematical models of biological and biomedical systems. It hosts a vast selection of existing literature-based physiologically and pharmaceutically relevant mechanistic models in standard formats. Our mission is to provide the systems modelling community with reproducible, high-quality, freely-accessible models published in the scientific literature. More information about using BioModels such as [model submission](#), [update](#), [publication](#) can be found in the FAQ.

Category	Count
Submission / Update	-
Manually Curated	980 models
Non-curated	1,305 models
Auto generated	833 models
GO Chart	1,132 classes
BioModels Parameters	228,842 records

**Model of The Month** **September, 2020**

**Browse by Modelling Approach**  
This shows models distribution based on modelling approaches.  
Click on a slice to display models.

**Find us on Twitter**  
Tweets by @biomodels

Sie können sich ja mal dieses Covid-19 Modell anschauen

<https://www.ebi.ac.uk/biomodels/BIOMD0000000955>

Das in Nature Medicine publiziert wurde.