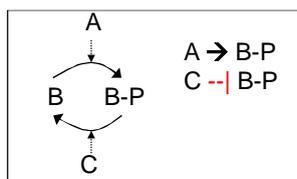


IV. Regulatorische und Signaltransduktionsnetzwerke

Wdh aus Kapitel I: Einleitung zu Signaltransduktionssystemen



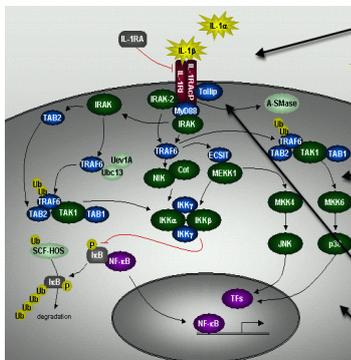
(Protein A aktiviert (z.B. phosphoryliert) Protein B)
 (Protein C inhibiert (z.B. dephosphoryliert) Protein B)

Modelle: gerichtete Vorzeichengraphen (rechts),
 Boolesche Netzwerke,
 stöchiom. Reaktionsnetzwerk (links)

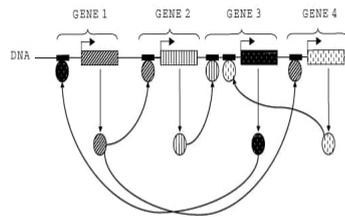
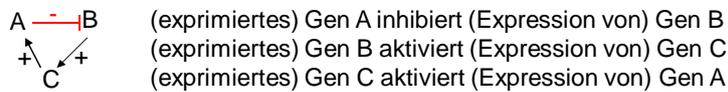
Knoten: Proteine, Metabolite, Gene

Kanten: Aktivierungs/Deaktivierungsprozesse

Signaltransduktionsnetzwerke sind gekennzeichnet durch **Signal- und Informationsflüsse**.
 (Beteiligte Partner in einer Interaktion werden nicht verbraucht oder synthetisiert sondern nur (leicht) modifiziert, z.B. Phosphoryliert.)



Wdh. aus Kapitel I: Modellierung genregulatorischer Netzwerke



■ transcription factor binding site
 in promoter region
 → coding DNA
 ○ transcription factor
 Representation of a simple, fictional transcription factor network. All genes shown encode transcription factors that control the activity of genes encoding transcription factors.
 Schitt et al. *BMC Bioinformatics* 2007 8(Suppl 6):S9 doi:10.1186/1471-2105-8-S6-S9

Modelle: gerichtete Vorzeichengraphen, Boolesche Netzwerke,

Knoten: Gene (oder deren kodierte Proteine)

Kanten: Aktivierungs/Deaktivierungsprozesse (über Transkription)

Genregulatorische Netzwerke sind ebenfalls durch **Signal- und Informationsflüsse** gekennzeichnet.

IV.1 Signalfluss vs. Massenfluss in zellulären Netzwerken

- **Massenfluss:** $A + B \rightarrow C + D$
Edukte (A, B) werden konsumiert, Produkte (C,D) entstehen neu
- **Signalfluss:** $A \rightarrow B$
ein Protein/Gen A wirkt auf/aktiviert/inhibiert ein anderes Protein/Gen B, z.B. durch Phosphorylierung; A und B werden weder verbraucht noch konsumiert (B eventuell lediglich modifiziert)

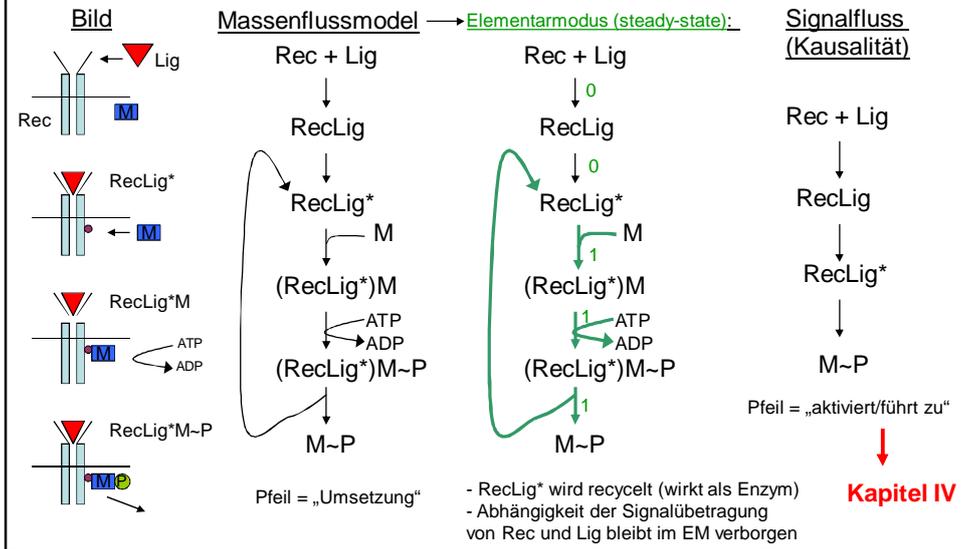
(Gen)regulatorische und Signaltransduktionsnetze sind auf unterster Ebene in Reaktionsnetzwerken (mit Massenflüssen) eingebettet. Jedoch lassen sich in der stöchiometrischen Darstellungsweise nur bedingt die für sie charakteristischen Signalflüsse identifizieren.

Was benötigt wird, sind **kausale Netzwerkmodelle**.

Probleme stöchiometrischer steady-state Analysen in Signalflossnetzwerken

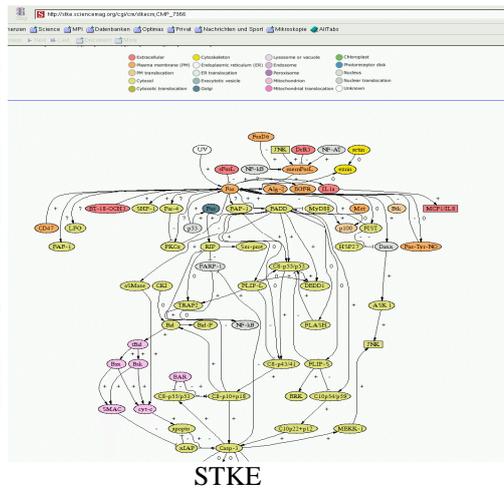
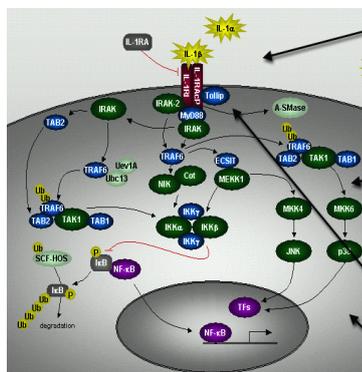
Signalübertragung erfolgt oft ohne Nettoverbrauch des Signalträgers

Bsp: Receptor-Liganden-Komplex und nachfolgende Phosphorylierung



IV.2 Interaktionsgraphen (Wirk-, Abhängigkeitsgraphen)

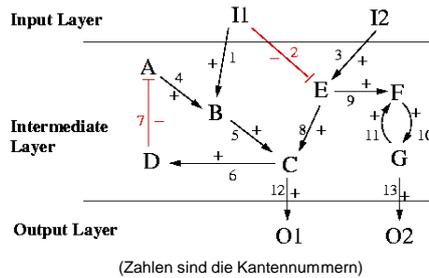
„Cartoons“ von Signaltransduktionsnetzen in Datenbanken sind typischerweise Interaktionsgraphen



IV.2.1 Definition von Interaktionsgraphen

- **Interaktionsgraph (IG):** gerichteter Graph G mit Vorzeichen für die Kanten: $G=(V,E)$; Kante $h \in E$ wird ausgedrückt durch $h=(X,Y,s)$ mit $X,Y \in V$ und Vorzeichen $s \in \{+,-\}$
- Vorzeichen gibt an, ob Anfangsknoten positiven (aktivierenden) oder negativen (inhibierenden) Einfluss auf den Endknoten ausübt

Bsp: IG1



Speicherung: wie bei „normalen“ Graphen (Adjazenz/Inzidenzmatrix ...) + Vorzeichen für die Kanten. Z.B. Inzidenzmatrix für IG1:

$$I = \begin{pmatrix} + & - & + & + & + & + & - & + & + & + & + & + & + \\ -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 1 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{matrix} I1 \\ I2 \\ A \\ B \\ C \\ D \\ E \\ F \\ G \\ O1 \\ O2 \end{matrix}$$

- IGs sind oft strukturiert in *Inputschicht* (mit *Inputs* = Knoten ohne Vorgänger), *Verarbeitungsschicht* und *Outputschicht* (mit *Outputs* = Knoten ohne Nachfolger) → Inputs/Outputs definieren Systemgrenzen

IV.2.2 Interaktionsgraphen und Jacobi-Matrix

- IGs sind qualitative Modelle für kausale Zusammenhänge (als solche werden wir sie vornehmlich betrachten) und werden aus Beobachtungen abgeleitet
- IGs können aber auch die Abhängigkeitsstruktur eines dynamischen Modells (gegeben in Form von Differentialgleichungen $\dot{\mathbf{x}} = \mathbf{F}(\mathbf{x})$) abbilden: dazu analysiert man den der Jacobi-Matrix $\mathbf{J}_F(\mathbf{x})$ zugeordneten Graphen $\mathbf{G}(\mathbf{x})$

$$\dot{\mathbf{x}} = \frac{d\mathbf{x}}{dt} = \begin{pmatrix} \dot{x}_1 \\ \vdots \\ \dot{x}_m \end{pmatrix} = \mathbf{F}(\mathbf{x}, t) = \begin{pmatrix} F_1(\mathbf{x}, t) \\ \vdots \\ F_m(\mathbf{x}, t) \end{pmatrix} \longrightarrow \text{Jacobi-Matrix: } \mathbf{J}_F(\mathbf{x}) = \frac{\partial \mathbf{F}(\mathbf{x})}{\partial \mathbf{x}} = \begin{pmatrix} \frac{\partial F_1}{\partial x_1} & \dots & \frac{\partial F_1}{\partial x_m} \\ \vdots & \dots & \vdots \\ \frac{\partial F_m}{\partial x_1} & \dots & \frac{\partial F_m}{\partial x_m} \end{pmatrix}$$

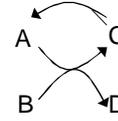
$J_{ik}(\mathbf{x}) = \frac{\partial F_i}{\partial x_k}$ ist abhängig von $\mathbf{x}(t)$ (und damit von t) und gibt an, wie Variable k die Variable i beeinflusst (wenn $J_{ik} > 0$ ($J_{ik} < 0$) dann pos. (neg.) Effekt auf Rate (DGL) von x_i ; bei $J_{ik} = 0$ kein Effekt)

Erinnere: $\mathbf{x}(t) \sim \mathbf{x}_0 + \mathbf{F}(\mathbf{x}_0) + \mathbf{J}_F(\mathbf{x}_0) \boldsymbol{\varepsilon} = \mathbf{x}_0 + \mathbf{J}_F(\mathbf{x}_0) \boldsymbol{\varepsilon}$
(Linearisierung in Umgebung von Gleichgewichtspunkt \mathbf{x}_0 (wo $\mathbf{F}(\mathbf{x}_0) = 0$) nach Auslenkung: $\mathbf{x} = \mathbf{x}_0 + \boldsymbol{\varepsilon}$)

$\mathbf{G}(\mathbf{x}) = \text{sgn}(\mathbf{J}_F(\mathbf{x})) =$ transponierte Adjazenzmatrix des der Jacobi-Matrix unterliegenden Interaktionsgraphen

Beispiel: Interaktionsgraphen und Jacobi-Matrix

Beispiel J1: $A + B \rightarrow C + D$ (Annahme: Massenwirkungsgesetz)
 $2C \rightarrow A$

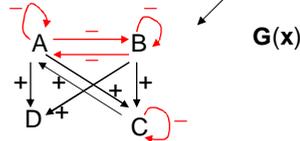


$$\dot{\mathbf{x}} = \begin{pmatrix} \dot{x}_A \\ \dot{x}_B \\ \dot{x}_C \\ \dot{x}_D \end{pmatrix} = \mathbf{F}(\mathbf{x}) = \mathbf{N}\mathbf{r}(\mathbf{x}) = \begin{pmatrix} -1 & 1 & 0 \\ -1 & 0 & 0 \\ 1 & -2 & 0 \\ 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} k_1 x_A x_B \\ k_2 x_C^2 \end{pmatrix}$$

Vorzeichenstruktur von \mathbf{J}_F hier konstant, wenn Konzentrationen \mathbf{x} und k_1, k_2 positiv

$$\mathbf{J}_F(\mathbf{x}) = \frac{\partial \mathbf{F}(\mathbf{x})}{\partial \mathbf{x}} = \mathbf{N} \frac{\partial \mathbf{r}(\mathbf{x})}{\partial \mathbf{x}} = \begin{pmatrix} -k_1 x_B & -k_1 x_A & 2k_2 x_C & 0 \\ -k_1 x_B & -k_1 x_A & 0 & 0 \\ k_1 x_B & k_1 x_A & -4k_2 x_C & 0 \\ k_1 x_B & k_1 x_A & 0 & 0 \end{pmatrix} \rightarrow \text{sgn}(\mathbf{J}_F(\mathbf{x})) = \begin{pmatrix} - & - & + & 0 \\ - & - & 0 & 0 \\ + & + & - & 0 \\ + & + & 0 & 0 \end{pmatrix} \begin{matrix} A \\ B \\ C \\ D \end{matrix}$$

Interaktionsgraph $\mathbf{G}(\mathbf{x})$ (Vorzeichenstruktur) der Jacobi-Matrix zeigt, wie sich die Zustandsvariablen im System beeinflussen



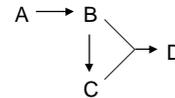
Interaktionsgraphen und Jacobi-Matrix

- es gibt Systeme, bei denen die Vorzeichenstruktur (der Interaktionsgraph) der Jacobi-Matrix invariant ist, d.h. $\mathbf{G}(\mathbf{x})$ konstant für alle \mathbf{x}

- siehe vorhergehendes Beispiel J1

- die Vorzeichenstruktur $\mathbf{G}(\mathbf{x})$ kann aber i.A. für verschiedene \mathbf{x} variieren und sich daher im transienten Verlauf auch verändern

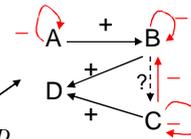
Beispiel J2: $A \rightarrow B$ (Annahme: Massenwirkungsgesetz, irreversible Reaktionen)
 $B \rightarrow C$
 $B + C \rightarrow D$



$$\dot{\mathbf{x}} = \begin{pmatrix} \dot{x}_A \\ \dot{x}_B \\ \dot{x}_C \\ \dot{x}_D \end{pmatrix} = \mathbf{F}(\mathbf{x}) = \mathbf{N}\mathbf{r}(\mathbf{x}) = \begin{pmatrix} -1 & 0 & 0 \\ 1 & -1 & -1 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} k_1 x_A \\ k_2 x_B \\ k_3 x_B x_C \end{pmatrix}$$

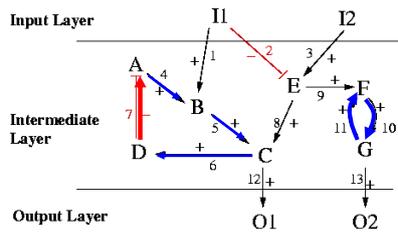
Vorzeichenstruktur von $\mathbf{J}_F(\mathbf{x})$ nur zum Teil konstant $\rightarrow \mathbf{G}(\mathbf{x})$ hängt von \mathbf{x} ab

$$\mathbf{J}_F(\mathbf{x}) = \frac{\partial \mathbf{F}(\mathbf{x})}{\partial \mathbf{x}} = \mathbf{N} \frac{\partial \mathbf{r}(\mathbf{x})}{\partial \mathbf{x}} = \begin{pmatrix} -k_1 & 0 & 0 & 0 \\ k_1 & -k_2 - k_3 x_C & -k_3 x_B & 0 \\ 0 & k_2 - k_3 x_C & -k_3 x_B & 0 \\ 0 & k_3 x_C & k_3 x_B & 0 \end{pmatrix} \rightarrow \text{sgn}(\mathbf{J}_F(\mathbf{x})) = \begin{pmatrix} - & 0 & 0 & 0 \\ + & - & - & 0 \\ 0 & ? & - & 0 \\ 0 & + & + & 0 \end{pmatrix} \begin{matrix} A \\ B \\ C \\ D \end{matrix}$$



IV.2.3 Feedback Loops (Rückkopplungsschleifen)

- **Feedback Loops (FL):** sind gerichtete Kreise im Interaktionsgraphen (erinnere: Kreis ist Pfad, bei dem Anfangs- und Endknoten gleich sind und alle anderen Knoten nur einmal durchlaufen werden)
alternative Def.: *minimales Teilnetz*, über das eine Spezies sich selbst beeinflusst
- jeder FL hat positives oder negatives Vorzeichen, was sich durch Multiplikation der Vorzeichen der beteiligten Kanten ergibt



- **positiver feedback loop:**
gerade Anzahl an “-” Kanten
Beispiel: Kantenfolge (10,11)
- **negativer feedback loop:**
ungerade Anzahl an “-” Kanten
Beispiel: Kantenfolge (4,5,6,7)

FL sind von zentraler Bedeutung für das dynamische Verhalten eines Systems

	Positive FL	Negative FL
# negativer Kanten	gerade Anzahl	ungerade Anzahl
Beispiel		
Dynamische Eigenschaften	<ul style="list-style-type: none"> • wirken verstärkend • können multistabiles Verhalten erzeugen → der IG der Jacobi-Matrix $J_F(x)$ eines bi-/multi-stabilen Systems muß für mindestens ein x positiven FL haben 	<ul style="list-style-type: none"> • wirken stabilisierend, abschwächend → IG der Jacobi-Matrix $J_F(x)$ eines Systems mit zyklischem oder Punktattraktor muß für mindestens ein x negativen Kreis haben
Biologische Bedeutung	<p>Differenzierung, Entscheidungsprozesse</p>	<p>Homöostase, Oszillationen</p>

Feedback Loops im Interaktionsgraphen der Jacobi-Matrix

Positive Feedback Loops sind Voraussetzung für Multistationarität

Theorem 1:

(auch bekannt als Thomas-Conjecture, vollständig bewiesen erst 2003)

Gegeben sei ein (i.A. mehrdimensionales) dynamisches System mit Differentialgleichung $\mathbf{x}' = \mathbf{F}(\mathbf{x})$. Man betrachte offenes (mehrdimensionales) Intervall Ω im Vektorraum der Zustandsgrößen (hier Konzentrationen) \mathbf{x} . Wir nehmen an, dass $\mathbf{F}(\mathbf{x})$ stetig differenzierbar ist, d.h. die partiellen Ableitungen $\partial F_i / \partial x_j(\mathbf{a})$ für ein beliebiges $\mathbf{a} \in \Omega$ existieren und bilden die Jacobi-Matrix $\mathbf{J}_F(\mathbf{a})$.

Gibt es mindestens zwei verschiedene (nicht-degenerierte) Gleichgewichtspunkte \mathbf{x} mit $\mathbf{F}(\mathbf{x}) = \mathbf{0}$ so gibt es mindestens ein \mathbf{a} , sodass $\mathbf{G}(\mathbf{a}) = \text{sgn}(\mathbf{J}_F(\mathbf{a}))$ einen positiven Kreis hat. (Oder: hat $\mathbf{F}(\mathbf{x})$ an keiner Stelle des Intervalls einen positiven Kreis, so ist \mathbf{F} injektiv).

Bemerkungen:

- ein positiver Kreis ist also eine *notwendige* (aber nicht hinreichende) Bedingung für Multistationarität
- ist die Vorzeichenstruktur $\text{sgn}(\mathbf{J}_F(\mathbf{x}))$ (der Interaktionsgraph $\mathbf{G}(\mathbf{x})$) invariant, kann sehr leicht getestet werden, ob Multistationarität überhaupt auftreten kann

Feedback Loops im Interaktionsgraphen der Jacobi-Matrix

Negative Feedback Loops sind Voraussetzung für periodisches Verhalten

Theorem 2:

Gegeben sei ein (i.A. mehrdimensionales) dynamisches System mit Differentialgleichung $\mathbf{x}' = \mathbf{F}(\mathbf{x})$. Man betrachte offenes (mehrdimensionales) Intervall Ω im Vektorraum der Zustandsgrößen (hier Konzentrationen) \mathbf{x} . Wir nehmen an, dass $\mathbf{F}(\mathbf{x})$ stetig differenzierbar ist, d.h. die partiellen Ableitungen $\partial F_i / \partial x_j(\mathbf{a})$ für ein beliebiges $\mathbf{a} \in \Omega$ existieren und bilden die Jacobi-Matrix $\mathbf{J}_F(\mathbf{a})$. Wir nehmen hier ferner an das $\mathbf{G}(\mathbf{x}) = \text{sgn}(\mathbf{J}_F(\mathbf{x}))$ konstant (unabhängig von \mathbf{x}) ist

Hat das System eine stabile periodische Lösung (einen stabilen Grenzyklus), so gibt es in $\mathbf{G}(\mathbf{x})$ mindestens einen negativen Kreis der Länge 2 oder größer.

Bemerkungen:

- negativer Kreis der Länge 2 ist *notwendige* (aber nicht hinreichende) Bedingung für stabiles periodisches Verhalten (oft dienen negative Kreise der Homöostase)
- der Satz gilt nicht bei nichtkonstanter Vorzeichenstruktur $\mathbf{G}(\mathbf{x})$!

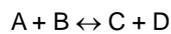
Feedback Loops im Interaktionsgraphen der Jacobi-Matrix

Beispiele für Anwendung von Theorem 1 und 2

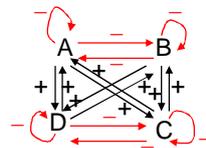
- $A \rightarrow B \rightarrow C \rightarrow \dots \rightarrow Z$: weder Multistationarität noch Oszillationen möglich
- Beispiele J1 + J2 (siehe Folie vorn): Multistationarität und Oszillationen theoretisch möglich, da pos. und neg. Feedback Loops existieren

Es gibt einige Fälle, wo sich Theoreme 1 und 2 sinnvoll einsetzen lassen, insbesondere in genetischen Netzwerken, wo keine Massenwirkungskinetik, sondern Inhibierungs/Aktivierungskinetik (z.B. Hill-Funktion) verwendet wird.

Leider treten auch in einfachen Reaktionsnetzwerken häufig triviale positive und negative Kreise in der Jacobi-Matrix auf (insbes. durch bi-molekulare und reversible Reaktionen).



Interaktionsgraph bei
Massenwirkungskinetik
→ voll verknüpft; viele
positive + negative Kreise



Es gibt verschiedene Ansätze die gerade für stöch. Netzen stärkere Aussagen zu Multistationarität und Oszillationen liefern, z.B. Craciun et al. und Feinberg et al. („Chemical Reaction Network Theory“). Damit kann z.B. gezeigt werden, dass J1 und J2 nicht multistabil sein können.

Berechnung (Enumeration) von Feedback Loops

- Anzahl der Kreise in einem gerichteten Graphen kann exponentiell wachsen
- mehrere Möglichkeiten zur Berechnung (Enumeration) aller Kreise, z.B.
 - a) Über Tiefensuche (Tarjan, Johnson)
 - b) Elementarmodenalgorithmus

Berechnung (Enumeration) von Feedback Loops: Alternativ über EM-Algorithmus

- anderer Ansatz: nutze konzeptuellen Zusammenhang zu Elementarmoden (EMen):
 - repräsentiere Kreise als Vektoren \mathbf{e} mit $|\mathbf{E}|$ Komponenten und setze $e_i=1$ für alle Kanten i die im Kreis durchlaufen werden und sonst $e_i=0$
 - Kreise \mathbf{e} sind durch folgende drei Eigenschaften (die denen der EMen äquivalent sind) definiert:
 - 1) $\mathbf{I}\mathbf{e} = \mathbf{0}$ (\mathbf{I} = Inzidenzmatrix);
 - 2) $\mathbf{e} \geq \mathbf{0}$ (per Definition)
 - 3) Es gibt keinen Kreis \mathbf{c} mit $\text{support}(\mathbf{c}) \subset \text{support}(\mathbf{e})$ (sonst \mathbf{e} kein Kreis)
 - Bsp: man überprüfe, dass $(0,0,0,1,1,1,1,0,0,0,0)^T$ der negative Kreis in IG1 ist
 - alle Vektoren, die Eigenschaft (1) erfüllen, heissen Zirkulation (vgl. mit stationäre Flussverteilungen).
 - Kreise können über EM-Algorithmus berechnet werden (s. Kap. III);
Hinweis: normiere nach Berechnung die Komponenten des Vektors \mathbf{e} auf 1;
Vorzeichen eines Kreises ergibt sich aus Multipl. der Kanten-Vorzeichen
 - Viele Eigenschaften der EMen übertragen sich auf Kreise/Feedback Loops
 - Tarjan Algorithmus zur Berechnung aber gewöhnlich schneller (bessere Ausnutzung der Graph-Eigenschaften)

Feedback Loops und Minimale Cut Sets

Viele Eigenschaften der EMen übertragen sich auf FL, z.B.:

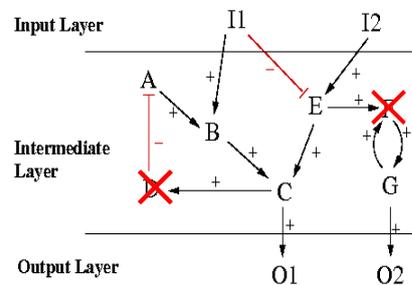
- Bedeutung (Auftrittshäufigkeit) einzelner Knoten/Kanten für Feedback-Struktur
- Berechnung von minimal cut sets ...

Minimal cut sets (für IG): minimal Menge an Schnitten, die eine vorgegebene Auswahl an feedback loops unterbricht

(prinzipiell kann man Kanten oder Knoten als Eingriffsstelle betrachten, aber in Signalflossnetzen sind Knoten meist relevanter)

Beispiel: Unterbreche *alle* feedback loops in IG1 (durch Schnitte in Knoten)

→ 8 MCSs: $\{A,B,C,D\} \times \{F,G\}$
z.B. $\{D,F\}$



Berechnung: minimal hitting sets (Kap. III)

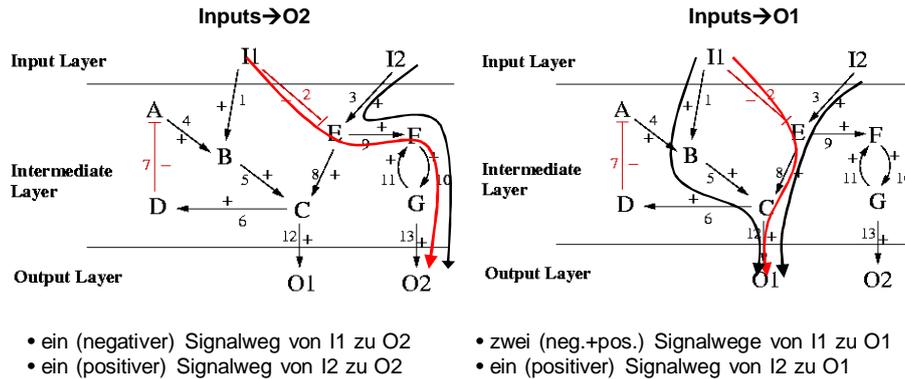
Anwendungen:

- Unterdrücken von Oszillationen (unterbreche negative FL)
- Unterdrücke bistabiles Verhalten (unterbreche positive FL)

IV.2.4 Signalwege im Interaktionsgraphen

- **Signalweg:** ist gerichteter Weg im Interaktionsgraphen zwischen vorgegebenen Start- und Zielknoten
[alternative Def.: *minimales Teilnetz*, über das eine Spezies eine andere beeinflusst]
- jeder Signalweg hat positives oder negatives Vorzeichen, was sich durch Multiplikation der Vorzeichen der beteiligten Kanten ergibt

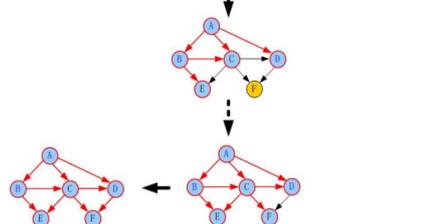
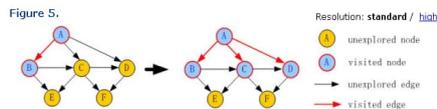
Bsp: alle Signalwege in IG1, die Input- mit Outputknoten verbinden



Berechnung (Enumeration) von Signalwegen

- a) klassisch (Graphentheorie):
Breiten/Tiefensuche

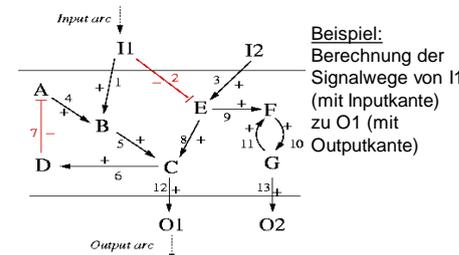
(Vorzeichen der Wege werden wieder am Ende zugewiesen)



A simple example is shown to illustrate how breadth-first search method works. Given a network G, first label all the nodes of G and get its parent index. Then generate paths starting from a root node. The nodes with shorter distance to the root node are explored preferentially. Paths starting from root to target are returned when all the nodes are visited. In the network below, take A as the root node and E, F as target nodes. The resulting path sets are {A,B,E}, {A,C,E}, {A,B,C,E}, {A,C,F}, {A,D,F}, {A,C,D,F}, {A,B,C,F} and {A,B,C,D,F}.
Liu et al. BMC Bioinformatics 2006 7:515 doi:10.1186/1471-2105-7-515

- b) alternativ (lineare Algebra):
Elementarmodenansatz

füge in der Inzidenzmatrix Inputkante (Null-Spalte mit einer +1 am Startknoten X) und Outputkante (Null-Spalte mit einer -1 am Zielknoten Y) ein und benutze Elementarmoden-Algorithmus wie bei Kreisen (liefert dann alle Kreise und die input/output-Signalwege)



Dies zeigt nochmals die konzeptuelle Ähnlichkeit zwischen Elementarmoden (Massenflussnetze) und Feedback Loops/Signalwegen in Signalflossnetzen. Von den Laufzeiten her sind aber gewöhnlich klassische Breiten/Tiefensuche (wie Tarjan) zu bevorzugen.

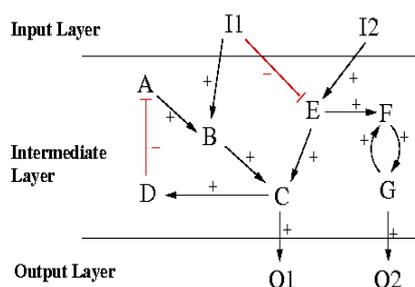
Anwendung von Signalwegen

Berechnung aller Signalwege zwischen Knoten X (Start) und Y (Ziel, Target) zeigt:

- auf welchen Wegen kann X die Spezies Y (indirekt) beeinflussen kann (Beachte: jedes Paar von Signalwegen von X nach Y bilden Feedforward Loop)
- welche Elemente (Knoten/Kanten) sind wichtig/essentiell für Signalübertragung von X nach Y
- Weglängen: welches sind die kürzesten (~"schnellsten") Wege
- Vorhersagen qualitativer Effekte von Perturbationen falls zwischen X und Y nicht pos. *und* negative Wege gleichzeitig existieren → Klassifikation der Abhängigkeit zwischen X und Y ...

IV.2.5 Netzwerkweite Abhängigkeiten in Interaktionsgraphen

Klassifiziere für ein geordnetes Paar von Knoten (X,Y) den Einfluss von X auf Y:



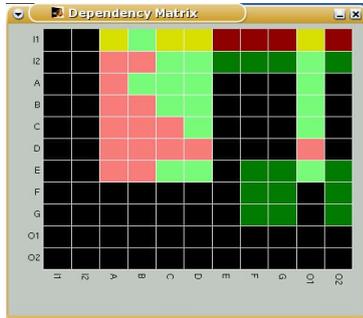
- X ohne Einfluss auf Y: kein Signalweg von X nach Y (z. B.: B → F)
- X ist Aktivator von Y: alle Wege von X nach Y sind positiv (z.B. I2 → O2)
- X ist Inhibitor von Y: alle Wege von X nach Y sind negativ (z.B. I1 → O2)
- X ist ambivalenter Faktor für Y: mindestens ein Weg von X zu Y ist positiv und mindestens einer ist negativ (z.B. I1 → O1)

- Verfeinerte Klassifizierung von Aktivatoren und Inhibitoren:
X ist starker Aktivator/Inhibitor von Y falls X Aktivator/Inhibitor von Y ist und kein Weg von X nach Y über einen Knoten läuft, der in einem negativen Kreis liegt (sonst ist X ein schwacher Aktivator/Inhibitor von Y)

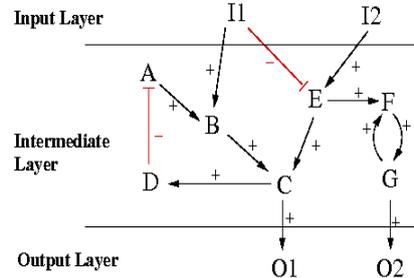
Bsp: I2 ist starker Aktivator von E,F,G and O2 aber schwacher Aktivator von C und O1

- Beachte: Klassifikation auch für (X,X) – hierfür nehme man die Feedback Loops

Abhängigkeitsmatrix



- starker Aktivator
- schwacher Aktivator
- starker Inhibitor
- schwacher Inhibitor
- ambivalenter Faktor
- ohne Einfluss

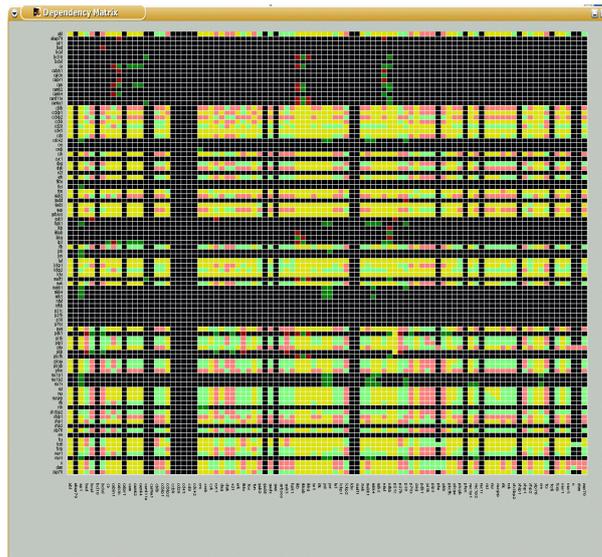


Prädiktion qualitativer Effekte nach Perturbationen (aus Ruhelage heraus)

- I2 ist starker Aktivator von O2: O2-Level reagiert positiv (negativ) auf positive (negative) Änderung des I2-Levels
- I2 ist schwacher Inhibitor von A: A-Level reagiert in einem Zeitfenster $[0, t_d]$ negativ (positiv) auf positive (negative) Änderung des I2-Levels
- F ohne Einfluss auf O1: O1-Level bleibt unverändert wenn F-Level verändert wird
- I1 ambivalenter Faktor für O1: O1-Level reagiert positiv oder negativ auf Änderung des I1-Levels (hängt von Kinetik ab)

Abhängigkeitsmatrix

- zahlreiche qualitative Prädiktionen für Perturbationsexperimente in großen Netzwerken, die falsifiziert werden können ("wertlos": ambivalente Faktoren)



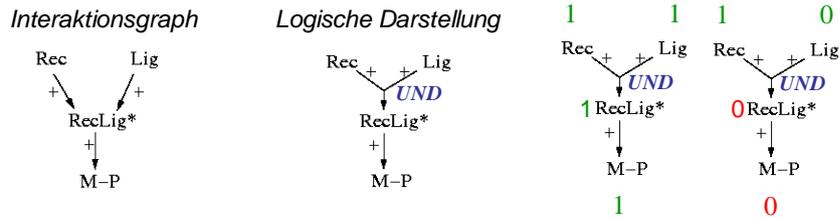
Bsp: großes Netzwerk mit 94 Knoten

IV.3 Logische (Boolesche) Netzwerke

Motivation:

- Interaktionsgraphen sind nicht deterministisch (z.B. wie werden mehrere Eingänge in einen Knoten kombiniert, d.h. welche Kombinationen führen zur Aktivierung?)
- UND-Verknüpfungen in Graphen nicht darstellbar

Bsp: Rezeptor-Liganden-Komplex (siehe Kap. IV.1)



- sind Rezeptor (Rec) und Ligand jeweils allein oder nur gemeinsam in der Lage, Molekül M zu phosphorylieren?
(natürlich nur gemeinsam – aber das wird aus der Darstellung im IG nicht klar)

- Rezeptor (Rec) und Ligand (Lig) werden benötigt, um Molekül M zu phosphorylieren
- diskrete Simulation möglich

Sind Boolesche Ausdrücke für biologische Interaktionen überhaupt gerechtfertigt?

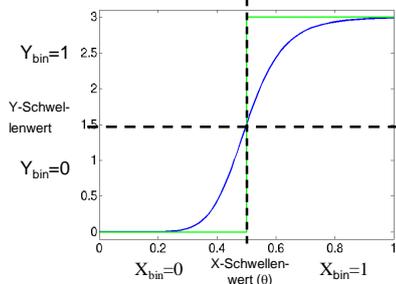
Viele Interaktionen (Aktivierungen/Inhibierungen) haben sigmoide Kennlinien (z.B. Hill-Kinetik) und sind daher annähernd als Schalterfunktion beschreibbar
→ diskrete Approximation mit “aus” (0) und “an” (1)

Aktivierung

(Bsp: Phosphorylierung des Enzyms Y durch Kinase X)

(steigende) Hill-Kinetik: $y = H^+(x) = v \frac{x^n}{\theta^n + x^n}$

(n bestimmt Steilheit, v die Höhe; hier n=8, v=3, θ=0.5)

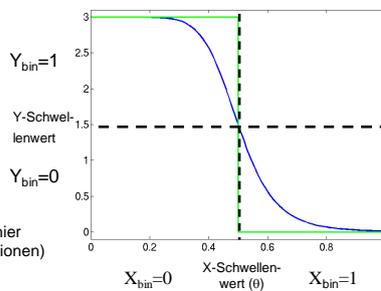


Logik:
 $x < 0.5 (X_{bin}=0) \rightarrow y < 1.5 (Y_{bin}=0)$
 $x > 0.5 (X_{bin}=1) \rightarrow y > 1.5 (Y_{bin}=1)$

Inhibierung

(Bsp: Dephosphorylierung von Y durch Phosphatase X)

(fallende) Hill-Kinetik: $y = H^-(x) = v \frac{\theta^n}{\theta^n + x^n}$



Logik:
 $x < 0.5 (X_{bin}=0) \rightarrow y > 1.5 (Y_{bin}=1)$
 $x > 0.5 (X_{bin}=1) \rightarrow y < 1.5 (Y_{bin}=0)$

Multi-Level Logik: Es können auch mehrere logische (0,1,2, ...) Zustände für die Knoten betrachtet werden, z.B. wenn eine Komponente X eine Spezies Y mit Schwellenwert X_1 und eine andere Spezies Z mit Schwellenwert X_2 sigmoid aktiviert.

$$\text{Bsp: } X \geq 1 \rightarrow Y=1; X=2 \rightarrow Z=1$$

Im folgenden wird nur der binäre Fall betrachtet.

Übertragung auf Multi-Level ist aber leicht zu bewerkstelligen.

IV.3.1 Definitionen

Logisches (Boolesches) Netzwerk: Tupel $L=(V,F)$

- Menge von Knoten $V=\{v_1, \dots, v_n\}$ (=Gene, Proteine, Rezeptor, ...), die als Boolesche Variablen mit Zuständen $x_1, \dots, x_n \in \{0,1\}$ aufgefasst werden (weitere diskrete Werte im Falle von Multi-Level-Variablen möglich).
- Liste von Booleschen (logischen) Funktionen $F=(F_1, \dots, F_n)$ mit $F_i: \mathbf{x} \rightarrow \{0,1\}$.

[Manchmal schreibt man als Argument für die Boolesche Funktion F_i präziser nur diejenigen Knoten $x_{i_1}, x_{i_2}, \dots, x_{i_k} \in V$ auf, die im Interaktionsgraph wirklich auf v_i wirken, also: $x_i = F_i(x_{i_1}, x_{i_2}, \dots, x_{i_k})$]

Boolesche Funktionen:

- verknüpft Boolesche Variablen mit Booleschen Operatoren
- Boolesche Operatoren: NOT, OR, AND, XOR, NAND,
- Boolesche Funktionen können oft mehrdeutig geschrieben werden:

$$\text{Bsp: } x = (z \text{ AND NOT}(y)) \text{ OR } (\text{NOT}(z) \text{ AND } y) = z \text{ XOR } y$$

- jede Boolesche Funktionen lässt sich (für viele biologische Interaktionen intuitiv) als *Summe von Produkten (disjunktive Normalform)* darstellen: diese verwendet nur NOT (!), OR (+) und AND (·) und verknüpft AND-Terme mit OR:

$$\text{Bsp1) } x = z \cdot y + !z \cdot y \quad (\text{ist wie oben } x = z \text{ XOR } y)$$

$$\text{Bsp2) } x = z + !u \cdot y + !z \cdot v$$

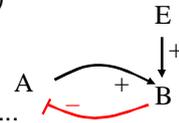
IV.3.2 Aufstellen von logischen Netzwerken biologischer Interaktionen

Gewöhnlich startet die Konstruktion des logischen Netzes vom Interaktionsgraphen:

Schritt 1: Interaktionsgraph (aus qualitativer Beobachtung)

- Bsp: Gen A beeinflusst Gen B positiv
- Gen B beeinflusst Gen A negativ
- Gen E beeinflusst Gen B positiv

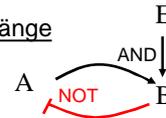
→ Signalwege, Feedback Loops, Abhängigkeitsmatrix, ...



Schritt 2: Präzisiere logische Verknüpfung mehrfacher Eingänge

(= stelle Boolesche Funktionen für jeden Knoten auf)

- Bsp: $A = !B$
- $B = A \cdot E$
- $E = E_0$ (Input)



Logische Netzwerke können auf unterschiedliche Art und Weise dargestellt werden:

- i) bzgl. der logischen Funktionen (z.B. disjunktive oder konjunktive Normalform)
- ii) graphische Darstellung
 - a) schreibe logische Funktion an den Interaktionsgraphen (siehe Grafik oben)
 - b) verwende disjunktive Normalform für die logischen Funktionen und stelle Netz als (gerichteten + vorzeichenbehafteten) **Hypergraphen** dar

Logische Netzwerke als Hypergraphen

Logische Netzwerke, deren logische Funktionen in disjunktiver Normalform vorliegen, können direkt als gerichtete Hypergraphen dargestellt werden:

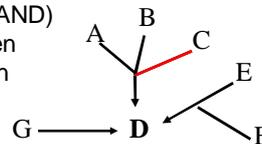
Bsp: Logische Funktion für Knoten D:

- D aktiv falls "G aktiv OR (E AND F aktiv)
- OR (A AND B aktiv AND C inaktiv)

$$D = G + E \cdot F + A \cdot B \cdot !C$$

Darstellung als Hypergraph

- ANDs: Hyperkanten (verknüpfen ihre Startknoten mit AND)
- ORs: jede Hyperkante, die in einen bestimmten Knoten zeigt, ist eine (unabh.) Möglichkeit diesen zu aktivieren
- NOTs: werden gesondert an den Zweigen der Hyperkanten markiert (z.B. „-“ oder rote Farbe)



Logische Netze als gerichtete Hypergraphen:

- jede Hyperkante hat genau einen Endknoten
- jeder Startknoten einer Hyperkante hat zugeordnetes Vorzeichen („-“ heißt NOT)
- die Startknoten einer Hyperkante werden entsprechend ihres Vorzeichens (mit/ohne NOT) verUNDet
- die Boolesche Funktion eines Knotens ergibt sich aus VerODERung aller der in den Knoten zeigenden Hyperkanten

$$\begin{aligned}
 H &= (V, E) \\
 &= (\{A, B, C, D, E, F, G\}, \\
 &\quad \{ \{ \{ (G, +) \}, \{ D \} \}, \\
 &\quad \{ \{ (A, +), (B, +), (C, -) \}, \{ D \} \}, \\
 &\quad \{ \{ (E, +), (F, +) \}, \{ D \} \} \})
 \end{aligned}$$

Vorteile der Hypergraphen-Darstellung:

- intuitiv les- und interpretierbar;
- bequem speicherbar mit 2 Matrizen (oder Listen): Inzidenzen+NOTs
- unterliegender Interaktionsgraph leicht ableitbar: splitte die ANDs auf!

Zurück zu unserem Beispiel:

Hypergraph-Darstellung:



$$\begin{aligned}
 H &= (V, E) \\
 &= (\{A, B, E\}, \\
 &\quad \{ \{ (A, +), (E, +) \}, \{B\} \}, \\
 &\quad \{ \{ (B, -) \}, \{A\} \} \})
 \end{aligned}$$

Inzidenz/NOT-Matrix des Hypergraphen

$$\mathbf{I} = \begin{pmatrix} -1 & 1 \\ 1 & -1 \\ -1 & 0 \end{pmatrix} \begin{matrix} \rightarrow A \\ \rightarrow B \\ \rightarrow E \end{matrix}, \quad \text{NOTs} = \begin{pmatrix} 0 & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix} \begin{matrix} \rightarrow A \\ \rightarrow B \\ \rightarrow E \end{matrix}$$

IV.3.3 Dynamik Boolescher Netzwerke

Synchrones/Asynchrones Schalten in logischen Netzen

Wenn man ein logisches Netzwerk definiert und sein (diskretes) dynamisches Verhalten untersuchen will, muss man sich für synchrones oder asynchrones Schaltverhalten entscheiden. Das Schaltverhalten bestimmt wie Zustand $\mathbf{z}(t+1)$ aus $\mathbf{z}(t)$ folgt.

Synchrones/asynchrones Schaltverhalten lässt sich am besten über die Wahrheitstabelle des logischen Netzes erklären.

Wahrheitstabelle

Zustand (Z)		Bild=f(Z)			
e	a	b	E	A	B
0	0	0	0	1	0
0	0	1	0	0	0
0	1	0	0	1	0
0	1	1	0	0	0
1	0	0	1	1	0
1	0	1	1	0	0
1	1	0	1	1	1
1	1	1	1	0	1

Logisches Steady State (LSS)

$\mathbf{z} = \mathbf{F}(\mathbf{z})$ System verharrt

Schaltvorgang falls $\mathbf{z} \neq \mathbf{F}(\mathbf{z})$

Synchron: $\mathbf{z}(t+1) = \mathbf{f}(\mathbf{z}(t))$;
alle $z_i \neq F_i(\mathbf{z})$ schalten gleichzeitig

Asynchron: es schaltet immer nur ein $z_i \neq F_i(\mathbf{z})$;
nichtdeterministisch!

$\mathbf{z}(t) \quad \mathbf{z}(t+1)$
011 \rightarrow 000

$\mathbf{z}(t) \quad \mathbf{z}(t+1)$
011 \rightarrow 010
 \rightarrow 001

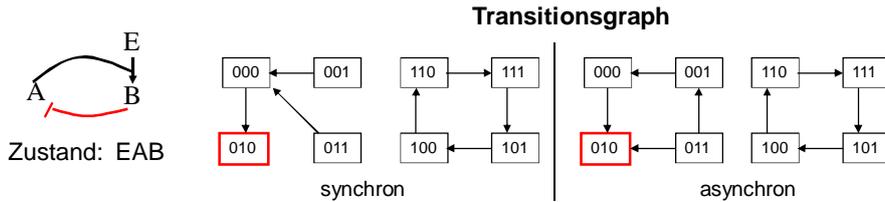
Wahrheitstabelle gibt für alle 2^n möglichen Zustände des Netzes die laut logischen Funktionen resultierenden Bilder an

Synchrones Schalten

- CONTRA: alle Prozesse mit exakt gleicher Dauer (Schaltzeit) → biol. unrealistisch
- PRO: deterministisch

Asynchrones Schalten

- CONTRA: nichtdeterministisch (resultierend in dichteren Transitionsgraphen)
- PRO: berücksichtigt unterschiedliche Dauer der Prozesse (wenn Schaltzeit τ_i für jeden Knoten i bekannt ist, kann diese einbezogen werden)



Attraktoren:

- **Punktattraktoren** (logische steady states): *unabhängig* vom Schaltverhalten
Bsp. oben: 010
- **zyklische Attraktoren** (*abhängig* vom Schaltverhalten)
Bsp. oben: 110-111-101-100 (hier zufällig im synchronen und asynchronen Fall identisch))
- jeder Attraktor ist charakterisiert durch seinen **Einzugsbereich**
Bsp oben: alle 0xx Zustände gehen zu 010; alle 1xx – Zustände sind im zykl. Attraktor)

Diskrete vs. kontinuierliche Dynamik

- diskrete Dynamik wird/wurde vor allem in genregulatorischen Netzen untersucht (Knoten = Gene oder Genprodukte)
- Phänomene des kontinuierlichen Verhalten können zu einem gewissen Grade qualitativ wiedergegeben werden
- Fokus hier auf biologisch relevanterem asynchronem Schalten
- zum Vergleich diskrete/kontinuierliche Dynamik verwenden wir wieder für die kontinuierliche Dynamik (monoton) wachsende bzw. fallende Hill-Funktionen:

Positive Regulation:
$$H^+(x) = \frac{x^n}{\theta^n + x^n} ; n > 1 \quad (n=1 \text{ ist Michaelis-Menten-Kinetik})$$

Negative Regulation:
$$H^-(x) = \frac{\theta^n}{\theta^n + x^n} ; n > 1$$

Für $n > 1$ ergeben sich sigmoide (bzw. gespiegelt sigmoide) Kurven, deren Steilheit mit n steigt. θ ist der Schwellenwert, wir nehmen hier immer an, er sei in einem geeigneten Bereich.

Diskrete vs. kontinuierliche Dynamik

Bsp 1: positiver feedback mit 2 Repressoren:

Kontinuierlich (Differentialgleichungen)

$$x'(t) = k_1 H_1^-(y) - k_x x$$

(k_x und k_y sind Abbauraten)

$$y'(t) = k_2 H_2^-(x) - k_y y$$

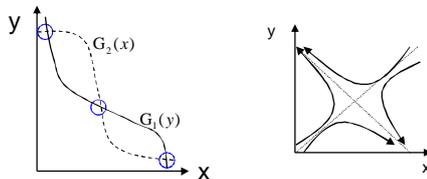
Suche nach steady states (ss):

$$x'(t) = y'(t) = 0$$

$$\rightarrow x = k_1 H_1^-(y) / k_x = G_1(y)$$

$$\rightarrow y = k_2 H_2^-(x) / k_y = G_2(x)$$

($G_1(y)$ und $G_2(x)$ sind Nullklinen mit sigmoidaler Gestalt)



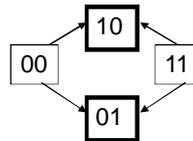
Schnittpunkte der Nullklinen sind die ss; Anzahl der ss hängt von Form (Steilheit) der Hill-Funktionen und 2 stabile ss. Anfangszustände entscheiden, welche der beiden stabilen ss angelaufen wird.

ab: minimal: 1
maximal (wie hier im Bild): 3

Diskret (Logische Beschreibung)

Logische Gleichungen: $X = !Y$
 $Y = !X$

Transitionsgraph der Zustände XY (asynchron):



Zwei stabile Punktattraktoren: 01 und 10

- Logische Beschreibung gibt korrekt wieder, dass bis zu zwei stabile stationäre Zustände existieren (können).
- Hier wie in der kontinuierlichen Beschreibung zeigt sich wie positiver Feedback Loop zu Multistationarität führt.

Diskrete vs. kontinuierliche Dynamik

Positiver feedback mit 2 Repressoren

reales Beispiel: Bacteriophage λ
(Bakteriophagen = „Bakterienviren“)

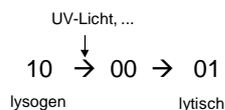


Die Bacteriophage dringt in die Wirtszelle (z.B. *E.coli*) ein und geht entweder in den lysogenen (Einbau der Phagen-DNA in das Bakteriengenom) oder in den lytischen/virulenten (Replikation der Phagen-DNA, Bildung neuer Phagen und Zellyse) Zustand über.

Die Bacteriophage λ hat 2 Schlüsselgene *cI* und *cro*, deren Produkte 2 Repressoren sind, die gegenseitig ihre Gene reprimieren. Die jeweilige Aktivität der beiden Repressoren bestimmt welche der beiden möglichen (steady state) Zustand der Phage vorliegt:

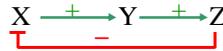
cI on, *cro* off (10): Phage inaktiv (lysogen; Phagen-DNA in Wirtsgenom eingebaut)
cI off, *cro* on (01): Phage im lytischen Zustand (virulent; Phage wird repliziert)

Bestimmte Signale oder Umweltbedingungen (z.B. UV-Bestrahlung) können zum Absinken des *cI*-Repressors führen. Dadurch kann nun das *cro*-Gen expremiert werden, welches nun *cI* reprimiert und den Wechsel zum lytischen Dasein einläutet. Auch wenn die Ursache (z.B. UV-Licht) wieder verschwunden ist (transientes Signal), so hat die Phage einen irreversiblen Switch vollzogen und einen neuen (steady-state) Zustand eingenommen. Entscheidend für dieses Umschaltverhalten ist der positive feedback loop.



Diskrete vs. kontinuierliche Dynamik

Bsp 2: negativer feedback mit 3 Elementen



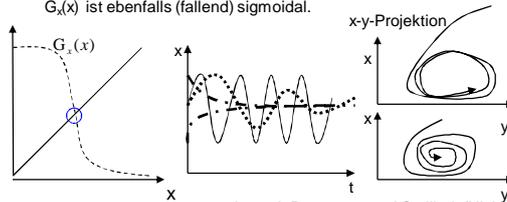
Kontinuierlich (Differentialgleichungen)

$$\begin{aligned} x'(t) &= k_1 H_1^-(z) - k_x x \\ y'(t) &= k_2 H_2^+(x) - k_y y \\ z'(t) &= k_3 H_3^+(y) - k_z z \end{aligned}$$

Steady states (ss):
 $x'(t) = y'(t) = z'(t) = 0$
 $\rightarrow x = k_1 H_1^-(z) / k_x = G_1^-(z)$
 $\rightarrow y = k_2 H_2^+(x) / k_y = G_2^+(x)$
 $\rightarrow z = k_3 H_3^+(y) / k_z = G_3^+(y)$

(k_x und k_y und k_z sind Abbauraten)

$G_1(z)$, $G_2(x)$ und $G_3(y)$ sind Nullklinen mit sigmoidaler Gestalt. Im ss folgt $x = G_1^-(G_2^+(G_3^+(x))) = G_x(x)$ (analog für y und z). $G_x(x)$ ist ebenfalls (fallend) sigmoidal.



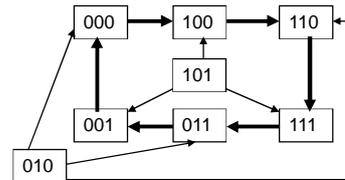
$G_x(x)=x$ hat nur einen Schnittpunkt (das Gleiche ergibt sich für y und z) und somit existiert nur ein ss. Der ss kann stabil oder instabil sein.

Je nach Parameter und Steilheit (Nichtlinearität) der Hill-Funktionen ergibt sich (i) stabiler ss (Knoten/Wirbel-artig), der homöostatisch wirkt; oder (ii) ein instabiler „Sattel-Wirbel“ mit stabilem Grenzzyklus. Dieser stellt oszillatives Verhalten dar.

Diskret (Logische Beschreibung)

Logische Gleichungen:
 $X = !Z \quad Y = X \quad Z = Y$

Transitionsgraph der Zustände XYZ (asynchron):



Kein Punktattraktor. Ein zyklischer Attraktor bestehend aus 6 Zuständen; der Einzugsbereich ist der gesamte Zustandsraum.

- Logische Beschreibung gibt korrekt wieder, dass Oszillationen auftreten können. Homöostase („ss liegt zwischen 0 und 1“) kann nicht abgebildet werden und äußert sich ebenfalls als Zyklus im logischen Modell.

Diskrete vs. kontinuierliche Dynamik

Zusammenfassung

- Logische Modelle können bis zu einem gewissen Grade kontinuierliches Verhalten qualitativ wiedergeben.

Faustregeln:

- Logische Steady States entsprechen (meist) stabilem Gleichgewichtspunkt im kontinuierlichen System.
- Zyklische Attraktoren im logischen Modell können Homöostase oder Oszillationen im kontinuierlichen System bedeuten.
- Oszillationen im kontinuierlichen System haben (meist) komplementären zyklischen Attraktor im logischen Modell.
- Transienter Verlauf im kontinuierlichen System entspricht im diskreten Modell (meist) einem Pfad im Transitionsgraphen.

IV.3.4 Signalfflussanalyse in Logischen Modellen von Signaltransduktionsnetzen (STN)

- Verhalten der STN wird stark durch ihre Inputs bestimmt
- weniger Feedback Loops als in genregulatorischen Netzen; sie haben meist die Rolle des Ausschaltens von Signalen nach erfolgter Weiterleitung
- Idee: bestimme die qualitative Antwort eines STN auf einen externen Stimulus (Input/Output-Verhalten) im logischen Modell durch Berechnung des resultierenden logischen steady states (LSS)



- zur Berechnung des LSS benötigen wir eine Technik, die es uns erlaubt, mit unbekanntem (initialen) Booleschen Zuständen umzugehen:



→ dies ermöglicht die **dreiwertige Logik**

Dreiwertige Logik

- mögliche Zustände:
0 (aus), 1 (an), * (unbekannt)
- die Regeln für UND, ODER und NICHT leiten sich direkt aus den binären Varianten dieser Operatoren ab wenn man berücksichtigt, dass * als Platzhalter für eine mögliche 0 oder 1 steht

A UND* B

	B	0	1	*
A	0	0	0	0
1	0	0	1	*
*	0	*	*	*

Dominiert durch Minimum.
Analog erweiterbar auf n Inputs.

A ODER* B

	B	0	1	*
A	0	0	1	*
1	1	1	1	1
*	*	*	1	*

Dominiert durch Maximum.
Analog erweiterbar auf n Inputs.

NOT* A

A	0	1	*
NOT A	1	0	*

Mit dieser dreiwertigen Logik lässt sich sehr gut die Signalpropagation in einem Signalnetz, wo einige Knoten (typischerweise Inputs oder/und knock-outs/ins) auf einen Wert festgesetzt sind und der (Start-)Zustand der anderen Knoten zunächst unbekannt ist, simulieren (s.u.)

Logischer Steady State in Zwei/Dreiwertiger Logik

- **gegeben:**
 - Boolesches Netz: $L=(V,F)$ mit n Knoten $v_1 \dots v_n$ (mit Zuständen $\mathbf{x}=(x_1 \dots x_n)$)
 - einige festgesetzte Knotenzustände (typischerweise Inputs/Interventionen)
 - $I=\{i_1, i_2 \dots i_k\} \subset \{1, \dots, n\}$; (Indexmenge der festgesetzten Knoten)
 - $Z=\{x_{i_1}=b_{i_1}; x_{i_2}=b_{i_2}; \dots x_{i_k}=b_{i_k}\}, b_{i_1} \dots b_{i_k} \in \{0, 1\}$ (die festgesetzten Zustände)
- **gesucht:** resultierende Zustände (soweit auflösbar) im Booleschen Netz nach erfolgter Signaltransduktion
- **Ansatz:** überschreibe die Booleschen Funktion der festgesetzten Knoten, also $F_{i_1}=b_{i_1}; F_{i_2}=b_{i_2}; \dots; F_{i_k}=b_{i_k}$, und initialisiere ihre Zustände in \mathbf{x} mit diesen Werten während alle anderen Knoten mit „*“ (unbekannt) in \mathbf{x} initialisiert werden;
 - Bestimme dann den LSS* in dreiwertiger Logik ($\mathbf{x} = \mathbf{F}^*(\mathbf{x})$):

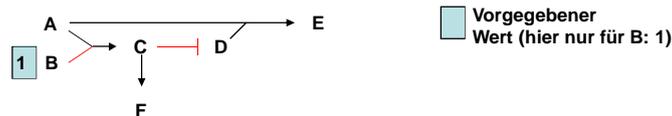
```

while(1) // Endlosschleife
{ suche Knoten u mit  $x_u = *$  und  $F_u^*(\mathbf{x})=0$  oder  $F_u^*(\mathbf{x})=1$ 
  → wenn so ein Knoten k existiert dann  $x_k = F_u^*(\mathbf{x})$ 
  → sonst: break (Abbruch der while Schleife – kein weiterer Knoten auflösbar) }

```

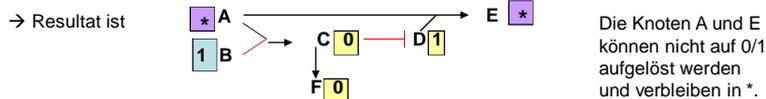
Die Schleife terminiert nach max. n Durchläufen und in \mathbf{x} steht der **eindeutige** LSS* in dreiwertiger Logik; alle 0/1 –Zustände in \mathbf{x} folgen außerdem auch als resultierende Zustände im „normalen“ Booleschen Netz.

- Beispiel (in Hypergraphen-Notation):



$x_B=1$ (und damit $F_B(\mathbf{x})=1$) ist festgesetzt;
alle anderen Zustände werden initialisiert mit $x_A = x_C = x_D = x_E = x_F = *$

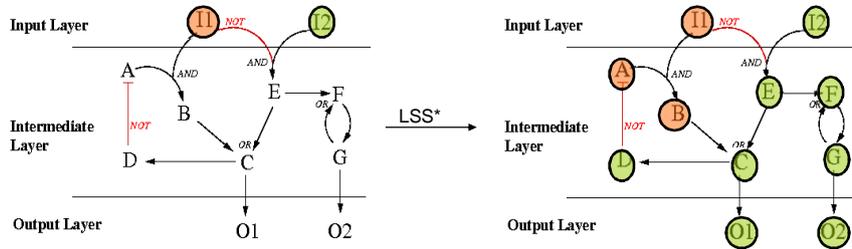
- 1. Iteration: es folgt, dass $F_C(\mathbf{x})=0$ ist → setze $x_C = 0$
- 2. Iteration: es folgt, dass $F_D(\mathbf{x})=1$ ist → setze $x_D = 1$
- 2. Iteration: es folgt, dass $F_E(\mathbf{x})=0$ ist → setze $x_E = 0$
- 3. Iteration: kein neuer Knoten kann von * auf 0 oder 1 umgeschrieben werden → Abbruch



- wenn ein Knoten auf 0 oder 1 gesetzt wird, kann er danach nicht mehr überschrieben werden (das folgt gerade aus der konservativen dreiwertigen Logik)
- die genaue Reihenfolge des Schaltens in der Realität ist ohne bekannte Schaltzeiten nicht immer eindeutig (z.B. hätte oben auch F vor D schalten können); jedoch ist der LSS* immer eindeutig und zeigt die (soweit bestimmbare) qualitative Antwort des Systems für vorgegebene Inputs oder Perturbationen/Interventionen

Beispiele zur Logischen Steady-State- und Signalflussanalyse

Szenario 1: Inputs (fix): I1=0, I2=1



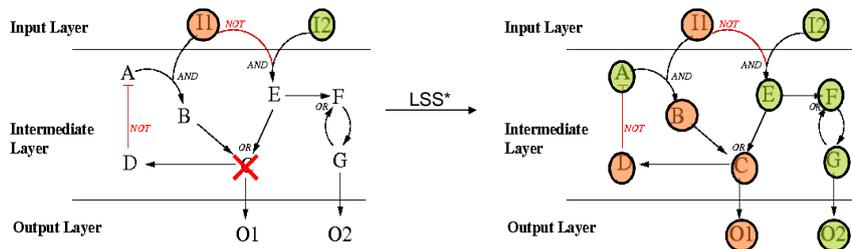
- = 1 = aktiv / vorhanden
- = 0 = inaktiv / nicht vorhanden
- ohne Kreis: undefiniert (*)

- alle resultierenden Zustände im LSS* in dreiwertiger Logik sind 0 oder 1
- diese folgen dann auch für das LSS in "normaler" Boolescher Logik (unabhängig ob asynch oder synch. Schalten!)
- Beachte: der genaue zeitliche Verlauf der Signalpropagation ist i.A. unklar; aber der (qualitative) Endzustand auf das Eingangssignal ist hier eindeutig (und entspricht dem kontinuierlichem Fall)

Logische Steady-State- und Signalflussanalyse

- Effekte von Eingriffen (z.B. knock-outs) auf das Ein/Ausgangsverhalten können getestet werden

Szenario 2: Inputs: I1=0, I2=1 (wie Szenario 1) + knock-out von C (C=0 fix)

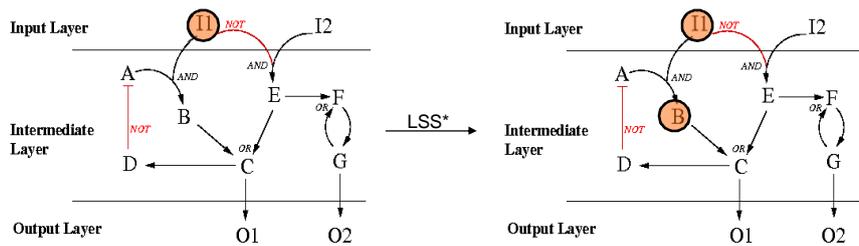


Logische Steady-State- und Signalflussanalyse

- ob tatsächlich alle Knotenzustände eindeutig aus den festgesetzten Knoten gefolgert werden können hängt von den *vorgegebenen Inputs/Interventionen* und der *Aktivität von Feedback Loops* ab;

→ wenn Knoten im LSS* in dreiwertiger Logik den Zustand * haben, kann ihr Zustand im LSS in der binären Logik nicht gefolgert werden

Szenario 3: Inputs: I1=0

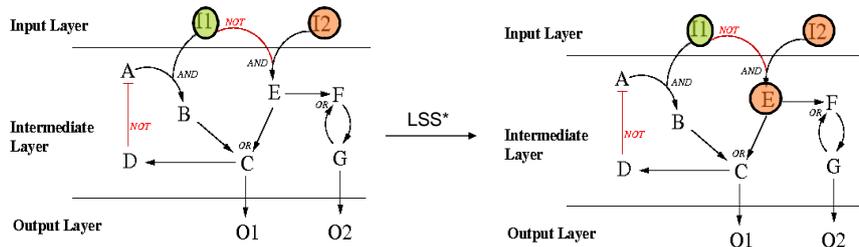


- außer I1=0 und B=0 bleiben alle anderen Zustände auf * (undefiniert)
- nur B=0 kann als (partieller) LSS für das Boolesche Netz gefolgert werden

Logische Steady-State- und Signalflussanalyse

- (aktive) Feedback Loops können Eindeutigkeit bzw. Existenz von logischen steady state in Boolescher Logik verhindern

Szenario 4: Inputs: I1=1, I2=0



→ nur E=0 kann gefolgert werden

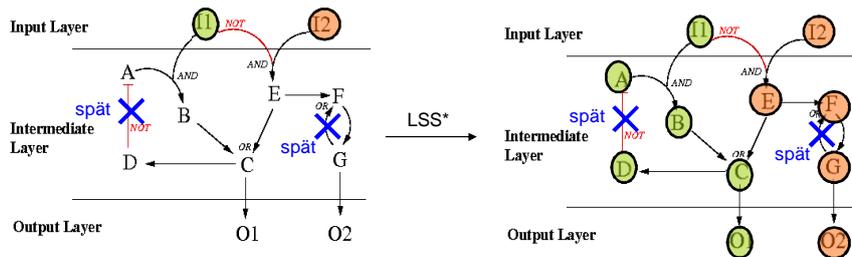
- positive FL: erzeugen Sensitivität bzgl. Anfangszustände (Multistationarität!!)
(z.B. $F_0=G_0=1 \rightarrow F=1$ und $G=1$ im Booleschen LSS)
- negative FL: Boolescher LSS existiert u.U. nicht (zyklischer Attraktor durch $A \rightarrow B \rightarrow C \rightarrow \neg D$)
entspricht im realen homöostatisches oder oszillatives Verhalten

→ Lösung: Unterscheide frühe/späte Ereignisse

Logische Steady-State- und Signalflussanalyse

- betrachtet man nur die initiale Antwort, lassen sich bestimmte Kanten oft als "kausal spätere Ereignisse" deklarieren und können dann für die initiale Antwort unberücksichtigt bleiben; dadurch werden insbesondere die Feedback Loops oft "aufgebrochen"

Szenario 5: Inputs: I1=1, I2=0; Interaktionen G→F und D→A inaktiv (späte Ereignisse)



→ Initiale Systemantwort auf Inputstimuli kann vollständig angegeben werden

→ logische Netzwerke mit azyklischem Interaktionsgraphen haben sowohl in zweiwertiger als auch in dreiwertiger Logik einen (gemeinsamen) eindeutigen logischen steady state mit 0/1 Belegung.

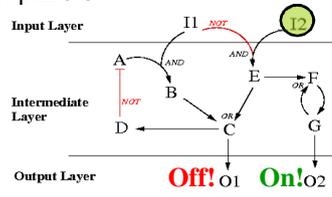
IV.3.5 Minimal Intervention Sets

- Ziel: Finde Eingriffe im logischen Netzwerk, das ein bestimmtes (gewünschtes) Verhalten des logischen Netzes (im steady state) erzwingt
- relevante Beispiele:
 - welche Eingangssignale bringen Zelle zur Apoptose?
 - welche Inhibitoren verhindern Apoptose, welche erzwingen sie?
 - wie kann ich die Aktivierung eines Transkriptionsfaktors verhindern; wie kann ich sie erzwingen?
- Konzept: Minimale Interventionsmengen (Minimal Intervention Sets)

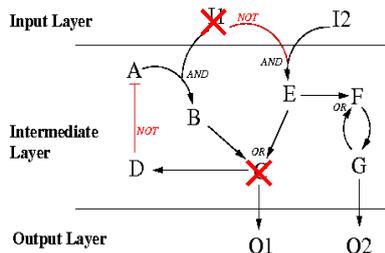
Definition: Minimal Intervention Sets

Minimal Intervention Set (MIS): support-minimale Menge von Eingriffen (konstitutive Aktivierungen + Inaktivierungen/Knock-outs), die ein gewünschtes Verhalten (= *Interventionsziel*) im steady state implizieren

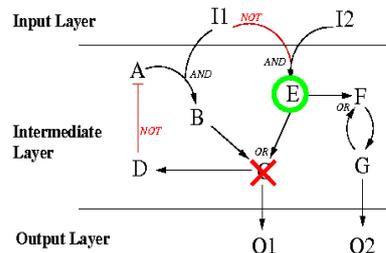
Beispiel: finde Interventionsstrategie, so dass
 - O1 inaktiviert wird und
 - O2 aktiviert wird
 unter der Annahme, dass I2 aktiv ist



MIS1 = { **I1=off, C=off** }



MIS2 = { **C=off, E=on** }



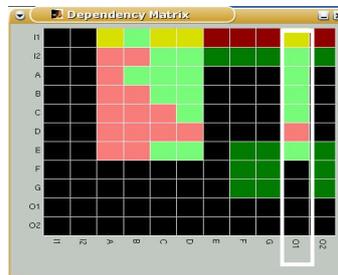
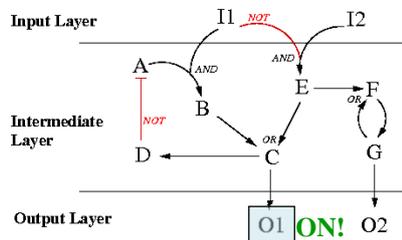
Insgesamt 43 minimale (!) Interventionsmengen

Berechnung der Minimal Intervention Sets

- aufwändig: teste jede Kombination; ob sie zu einem logischen steady state führt, wo Interventionsziel erfüllt ist: falls ja – MIS gefunden; teste keine Supermengen davon
 - hohe kombinatorische Komplexität: bei n Knoten bis zu 3ⁿ Kombinationsmöglichkeiten (im Beispiel: 3¹¹ ~ 177.000!)
- berechne nur MISs *mit geringerer Kardinalität* oder versuche *Suchraum einzuschränken!*

Beispiel: Einschränken des Suchraums durch Ausnutzen von Abhängigkeiten

Interventionsziel: **O1 on!**



- F, G, O2: kein Einfluss auf O1 → kein Eingriff
 - I2, E, C, A, B: reine Aktivatoren von O1 → nur Aktivierung
 - D: reine Inhibitoren von O1 → nur knock-out
 - I1: ambivalenter Faktor von O1 → konst. Aktivierung UND knock-out berücksichtigen
- Nur 10 (von 22) Eingriffen zu kombinieren

Anwendungen von Minimal Intervention Sets

Analog zu Minimal Cut Sets in metabolischen Netzen:

- Targetidentifikation und Interventionsstrategien (Wirkstoffforschung)
- fragile Netzwerkbereiche: kleine MISs sind "gefährlich": wenige Fehler können Verhalten ändern
- Diagnose: bestimme *Fehlermoden*, die (beobachtetes) abnormales Verhalten verursacht haben können

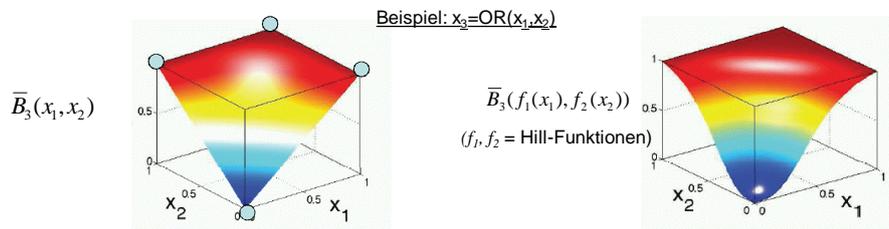
Beispiel: für bestimmten Inputstimulus / beobachtet man im Experiment, dass ein Transkriptionsfaktor T aktiviert vorliegt, obwohl dies nach Modell und Wissen nicht erwartet wird.

→ Man bestimme für Interventionsziel (T ="on") zusammen mit vorgegebenem Stimulus I die resultierenden MISs. Diese stellen die möglichen *Fehlermoden* dar (z.B. permanente Aktivierung/Deaktivierung eines Proteins durch Mutation), die dieses abnormale Verhalten erklären können.

IV.3.6 ODEfy: Von Booleschen zu Differentialgleichungs- (ODE-) Modellen

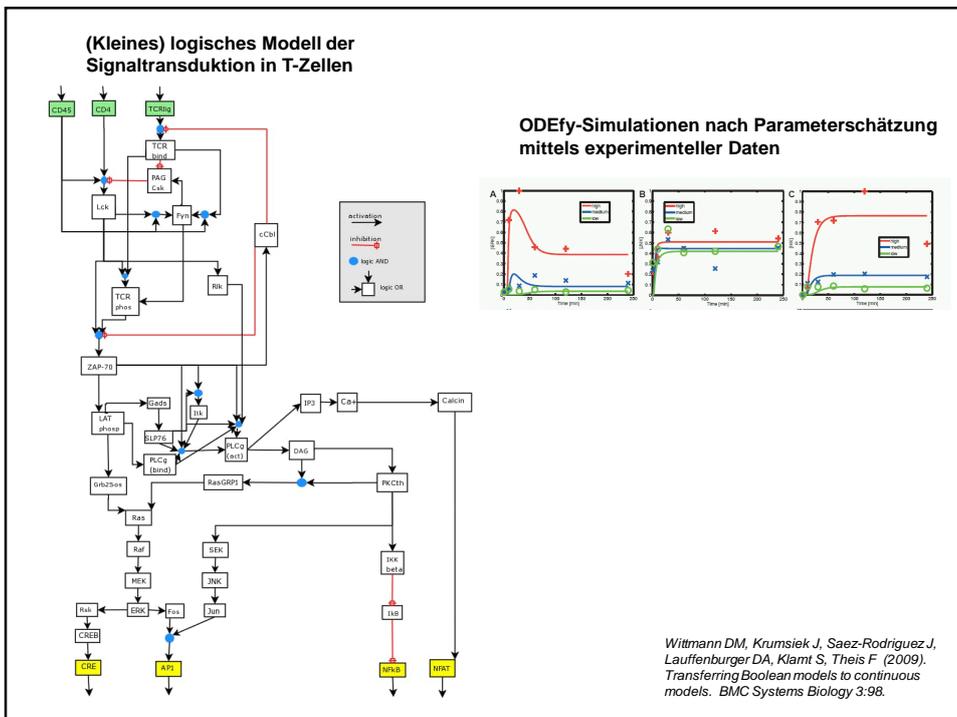
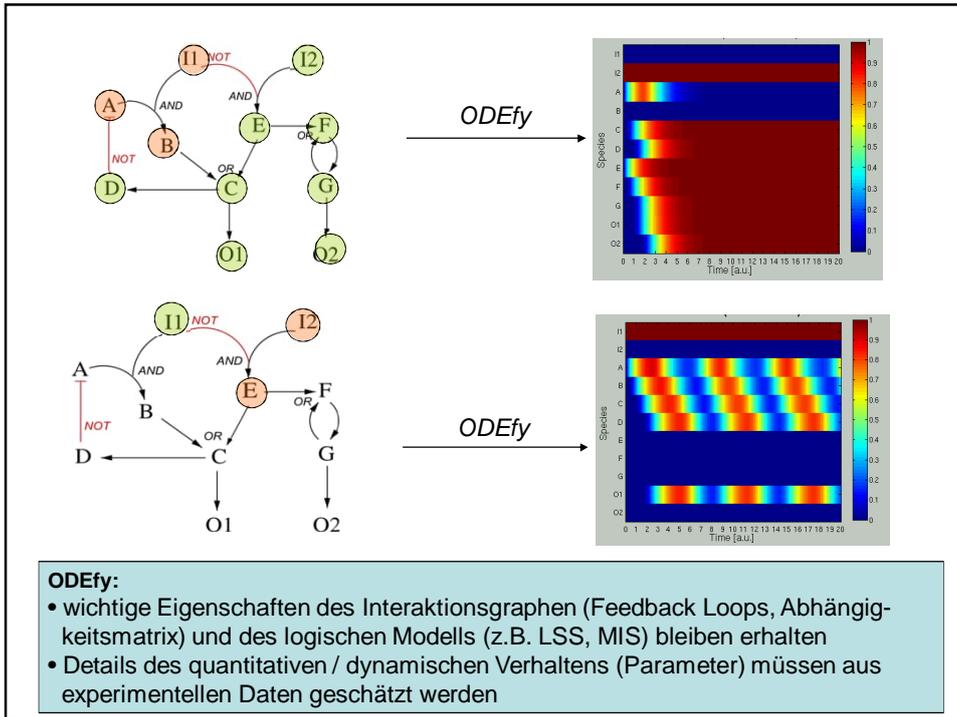
- gegeben: Boolesches Netzwerk mit Booleschen Variablen $x_1, x_2, \dots, x_N \in \{0,1\}$
- Ziel: DGL-Modell mit reelwertigen Variables $\bar{x}_1, \bar{x}_2, \dots, \bar{x}_N \in [0,1]$
- Ansatz: multivariate polynomiale Interpolation von Booleschen Funktionen:

$$\bar{B}^I(\bar{x}_1, \bar{x}_2, \dots, \bar{x}_N) := \sum_{x_1=0}^1 \sum_{x_2=0}^1 \dots \sum_{x_N=0}^1 \left[B(x_1, x_2, \dots, x_N) \prod_{i=1}^N (x_i \bar{x}_i + (1-x_i)(1-\bar{x}_i)) \right]$$

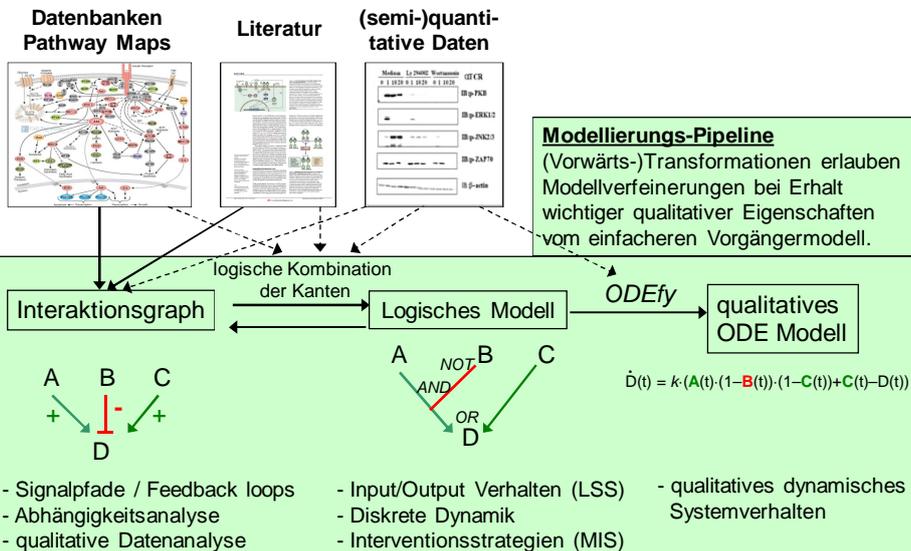


Wittmann DM, Krumsiek J, Saez-Rodriguez J, Lauffenburger DA, Klamt S, Theis F (2009). Transferring Boolean models to continuous models. *BMC Systems Biology* 3:98.

$$\dot{\bar{x}}_i = \frac{1}{\tau_i} (\bar{B}_i(\bar{x}_{i1}, \bar{x}_{i2}, \dots, \bar{x}_{iN_i}) - \bar{x}_i)$$



Zusammenfassung: Qualitative Modellierung von regulatorischen und Signaltransduktionsnetzwerken



IV.3.6 Beispiel 1 einer logischen Netzwerkanalyse: Logisches Modell des T-Zell-Rezeptor Signalnetzwerkes

(bearbeitet zusammen mit Institut für Immunologie (Gruppe B. Schraven) und Institut für mathematische Optimierung (Gruppe Robert Weismantel) an der Universität Magdeburg)

Signaltransduktionsnetze in der T-Zelle (Rahmen: hier untersucht)

- T-Zellen erkennen Fremdkörper und koordinieren Immunantwort

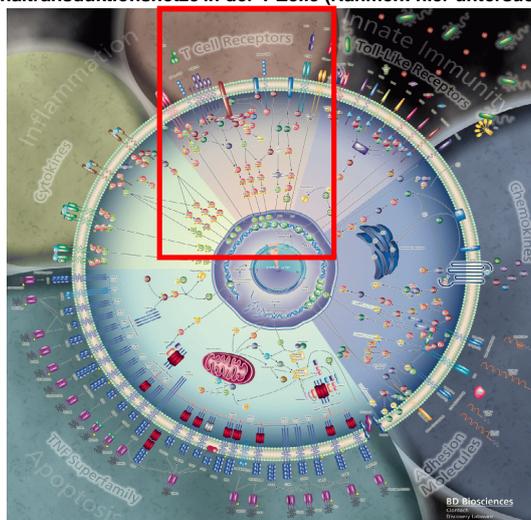
Inputs:

- T-Zell-Rezeptor
- CD4
- CD28

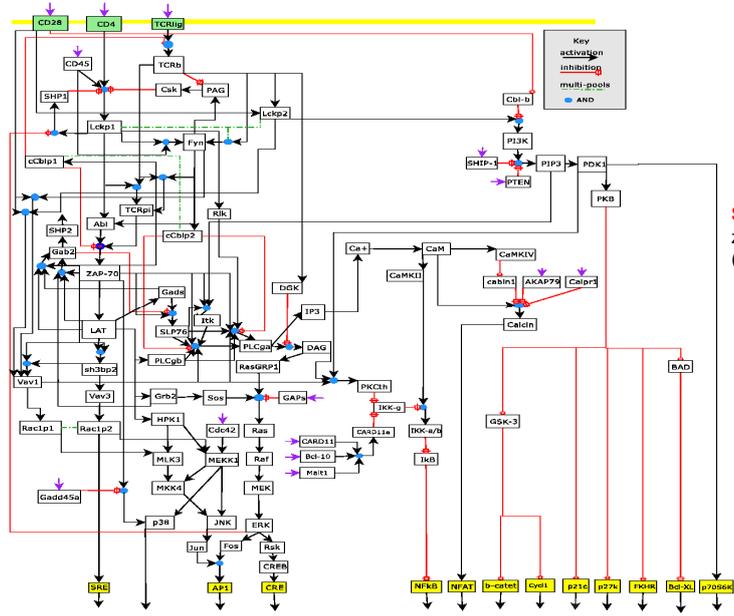
- Outputs sind verschiedene Transkriptionsfaktoren, die für die Aktivierung der T-Zelle (Proliferation) wichtig sind

- Fokus: „frühe Ereignisse“

Publikation: Saez-Rodriguez et al., 2007, *PLoS Computational Biology*, e3:163 (freier download)



Logisches Netzwerk der durch den T-Zell-Rezeptor induzierten Signaltransduktion (in Hypergraphenform)



123 Interaktionen
94 Knoten

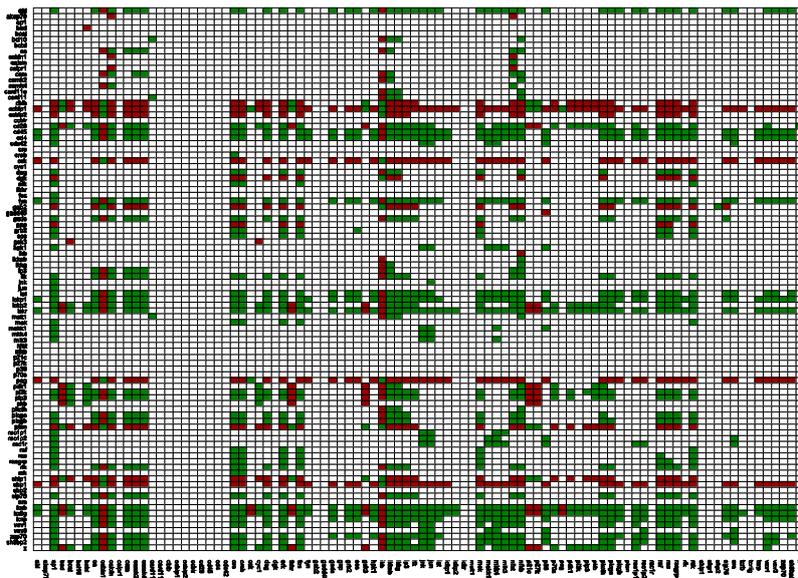
ca. **1,5 Jahre** Rekonstruktion (Literatur, Datenbanken)

Signalwege
z.B. 743 Wege von TCR (input) zu AP1 (output)

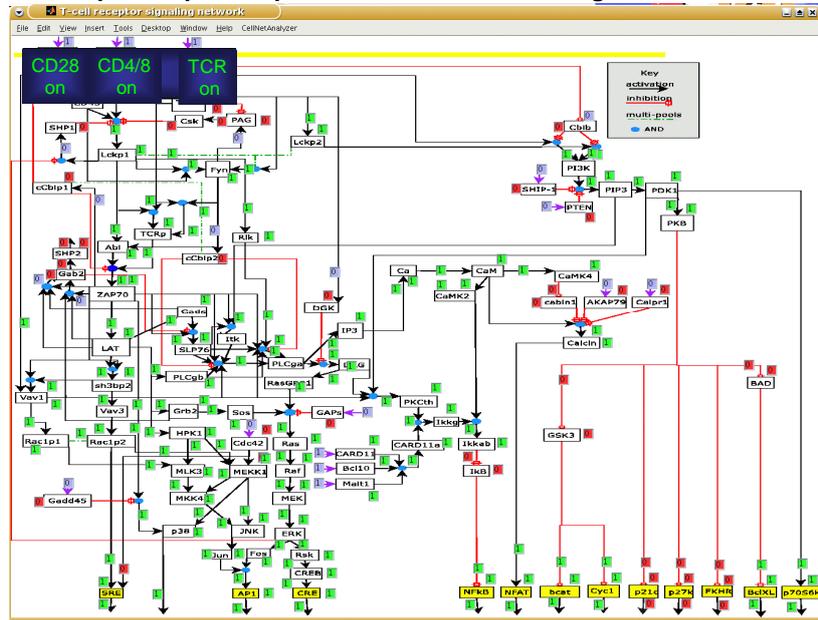
Feedback Loops
87 positive
111 negative

8 "späte" Events
→ "brechen" alle Feedback loops auf

Abhängigkeitsmatrix (für "frühe" Ereignisse)

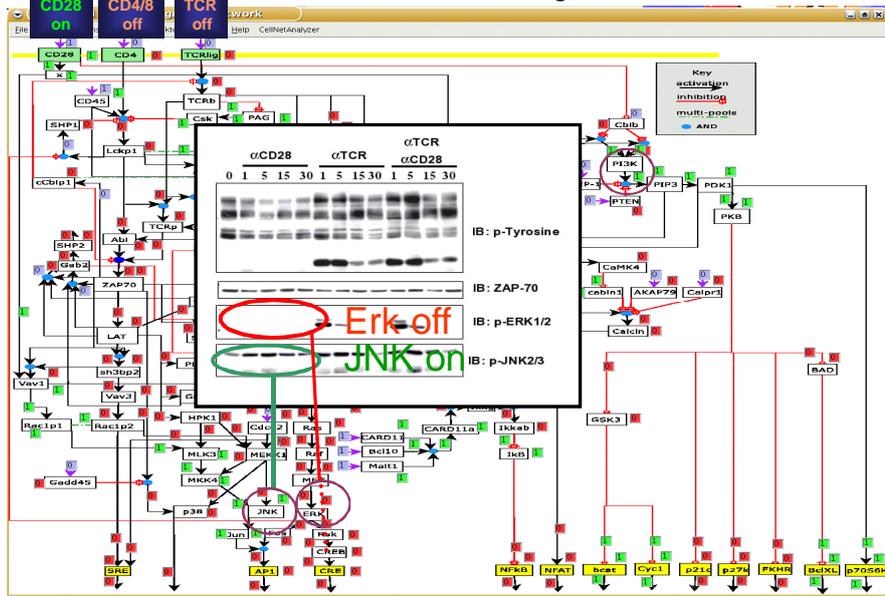


Input/Output-Response: "Standardaktivierung" von T-Zellen

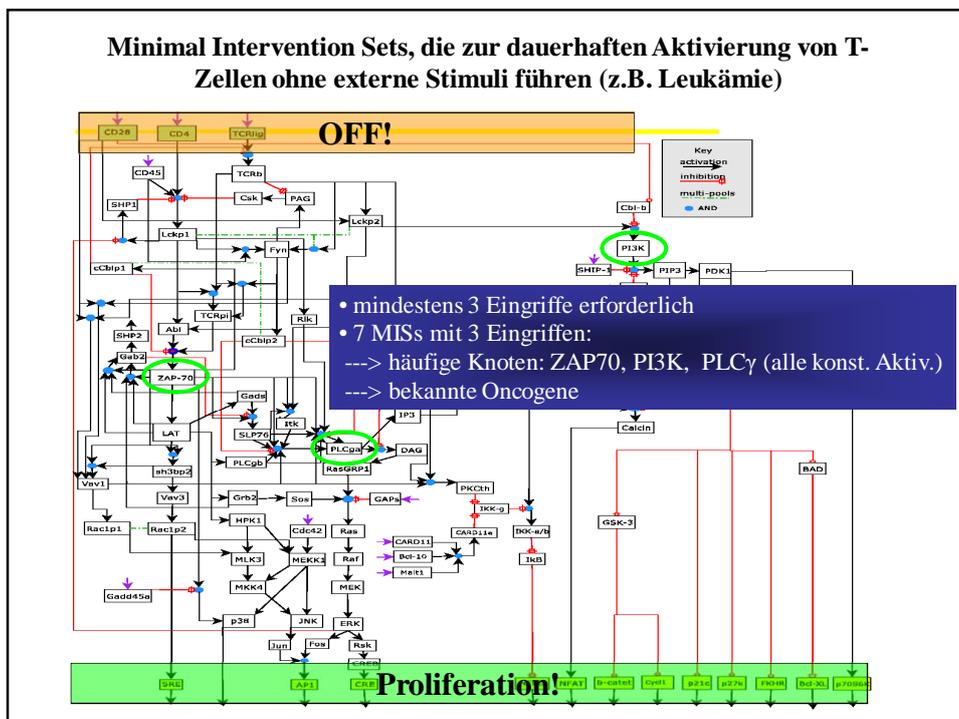
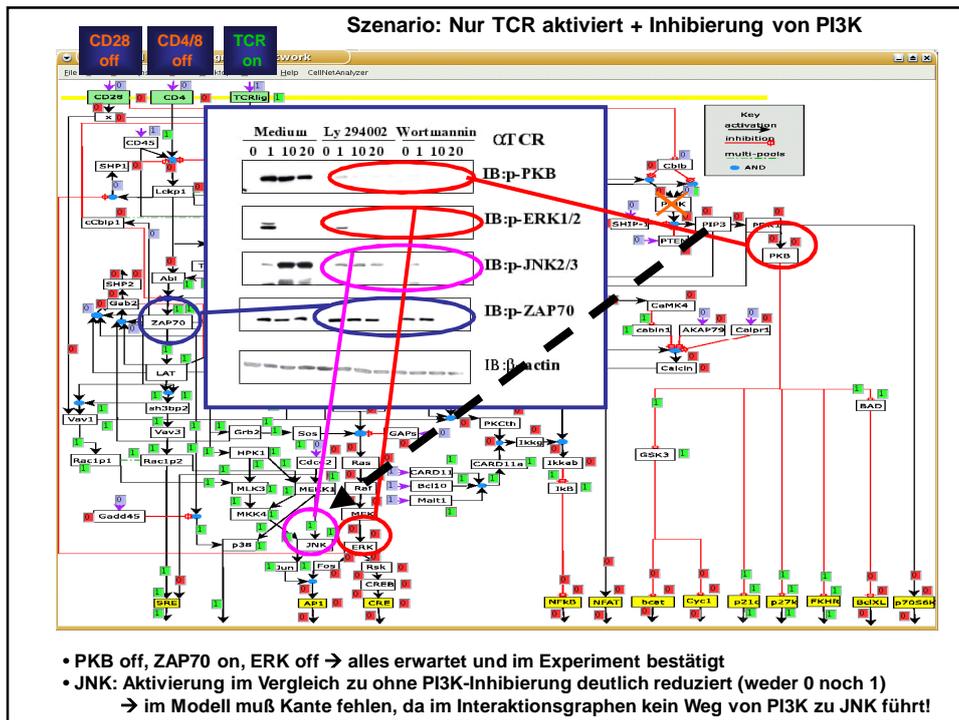


Resultat: Simulation gibt erwartetes qualitatives Verhalten wieder!

Szenario: Nur CD28-Aktivierung, CD4/8 + TCR aus

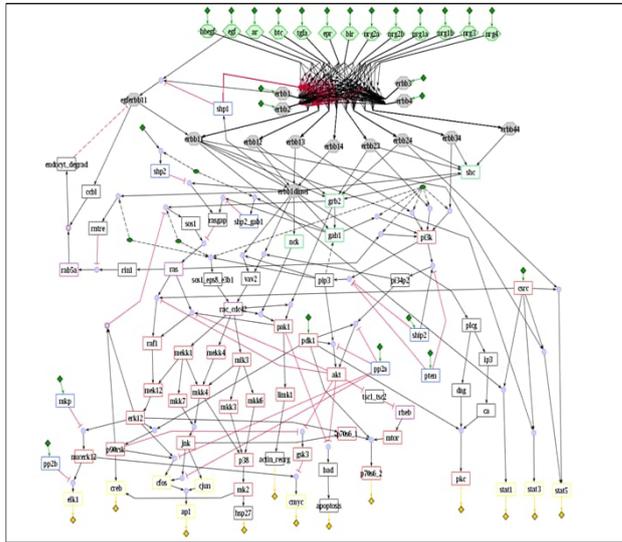


Ergebnis Simulation: ERK (MAP-Kinase) inaktiv und PI3K-Pathway aktiv (wie erwartet)
 aber: JNK (MAP-Kinase) aktiv??? → unerwartet
 Experiment bestätigt Prädiktion!! (ERK unphosphoryliert, JNK phosphoryliert)



IV.3.7 Beispiel 2 einer logischen Netzwerkanalyse: Logisches Modell des EGFR/ErbB-Rezeptor Signalnetzwerkes (EGF = Epidermal Growth Factor; Wachstumsfaktor)

- bekanntester Signalweg in Säugerzellen
- wichtig z.B. für Proliferation
- Fehler im Signalnetz z.B. bei Brustkrebs beteiligt
- Logisches Modell erstellt aus Literaturwissen:
 - 104 Knoten
 - 204 Interaktionen (Hyperkanten)
- 13 Liganden (Inputs), Outputs sind hauptsächlich Transkriptionsfaktoren
- Fokus: „frühe Ereignisse“ (Feedback Loops aus)
- hier gezeigte Resultate: Analyse des Interaktionsgraphen mit High-Throughput-Daten

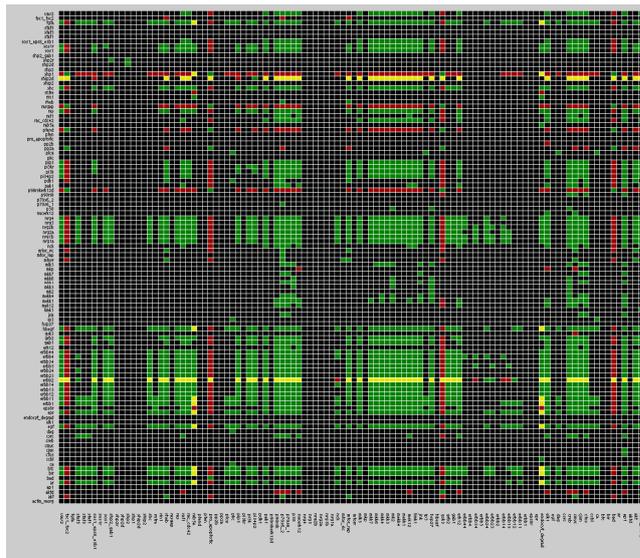


Publikation: Samaga et al., 2009, *PLoS Computational Biology*. (Freier Download)

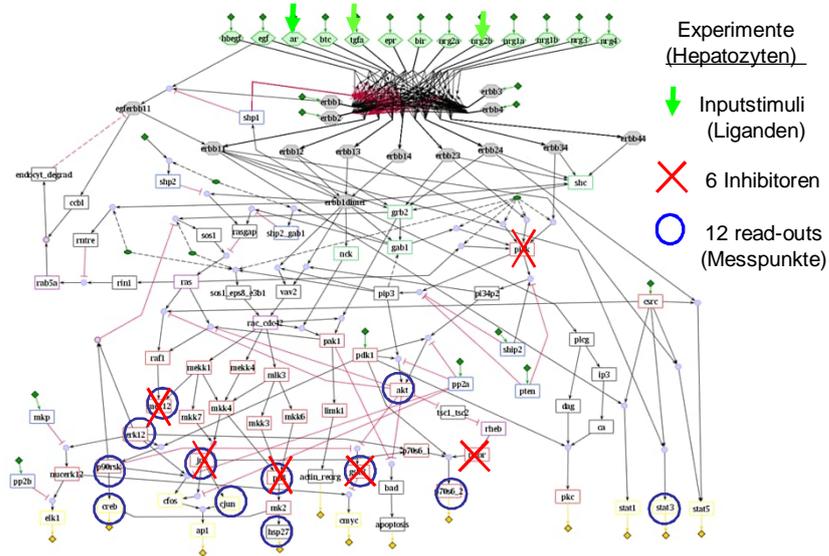
Abhängigkeitsmatrix für frühe Ereignisse

- strong activator
- weak activator
- strong inhibitor
- weak inhibitor
- ambivalent factor
- neutral factor

Tausende qualitative Prädiktionen
→ Experimentelle Verifikation ...?!



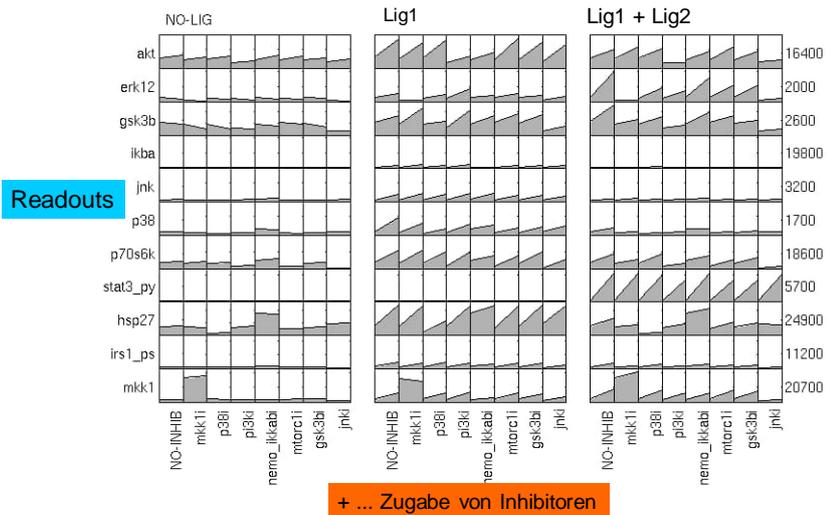
Perturbationsexperimente mit Phosphoproteomischen Daten (mit Peter Sorger, Julio Saez-Rodriguez: Harvard Medical School)



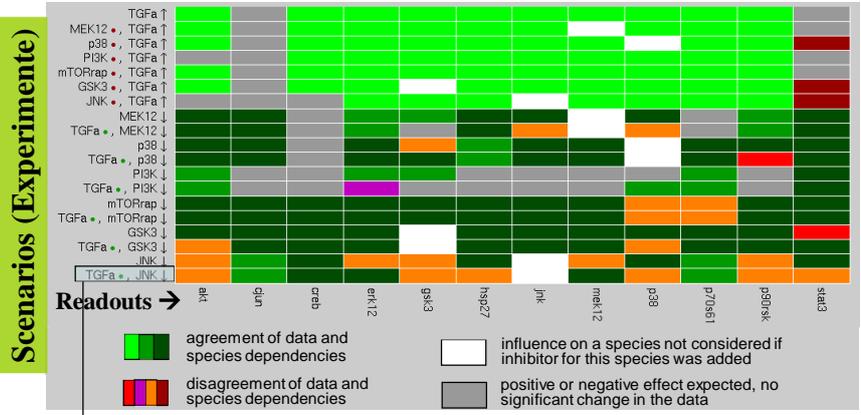
Luminex-basierte Phosphoproteomische Daten

(Peter Sorger, Leonidas Alexopoulos)

Stimulation mit (Kombinationen von) Wachstumsfaktoren/Hormonen



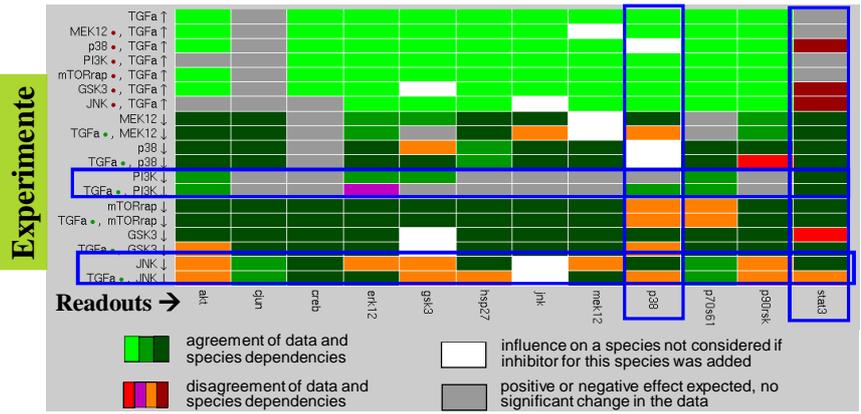
Vergleich: Prädizierte und gemessene „Aufs“ und „Abs“ im EGFR/ErbB Netzwerk:



Jeweils Vergleich zweier Szenarios: (i) mit JNK inhibitor und TGFalpha / (ii) nur TGFalpha (beschreibt dann den Effekt von JNK inhibitor)

Verglichen werden die erwarteten Aufs und Abs (aus Abhängigkeitsmatrix) mit den experimentell bestimmten

Vergleich: Prädizierte und gemessene „Aufs“ und „Abs“ im EGFR/ErbB Netzwerk:



Diskrepanzen zwischen Modell und Experiment (siehe blaue Boxen) führen zu neuen Hypothesen:

- geringe Sensitivität gegen PI3K Inhibitor: Rac Aktivierung unabhängig von PI3K!
- JNK Inhibitor ist offensichtlich unspezifisch!
- STAT3 Aktivierung via ErbB receptors findet in Hepatozyten scheinbar nicht statt ?
- fehlende Kante für p38 activation?

Referenzen

Signaltransduktion in Zellen (Biologie)

Schlessinger J: Cell signaling by receptor tyrosine kinases. *Cell* 2000, 103:211-25.
Downward J. The ins and outs of signalling. *Nature*, 2001, 411:759-762.

Qualitative Modellierung von Signaltransduktion (allgemein)

Papin JA, Hunter T, Palsson BO, Subramaniam S. Reconstruction of cellular signalling networks and analysis of their properties. *Nat Rev Mol Cell Biol*, 2005, 6:99-111.

H. de Jong, D. Ropers. Qualitative approaches towards the analysis of genetic regulatory networks, Z. Szallasi, V. Periwal, J. Stelling (eds), *System Modeling in Cellular Biology: From Concepts to Nuts and Bolts*, MIT Press, 2006, Cambridge, MA, 125-148.

Klamt S, Saez-Rodriguez, Lindquist J, Simeoni L and Gilles ED. A methodology for the structural and functional analysis of signaling and regulatory networks. *BMC Bioinformatics*, 2006, 7:56. <freely available at BMC Bioinformatics>

Logische Modellierung

Thomas R. and D'Ari R. *Biological Feedback*. CRC Press, Boca Raton, 1990.

Thomas R, Kaufman M: Multistationarity, the basis of cell differentiation and memory. I Structural conditions of multistationarity and other non-trivial behavior. *Chaos*, 2001, 11:170-179.

Albert R, Othmer HG: The topology of the regulatory interactions predicts the expression pattern of the Drosophila segment polarity genes. *J. Theor. Biology*, 2003, 223: 1-18.

Saez-Rodriguez J, Simeoni L, Lindquist JA, Hemenway R, Bommhardt U, Arndt B, Haus UU, Weismantel R, Gilles ED, Klamt S and Schraven B. A logical model provides insights into T cell receptor signaling. *PLoS Comp Biol* 2007, 3: e163.

Samaga R, Saez-Rodriguez J, Alexopoulos L, Sorger PK, Klamt S (2009) The logic of EGFR/ErbB signaling: Theoretical properties and analysis of high-throughput data. *PLoS Computational Biology*, 5(8):e1000438.

Samaga R, Klamt S (2013) Modeling approaches for qualitative and semi-quantitative analysis of cellular signaling networks. *Cell Communication and Signaling* 11:43.

Referenzen (2)

Analyse der qualitative Dynamik eines DGL-Systems auf der Basis von Rückkopplungsschleifen und der Netzwerkstruktur

Lipson, M. Differential and graphical approaches to multistability in chemical reaction networks. Undergraduate thesis, Harvard College. <http://www.hcs.harvard.edu/thesis/repo/34/>

Thomas R, Kaufman M: Multistationarity, the basis of cell differentiation and memory. I Structural conditions of multistationarity and other non-trivial behavior. *Chaos*, 2001, 11:170-179.

* Thieffry D. Dynamical roles of biological regulatory circuits. *Briefings in Bioinformatics*, 2007, 8:220-5.

* Gouze JL. Positive and negative circuits in dynamical systems. *J. Biol. Systems*, 1998, 6:11-15

Soule C. Graphical Requirements for multistationarity. *ComplexUs* 2003, 1:123-130.

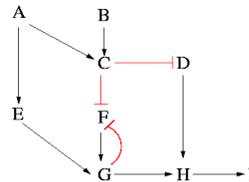
* Craciun G, Tang, Y, Feinberg M. Understanding bistability in complex enzyme-driven reaction networks. *PNAS*, 2006, 103: 8697-8702.

Feinberg, M. (1987). "Chemical reaction network structure and the stability of complex isothermal reactors: I. The deficiency zero and deficiency one theorems". *Chemical Engineering Science* 42 (10): 2229–2268

Übungen

Betrachten Sie nebenstehenden Interaktionsgraphen IGÜ1

- 1) Wieviele Feedback Loops hat der Graph und welche Signalwege führen von Input- zu Outputknoten?
- 2) Stellen Sie die Abhängigkeitsmatrix für IGÜ1 auf (mit Klassifizierung der paarweisen Abhängigkeiten nach schwachen/starken Aktivatoren und Inhibitoren, ambivalenten Faktoren und ohne Einfluss)
- 3) Betrachtet werden Perturbationsexperimente (= permanente Erhöhung der Konzentration bzw. des Aktivitätslevels eines Knotens aus einem steady state heraus) in einem System mit Interaktionsgraphen IGÜ1. Welche der folgenden Aussagen sind falsch?
 - a) Perturbation in D führt zu Anstieg in G.
 - b) Perturbation in A führt zu Anstieg in H.
 - c) Perturbation in B führt in initialer Response zum Absinken von D, danach steigt D an.
 - d) Perturbation in B führt in initialer Response zum Absinken von H, danach steigt H an.
 - e) Das System zeigt bistabiles Verhalten.



Übungen

- 4) Welche Minimalen Cut Sets (von Species) unterbrechen die Übertragung negativer Signale von A nach I in IGÜ1? Welche sind davon nützlich, wenn man die Übertragung positiver Signale von A nach I nicht unterbrechen will?
- 5) Stellen Sie die Differentialgleichungen, die Jacobi-Matrix und den Interaktionsgraphen der Jacobi-Matrix auf für folgendes Reaktionsnetzwerk mit Massenwirkungskinetik: R1 (irreversibel): $A + B \rightarrow 2 C + D$ und R2 (reversibel): $2 C \leftrightarrow B$
- 6) a) Öffnen Sie im CellNetAnalyzer das Netzwerk „T-cell model (Small)“
Machen Sie sich im Menü mit den folgenden Funktionen für IGs vertraut:
 - * *Convert to interaction graph*
 - * *Compute Strongly connected components*
 - * *Shortest Path and Species Dependencies*
 - * *Signaling Paths and Feedback Loops*
 - * *Minimal cut sets of paths ...*
- b) Wieviele Pfade von CD4 zu den Output-Knoten gibt es? Wieviele sind davon positiv/negativ? Berechnen Sie die Abhängigkeits-Matrix und diskutieren Sie die Ergebnisse der Pfade im Zusammenhang mit dieser Matrix!
- c) Wieviele (species) MCSs gibt es, die alle Pfade von CD4 zu NFkB und NFAT unterbrechen und dabei mind. einen Pfad von CD4 zu AP1 intakt lassen?
- d) Gibt es eine Spezies, die in allen feedback loops enthalten ist?

Übungen

- 7) Zeichnen Sie für das folgende Inzidenz/NOT-Matrix-Paar den (logischen) Hypergraphen des mit diesen beiden Matrizen gespeicherten Booleschen Netzes:

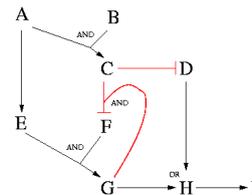
$$\mathbf{I} = \begin{pmatrix} -1 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 \\ 0 & -1 & -1 & 0 & 0 \\ 1 & -1 & 0 & 0 & 1 \\ 0 & 0 & 1 & -1 & 0 \\ 0 & 1 & 0 & 1 & -1 \end{pmatrix} \begin{matrix} A \\ B \\ C \\ D \\ E \\ F \end{matrix}, \quad \mathbf{NOT} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{matrix} A \\ B \\ C \\ D \\ E \\ F \end{matrix}$$

Geben Sie die Booleschen Funktionen für alle Knoten explizit an!

- 8) a) Nehmen Sie an, dass $A=1, B=1, C=0$ festgesetzt wird. Schreiben sie für diese Konstellation den resultierenden Transitiongraphen des Booleschen Netzes auf (jeweils für asynchrone und synchrone Schaltung).
Gibt es einen/mehrere Attraktor(en)?
- b) Machen Sie das Gleiche für den Fall wo $A=0, B=1, C=0, E=1$ fixiert sind!

Übungen

Betrachten sie nebenstehendes Boolesches Netz BNÜ1 welches in Hypergraphenform dargestellt ist.



- 9) Stellen Sie die Inzidenz- und NOT-Matrix des Netzwerkes auf. Wie sieht der dem Booleschen Netz unterliegende Interaktionsgraph aus?
- 10) Bestimmen Sie für die folgenden Szenarien die logischen steady states (in dreiwertiger Logik) und resultierenden Signalflüsse
- a) $A=0$
 - b) $A=0, B=1$
 - c) $A=1$
 - d) $A=1, B=0$
 - e) $A=1, B=0$ + Knock-out in F
 - f) $A=0, B=0$
 - g) $A=1, B=1$
 - h) $A=1, B=1$ + knock-in in G
- 11) Bestimmen Sie für die folgenden Interventionsziele in BNÜ1 die minimalen Interventionsmengen:
- a) $I=1$
 - b) $I=0$
 - c) $I=1, D=1$
 - d) $G=1$

Übungen

12) Implementieren Sie das Netzwerk von Aufgabe 9 im CellNetAnalyzer mit Hilfe des Network Composers!

Machen Sie sich dann im Menü mit den folgenden Funktionen zur Analyse von logischen Netzen vertraut:

- * *Compute Logical Steady States*
- * *(Logical) Minimal Intervention Sets*
- * *Compute Species Equivalence Classes*
- * *Convert to Interaction Graph*

Überprüfen Sie damit auch Ihre Ergebnisse von Aufgabe 9 – 11.

Übungen

13) Starten Sie im CellNetAnalyzer das Projekt „T-cell model large“

- a) *setzen Sie das default scenario + input bei CD28 auf 1 + time scale auf 1*
→ *bestimmen Sie den logischen steady state*
→ *welchen Zustand haben die MAP-kinasen ERK und JNK?*
- b) *Bestimmen Sie die Minimalen Interventionsmengen mit jeweils max. 2 Eingriffe für eine dauerhafte Aktivierung von NFkB. Als „Umgebungsbedingung“ nehmen Sie das Standardszenario.*
→ *wieviele einfache und wieviele Doppeleingriffe werden gefunden?*
→ *taucht CD4 als eines der Targets auf? Und CD28?*
- c) *Berechnen Sie die dependency matrix für die primären events, d.h. setzen Sie vorher Scenarios.../Set time scale auf 1 und benutzen Sie beim Berechnen der dep. Mat. die Option „Exclude reactions with given zero value“.*
→ *welchen Einfluss hat PI3K jeweils auf AP1, LAT, bad?*
- d) *Konvertieren Sie das logische Modell in einen Interaktionsgraphen („Convert to Interaction graph“)*
→ *wieviele feedback loops gibt es; wieviele sind negativ?*
→ *welcher Knoten ist am stärksten in den negativen feedbacks involviert?*
→ *wieviele Knoten muss man mindestens entfernen um ALLE Kreise aufzubrechen?*