

# Beispiele und künstliche Evolution oszillierender biochemischer Netzwerke: eine experimentelle Studie

Elisabeth Hellwig, Thomas Hinze, Thorsten Lenser, Peter Dittrich

26. März 2008

## Inhaltsverzeichnis

<b>1</b>	<b>Evolution von Signalnetzwerken</b>	<b>2</b>
1.1	Das ESIGNET-Projekt . . . . .	2
<b>2</b>	<b>Oszillierende Prozesse</b>	<b>3</b>
2.1	Ein circadianer Oszillator . . . . .	3
2.2	Ein Beispiel aus der Populationsdynamik . . . . .	5
<b>3</b>	<b>Testläufe mit dem SBMLEvolver</b>	<b>6</b>
3.1	Einfache Sinus-Schwingung . . . . .	6
3.2	Evolution nach einem target-csn: biochemisches Netz mit Oszillation . . . . .	8
3.2.1	Abbildungen . . . . .	8
3.3	Das Lotka-Volterra-Modell . . . . .	10
3.3.1	Das target-csn . . . . .	10
3.3.2	Die Versuche . . . . .	10
<b>4</b>	<b>In silico Evolution funktioneller Einheiten in biochemischen Netzwerken</b>	<b>12</b>
4.1	Evolution von oszillierenden Netzwerken . . . . .	12
4.2	Ergebnisse . . . . .	13
<b>5</b>	<b>Diskussion der Ergebnisse</b>	<b>16</b>

# 1 Evolution von Signalnetzwerken

Biochemische Zellsignalnetzwerke (cell signaling networks, CSNs) zu verstehen - wie sie aufgebaut sind, wie sie arbeiten und wie man sie beeinflussen kann - ist eine wichtige Herausforderung. In der Medizin kann man dieses Wissen verwenden, um Krankheiten zu bekämpfen, welche sich als Fehlfunktionen der natürlichen CSNs manifestieren. Aber damit lassen sich auch völlig neue, künstliche CSNs, wie zB. Nanomaschinen oder Molekulare Computer, schaffen.

Alle diese Anwendungsmöglichkeiten erfordern ein detailliertes Verständnis der Funktionsweise von Signalnetzwerken. Das heißt vor allem auch, aus welchen Teilnetzen bzw. Modulen sie aufgebaut sind, welche Fähigkeiten diese haben und wie die Schnittstellen zwischen ihnen beschaffen sind.

## 1.1 Das ESIGNET-Projekt

An dem EU-Projekt "Evolving Cell Signaling Networks in Silico"[3] arbeiten Forschungsgruppen an Universitäten in Birmingham (UK), Eindhoven (Niederlande), Dublin (Irland) und Jena (Deutschland) mit. Sie haben es sich zur Aufgabe gesetzt, CSNs mit Hilfe von evolutionären Algorithmen zu entwickeln, um mehr über geeignete Konstruktionsprinzipien und Anwendungsszenarien zu lernen.

Da man herausfinden möchte, welche Struktur ein CSN haben muss, um eine bestimmte Aufgabe zu bewältigen, wird ein evolutionärer Algorithmus verwendet: Das Problem und seine gewünschte Lösung werden vorgegeben, und die Netzwerkstruktur, die in der Lage ist, das Problem zu lösen, entwickelt sich per Evolution in silico. Mit dieser Methode soll auf der einen Seite die Möglichkeit geschaffen werden, mit gegebenen Stoffkonzentrationen der Zelle deren biochemische CSNs zu ermitteln. Auf der anderen Seite will man aber auch durch die Analyse der in silico evolvierten Netze etwas über den Zusammenhang zwischen Netzstruktur und Netzfähigkeit lernen, um dann gezielter neue, künstliche CSNs entwerfen zu können.

Im Rahmen des ESIGNET-Projektes werden vier grundlegende Ziele bearbeitet:

1. Es soll ein Verständnis für die Berechnungsfähigkeiten von CSNs entwickelt werden: Wie sehen Netzwerke aus, welche bestimmte (einfache) Aufgaben lösen? Gibt es strukturell verschiedene CSNs, die dieselben Berechnungen ausführen? Für welche Arten von Problemen eignen sich CSNs? Wie robust sind CSNs gegenüber dem Ausfall von Komponenten, Konzentrationsschwankungen oder verändertem Input? Methoden zum Bestimmen dieser Berechnungseigenschaften künstlicher und natürlicher CSN sollen entwickelt werden.
2. Bestehende Verfahren, also evolutionäre Algorithmen, zur Simulation bzw. Modellierung realer biochemischer Netzwerke sollen verbessert werden. Dabei werden 'echte', wie auch durch Modellvorstellungen angegliche, Evolution biologischer CSNs einbezogen. Dies schließt auch die Beantwortung der Frage ein, ob und für welche Aufgaben sie 'optimiert' sind.
3. Noch unbekannte Komponenten natürlicher Netzwerke sollen ermittelt werden.
4. Ein open source software package soll entwickelt werden, mit dem man CSNs evolvieren, simulieren und Komponenten vorhersagen kann.

Aus Arbeiten der Nachwuchsgruppe Biosystemanalyse an der Friedrich Schiller Universität Jena ist im Rahmen des ESIGNET-Projektes die Software SBMLEvolver[7] hervorgegangen, die eine künstliche Evolution von CSNs mit gewünschten Eigenschaften ermöglicht.

## 2 Oszillierende Prozesse

Die Fähigkeit biochemischer Netzwerke zur Oszillation spielt in der Natur eine große Rolle. Durch oszillierende Stoffkonzentrationen in der Zelle werden viele Prozesse angetrieben. So ist zum Beispiel die oszillierende  $Ca^{2+}$  Konzentration der Taktgeber für den Herzschlag, und auch die Reizweiterleitung in den Nervenzellen wird durch oszillierende Ionenkonzentrationen ermöglicht. Aber auch für langsamere Prozesse sind Oszillationen in der Zelle verantwortlich. Viele biologische, zB. circadiane, Rhythmen werden durch eine innere Uhr - ein oszillierendes Reaktionsnetz in der Zelle - am Laufen gehalten. Darunter fallen zB. die Änderung von Blutdruck, Körpertemperatur oder der Hormonzusammensetzung im Blut bei Wirbeltieren bzw. das tägliche Öffnen und Schließen der Blüten bei Pflanzen. Will man diese Prozesse am Computer analysieren, muss das simulierte CSN zu Oszillationen in der Lage sein.

Aber auch andere Prozesse aus unserer Umwelt lassen sich nur durch oszillationsfähige Modelle simulieren. Beispiele sind das periodische Wiederkehren von Epidemien oder ökonomischen Krisen, zahlreiche physikalische Phänomene wie Schwingungen (zB. das Quietschen einer Tür, das Streichen einer Geigensaite), Kreisbewegungen (zB. Exzenter-Rotation) und Wellen (zB. Elektromagnetismus).

In der Technik sind Oszillationen ebenfalls wichtig[2]. Durch elektrische Schaltungen mit Schwingquarzen wird Uhren, Herzschrittmachern und allen Arten von Computern der Takt vorgegeben. In der Signalübertragung verwendet man Oszillatoren als Trägerschwingung zum Aufmodulieren der Signale, also zB. auch für Telefon, Radio und Fernseher. Die Oszillationen werden hier im einfachsten Fall durch einen elektrischen Schwingkreis, bestehend aus Spulen und Kondensatoren, erzeugt. Die in einem Schwingkreis auftretende Dämpfung wird durch Rückkopplung, zB. mittels eines Transistors, aufgehoben. Eine weitere Möglichkeit sind elektrische Kippstufen, welche zB. den Takt für blinkende Signalleuchten vorgeben.

In vielen Fällen wird heutzutage allerdings ein digitaler Oszillator (Multivibrator) verwendet, der wesentlich kostengünstiger und sehr flexibel ist. Dieser diskretisiert die Oszillationen, welche in analogen Schaltungen durch die verwendeten Bauteile erzeugt werden. Digitale Oszillatoren zeichnen sich durch kurze Umschaltzeiten aus, besitzen aber den Nachteil, dass man die erzeugten Frequenzen nicht stufenlos einstellen kann.

### 2.1 Ein circadianer Oszillator

Circadiane Rhythmen[1] werden durch oszillierende biochemische Netzwerke gesteuert. Ein circadianer Rhythmus, der durch eine solche "biologische Uhr" angetrieben wird, muss drei Bedingungen erfüllen:

1. Der Zyklus hat eine ungefähre Periodenlänge von 24 Stunden und behält diese auch unter konstanten Umweltbedingungen, zB. Dunkelheit, bei. Er ist also keine unmittelbare Reaktion auf den Tag/Nacht-Wechsel der Sonne, sondern wird unabhängig von der Außenwelt aufrecht erhalten.
2. Die biologische Uhr kann durch äußere Reize neu eingestellt werden. Zum einen wird der innere Rhythmus, der nur *ungefähr* 24 Stunden entspricht, jeden Tag neu mit dem Auf- und Untergang der Sonne synchronisiert. Aber auch nach starken Verschiebungen (zB. um 12 Stunden) des äußeren Rhythmus hat sich die innere Uhr nach kurzer Zeit wieder angepasst. Das geschieht im einfachsten Fall durch lichtabhängige Reaktionen, zB. Aktivierungen oder Inhibierungen von Enzymen, die Einfluss auf die innere Uhr haben.
3. Die Periodenlänge ist in einem recht großen Toleranzbereich physiologischer Temperaturen stabil. Chemische Reaktionen laufen bei hohen Temperaturen im Allgemeinen schneller ab als bei niedrigen. Um eine ungünstige Veränderung des circadianen Rhythmus bei schwankender Körpertemperatur zu vermeiden, wird die Temperaturänderung kompensiert.

Ein gut untersuchter Modellorganismus mit circadianem Oszillator ist der Schimmelpilz *Neurospora Crassa*, welcher zB. abhängig von der Tageszeit Sporen zur Fortpflanzung produziert[5]. Die Oszillationen entstehen durch negative Feedback-Loops auf Transkriptionsebene der beteiligten Gene.

Hauptbestandteil des Oszillators[9] sind die Gene *frequency* und *vivid*, bzw. deren Genprodukte FRQ und VVD, und der Heterodimer *white collar complex*, WCC (Abb. 1). Die Konzentration von FRQ ändert sich periodisch im Laufe des Tages: Nachts ist sie am niedrigsten, morgens wird die FRQ-Produktion durch den als Transkriptionsfaktor wirkenden WC-Komplex angetrieben und mittags ist sie am höchsten. Im weiteren Verlauf des Tages sinkt sie durch vermehrte Phosphorylierung und Abbau des Proteins wieder.

Der periodische Ablauf entsteht durch Rückkopplung: FRQ und VVD hemmen die Bildung von WCC, während WCC die Bildung von FRQ und VVD aktiviert.

Die Synchronisation mit dem Rhythmus der Außenwelt erfolgt über lichtabhängige Proteine bzw. durch Licht-induzierbare Transkription von Genen (VVD und WC1).

Die *clock controlled genes*, also Gene, welche das circadiane Verhalten von *Neurospora Crassa* bewirken, werden von WCC im Verlaufe des Tages reguliert.

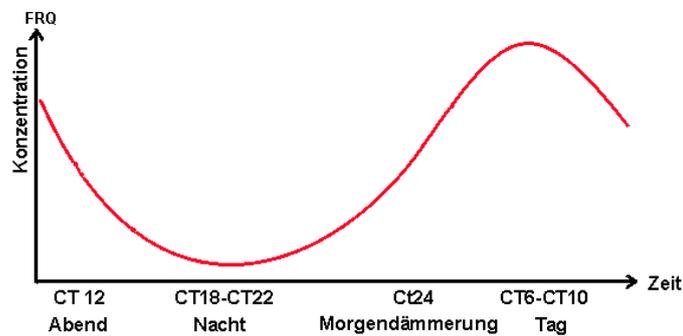
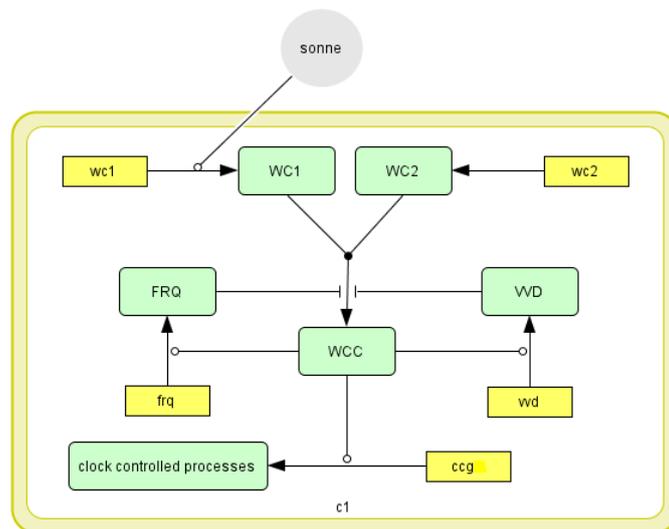


Abbildung 1: Der Oszillator von *Neurospora Crassa*

## 2.2 Ein Beispiel aus der Populationsdynamik

Oszillationen sind auch in makroskopischen Prozessen von Bedeutung, ein Beispiel hierfür sind die schwankenden Populationsdichten einer Räuber- und einer Beutespezies. Das System aus Räuber und seiner Beute kann durch folgende Differentialgleichungen beschrieben werden[6]:

$$\begin{aligned}\dot{x} &= x \cdot r_m \cdot \left(1 - \frac{x}{K}\right) - y \cdot b_m \cdot \frac{x}{x+M} \\ \dot{y} &= y \cdot b_m \cdot \frac{x}{x+M} - d \cdot y\end{aligned}$$

Die Gleichung der Beutespezies  $x$  besteht zum einen aus dem Wachstumsterm  $x \cdot r_m \cdot \left(1 - \frac{x}{K}\right)$  mit der maximalen Wachstumsrate  $r_m$ . Diese Zuwachsrates ist fallend in  $x$ , und für  $x > K$  sogar negativ. Deshalb geht  $x$  ohne Anwesenheit von Räubern gegen die konstante Populationsdichte  $K$ . Der Term  $-y \cdot b_m \cdot \frac{x}{x+M}$  steht für den Tod der Beuteindividuen durch die Räuber, welche eine maximale Fressrate  $b_m$  haben. Diese wird asymptotisch mit wachsendem  $x$  erreicht, entsprechend der Enzym-Sättigungskinetik mit Halbsättigungskonstante  $M$ .

Da man vereinfachend annimmt, dass die Räuberpopulation in dem Maße wächst, wie Beutetiere getötet werden, verwendet man den Term  $-y \cdot b_m \cdot \frac{x}{x+M}$  als Wachstumsmaß der Räuber. Der Ausdruck  $-d \cdot y$  steht schließlich für den Tod der Räuber mit der konstanten Rate  $d$ .

Das System zeigt im Grunde 3 verschiedene Verhaltensweisen. Bei sehr kleinen Werten für  $K$  ist der einzige asymptotisch stabile Fixpunkt  $(K, 0)$ , da die Beutepopulation zu klein ist, um den Räuber zu ernähren, so dass dieser ausstirbt. Wird  $K$  größer als der kritische Wert  $\hat{K}_1$ , existiert ein asymptotisch stabiler Fixpunkt mit  $y > 0$ , an den sich das System für zunehmendes  $K$  mit zunehmend starken, gedämpften Oszillationen annähert. Für  $K$  größer dem kritischen Wert  $\hat{K}_2 > \hat{K}_1$  wird aus dem Fixpunkt schließlich ein asymptotisch stabiler Grenzzyklus, das System oszilliert.

Die Oszillation der Populationsdichten des Modells kann man auch mit Hilfe des CellDesigners<sup>1</sup> simulieren (Abb. 2). Die Beute wird durch die chemische Spezies  $s2$  dargestellt, die Räuber durch  $s3$ . Die kinetischen Gesetze der Reaktionen sind hier:

$$\begin{aligned}v_1 &= s2 \cdot r_m \cdot \left(1 - \frac{s2}{K}\right) \\ v_2 &= s2 \cdot s3 \cdot \frac{b_m}{s2 + M} \\ v_3 &= s3 \cdot d\end{aligned}$$

Die in Abb. 2 dargestellten Oszillationen treten mit den Parametern  $r_m = 0.5$ ,  $K = 9$ ,  $b_m = 1$ ,  $M = 2$ ,  $d = 0.6$  und Ausgangskonzentrationen  $s2 = 6$ ,  $s3 = 2$ ,  $s1 = s4 = const.$  auf.

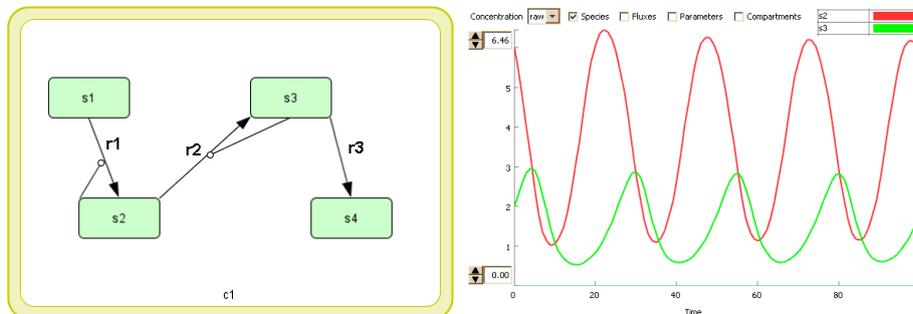


Abbildung 2: Das Räuber-Beute-Modell im CellDesigner

<sup>1</sup><http://www.systems-biology.org/cd/>

### 3 Testläufe mit dem SBMLEvolver

Die Software[7] erzeugt CSNs, welche in Form von SBML-Dateien gespeichert werden, mit Hilfe eines evolutionären Algorithmus, der nach folgendem Schema abläuft, bis die gewünschte Fitness erreicht oder eine Abbruchbedingung (Anzahl Iterationen) erfüllt ist:

1. Erzeugen der initialen Population aus zufällig generierten CSNs oder durch Mutieren eines vorgegebenen CSN
2. Parameter Fitting: Die Parameter der CSNs werden per Evolutionärer Strategie entsprechend der Zielfunktion optimiert
3. Berechnen der Fitness aller CSNs der Population mit Hilfe der Zielfunktion
4. Selektion der Eltern für die nächste Generation
5. Erzeugen der nächsten Generation durch Mutation

Die Zielfunktion, welche der Algorithmus zum Berechnen der Fitness verwenden soll, muss in der Datei `objectivefunction.txt` angegeben werden, zusätzlich zu Anzahl und Namen der Input- und Output-Spezies. Gibt man dort `‘io-table’` an und einen gewünschten Zeitverlauf der Output-Spezies, dann werden die Fitnesswerte der CSNs im Vergleich dazu berechnet. Mit der Zielfunktion `‘target-csn’` werden CSNs evolviert, welche im Verhalten einem in der Datei `target-csn.xml` vorgegebenen SBML Modell möglichst ähnlich sind. Nachfolgend werden ausgewählte experimentelle Ergebnisse zur künstlichen Evolution oszillierender CSNs mit der SBMLEvolver Software vorgestellt.

#### 3.1 Einfache Sinus-Schwingung

Vorgegeben wurden in der Datei `objectivefunction.txt` eine oder mehrere verschiedene Werteverläufe einer Sinus-Schwingung, als einfachste Form einer Oszillation. Die Läufe wurden mit den Parametern `-p 100 -s 50 -e 50 -I 10` gestartet.

Zunächst fanden fünf Fälle Verwendung, mit den `input`-Werten 1, 2, 3, 0.5 und 0.25. Der dazugehörige `output`-Werteverlauf besteht jeweils aus 50 Werten und umfasst im ersten Fall eine komplette Schwingung (0 bis  $2\pi$ ), im zweiten zwei und im dritten drei Schwingungen, im vierten eine halbe und im fünften eine viertel Schwingung (jeweils passend zum `input`).

Die Fitness der entstandenen Individuen (Abb. 3) lag bei durchschnittlich 1.38 (Optimum = 1), aber ihre Werteverläufe oszillierten nicht (Abb. 4). Bei den meisten Individuen näherte sich der `output`-Wert einem konstanten Wert an, oder er war durchgängig konstant. Die konstanten `output`-Werte der Individuen entsprechen meist recht genau dem Mittel aller Werte der Sinus-Schwingung: in den ersten drei Fällen 0, im vierten 0.6237 und im fünften 0.6338.

In späteren Versuchen erfolgte eine Variierung der in `objectivefunction.txt` vorgegebenen Werteverläufe, zB. wurde nur ein Fall mit `input`-Wert 1 und eine einfache Schwingung als `output`-Werte (50 Werte, später 99) verwendet. Die entstandenen Individuen hatten erneut eine Fitness um 1.38, und zeigten keine Oszillation (Abb. 5, 6).

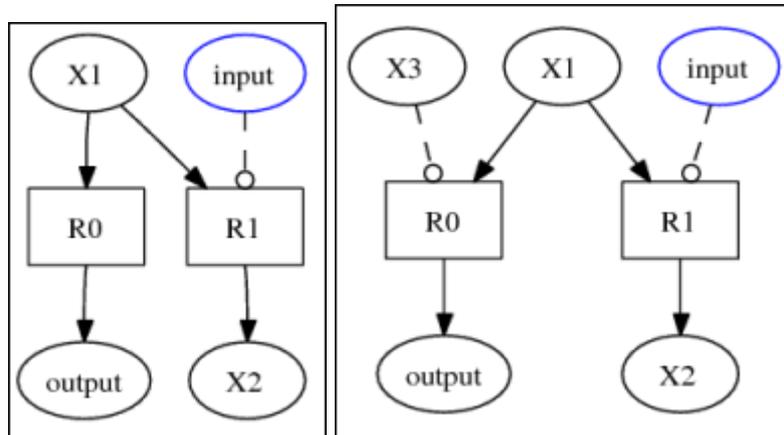


Abbildung 3: Zwei Beispiele für CSNs, welche mit mehreren Sinusschwingungen als Zielfunktion evolviert wurden

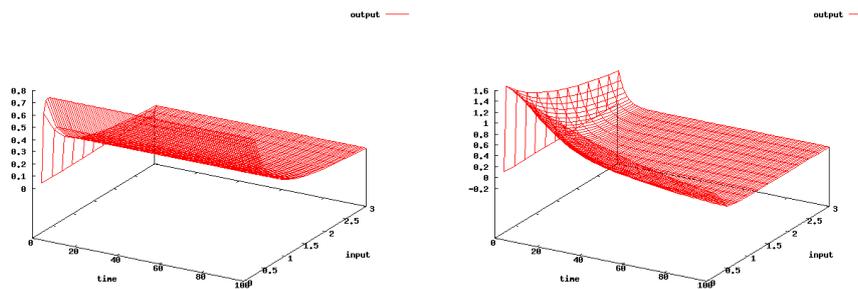


Abbildung 4: Typisches Verhalten der CSNs aus Abb. 3

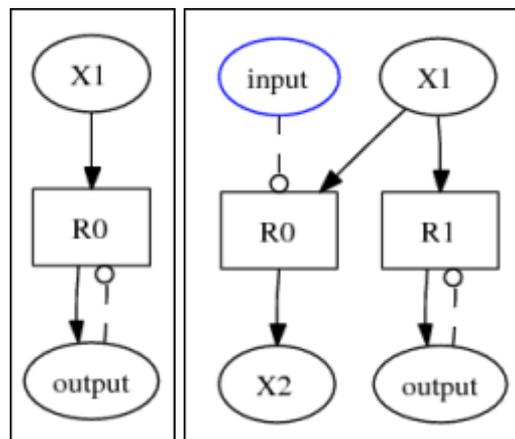


Abbildung 5: Zwei Beispiele für CSNs, welche mit einer Sinusschwingung als Zielfunktion evolviert wurden

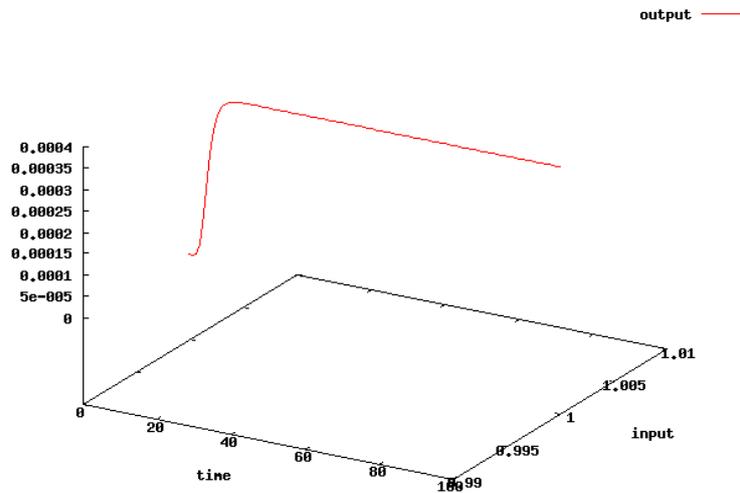


Abbildung 6: Verhalten des rechten CSNs aus Abb. 5

### 3.2 Evolution nach einem target-csn: biochemisches Netz mit Oszillation

Bei diesen Versuchen wurde ein SBML-Netzwerk mit bestimmtem, oszillierendem Verhalten als target-csn vorgegeben. Zur Berechnung der Fitness der CSNs vergleicht der Algorithmus die Zeitverläufe *aller* chemischen Spezies der CSNs mit denen des target-csn. Netzwerke, welche eine andere Anzahl chemischer Spezies als das target-csn haben, bekommen einen sehr schlechten (hohen) Fitnesswert. Ein Netzwerk, dessen dynamisches Verhalten exakt dem vorgegebenen Zeitverlauf entspricht, besitzt den Fitnesswert 0.

Aus der BioModels Database<sup>2</sup> für Modelle biochemischer Netzwerke wurde ein recht einfaches, oszillierendes Modell gewählt (Abb. 7, 8). Die Testläufe wurden mit den Parametern -L 5 3 10 -I 10 -e 5 durchgeführt. Ein Fehler im Programm führte allerdings dazu, dass der Algorithmus meist nach der fünften, spätestens aber nach der achten Generation mit Fehlermeldung abbrach. Leider kann man nicht sagen, welche Qualität die entstandenen Netzwerke bis dahin hatten, da sie bis zu diesem Zeitpunkt alle eine Fitness von mindestens 9999999.0 hatten, dh. die Anzahl ihrer chemischen Spezies entsprach meist nicht der des target-csn. Offenbar dauert es (vor allem bei großen Netzen) recht lange, bis Individuen mit der richtigen Anzahl Spezies entstehen. Es kann deshalb sinnvoller sein, ein Netz mit der richtigen Spezieszahl als Vorlage für die initiale Population vorzugeben.

#### 3.2.1 Abbildungen

<sup>2</sup><http://www.ebi.ac.uk/biomodels/>

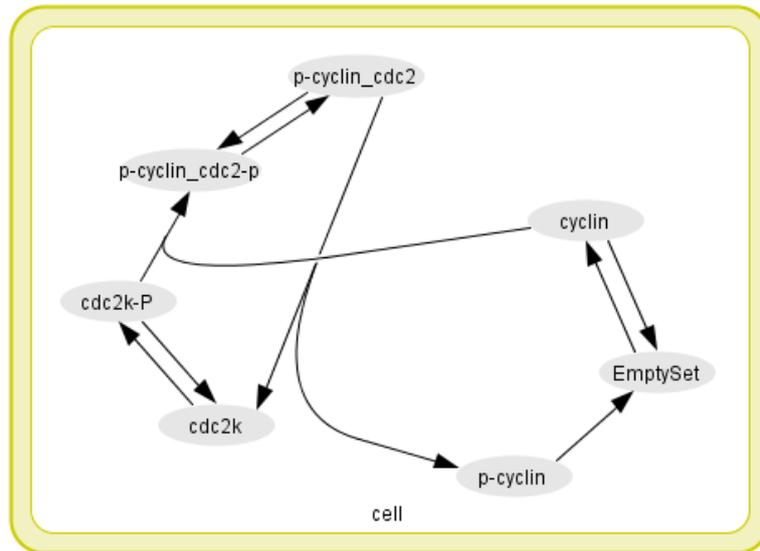


Abbildung 7: Ein reales biochemisches Netz mit Oszillation als target-csn

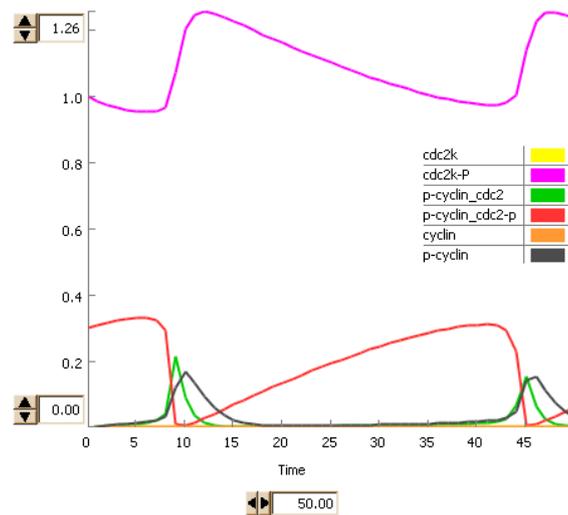


Abbildung 8: Die Oszillation des target-csn aus Abb. 7

### 3.3 Das Lotka-Volterra-Modell

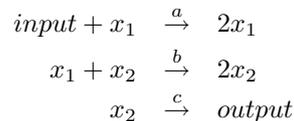
Für einen weiteren Versuch mit einem target-csn wurde ein CSN verwendet, welches die Populationsdynamik eines Räuber-Beute-Systems[4] modelliert - im Gegensatz zu den bisherigen CSNs, welche ausschließlich biochemische Netzwerke modellierten. Das verwendete Räuber-Beute-Modell ("Lotka-Volterra-Modell") ist einfacher aufgebaut als das in Kapitel 2.2 beschriebene. Die Populationsdichten der zwei Spezies oszillieren hier für jede beliebige Parameterkombination um einen metastabilen Fixpunkt.

#### 3.3.1 Das target-csn

Die zwei simplen Differentialgleichungen

$$\begin{aligned} \dot{x}_1 &= a \cdot x_1 - b \cdot x_1 \cdot x_2 \\ \dot{x}_2 &= b \cdot x_1 \cdot x_2 - c \cdot x_2 \end{aligned}$$

erzeugen eine Oszillation von  $x_1$  (der Beute) und  $x_2$  (den Räubern). Dabei sind  $a$ ,  $b$  und  $c$  die Reaktionsraten der drei entsprechenden Reaktionen:



Daraus ergibt sich das in Abb. 9 dargestellte Netzwerk mit dem in Abb. 10 dargestellten Verhalten. Die Konzentration von *input* ist dabei konstant 1.0, die Ausgangskonzentrationen von  $x_1 = 6.0$ , von  $x_2 = 2.0$  und von *output* = 0.0. Die Reaktionsraten  $a$ ,  $b$  und  $c$  haben alle den Wert 0.5.

#### 3.3.2 Die Versuche

Der SBMLEvolver wurde mit den Parametern `n -e 5 -L 10 5 5 -I 10` bzw. `-e 5 -I 15` gestartet, und brachte jeweils ähnliche Ergebnisse. Die Fitness fiel auf 1.2 bis 1.1. Aber auch hier zeigen die entstandenen Individuen nur konstantes oder nahezu konstantes Verhalten, welches meist einfach durch Reaktionen mit der Reaktionsrate 0 realisiert wurde. Ihre chemischen Spezies nähern sich in jedem Lauf bestimmten, ähnlichen Werten an. Zwei Beispiele sind in der Tabelle gegeben.

$x_1$	$x_2$	$x_3$	$x_4$
1.04024	0.341166	4.72291	2.28089

$x_1$	$x_2$	$x_3$	$x_4$
1.39309	0.224353	4.95099	1.78269

Dabei entspricht im ersten Beispiel  $x_1$  der target-csn Spezies *input*,  $x_3$  dem *output* und  $x_2$ ,  $x_4$  entsprechenden Spezies  $x_1$  und  $x_2$  aus dem target-csn (ungefähre Durchschnittswerte).

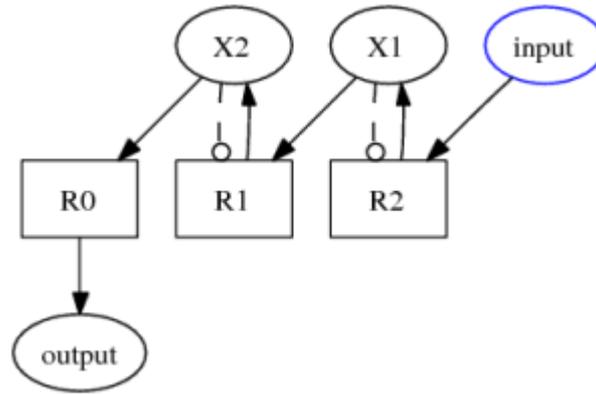


Abbildung 9: Ein Netzwerk als target-csn, welches das Lotka-Volterra-Modell modelliert

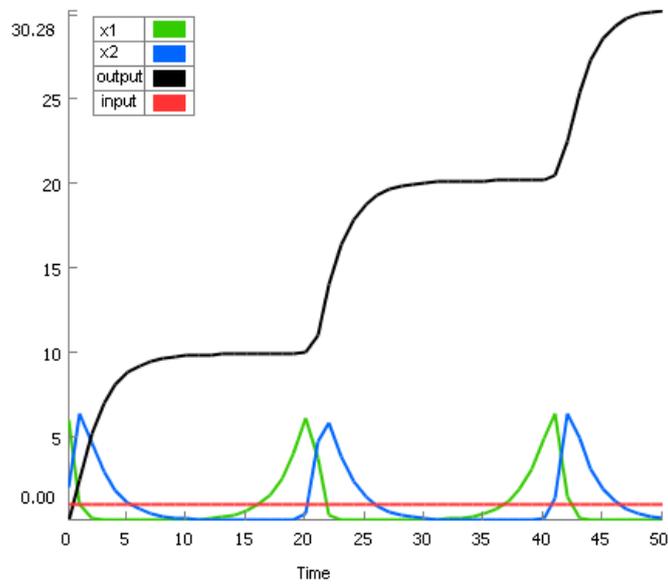


Abbildung 10: Die Oszillation des target-csn

## 4 In silico Evolution funktioneller Einheiten in biochemischen Netzwerken

In [8] wurden ebenfalls Methoden zur künstlichen Evolution biochemischer Reaktionsnetzwerke untersucht. Die zentralen Ergebnisse sollen im Folgenden kurz dargestellt werden. Es ist einfacher, ein komplexes biochemisches Netzwerk zu verstehen, wenn man es in Module, also Teilnetze mit bestimmten Fähigkeiten, unterteilt. Da das Wissen über die Eigenschaften solcher Netz-Module noch gering ist, wurden viele Netzwerke mit verschiedenen biologisch relevanten Fähigkeiten (Oszillatoren, bistabile Schalter, homöostatische Systeme und Frequenzfilter) mit einem evolutionären Algorithmus erzeugt und damit eine Datenbank zusammengestellt. Dabei fanden drei verschiedene Netzwerkkonstruktionen Verwendung, die alle mit Differentialgleichungssystemen simuliert wurden:

- Einfache biochemische Netzwerke basierend auf Massenwirkungskinetik, wobei Reaktionen der Typen  $A \rightarrow B$ ,  $A \rightarrow B + C$ ,  $A + B \rightarrow C$  und  $A + B \rightarrow C + D$  möglich waren.
- Protein-Protein-Netzwerke (Signaltransduktionsnetze) basierend auf Michaelis-Menten-Kinetik mit Enzym-Aktivierung und Inhibierung.
- Gen-Protein-Netzwerke basierend auf Hill-Kinetik, wobei die Proteine als Aktivator oder Repressor der Gene wirken konnten.

### 4.1 Evolution von oszillierenden Netzwerken

Der verwendete evolutionäre Algorithmus läuft nach folgendem Schema ab:

**(1) Initialisierung.** Die initiale Generation wird, je nach verwendeter Netzwerkstruktur, zufällig erzeugt.

- Massenwirkungskinetik: Erzeugen der Knotenmenge (Spezies), Verbinden jedes Knotens mit einer der möglichen Reaktionsarten mit anderen Knoten.
- Michaelis-Menten-Kinetik: Erzeugen einer Menge an Protein-Protein-Zyklen, Verbinden der Zyklen durch Aktivierung oder Inhibition.
- Hill-Kinetik: Erzeugen einer Menge an Gen-Protein-Paaren, Verbinden der Paare: das Protein eines Paares wirkt positiv oder negativ auf die Genexpression eines anderen Paares.

**(2) Berechnung der Fitness.** Per Zufall wird einer der Knoten, von dem nur Kanten weg führen, als Input-Knoten gewählt. Als Output-Knoten wird zufällig ein Knoten gewählt, der ein- und ausgehende Kanten hat. Dann wird das Netzverhalten mit zufällig generierten Startwerten simuliert. Die Zielfunktion, mit der die Fitness des Netzwerkes berechnet wird, hängt von der Fähigkeit ab, die das evolvierte Netzwerk haben soll. Um Oszillatoren zu evolvierten, wurde jeweils eine von zwei verschiedenen Zielfunktionen genutzt:

- Der Fehler zwischen dem Zeitverlauf der Simulation des Netzwerkes und einer Reihe von alternierenden Punkten soll minimiert werden.
- Die Eigenwerte des Differentialgleichungssystems werden berechnet. Wenn sie bei bestimmten Parameterwerten komplex werden, kann das System oszillieren. Diese Eigenschaft wird ausgenutzt, indem der Ausdruck  $\epsilon = \frac{\prod_i \lambda_i^R}{\prod_i (1 - 0.99 * \exp(-|\lambda_i^I|))}$  minimiert wird ( $i$ : Indizes der Spezies).

**(3) Selektion.** Zunächst werden die Netzwerke nach ihrer Fitness sortiert. Dann werden sie, je nach Art der Netzwerkstruktur, verschieden ausgewählt.

- Massenwirkungskinetik und Michaelis-Menten-Kinetik: Truncation Selection, dh. die besten 30% dienen als Eltern für die nächste Generation.
- Hill-Kinetik: Elitismus in Kombination mit Tournament Selection, dh. die besten 10% werden direkt in die nächste Generation übernommen. Die Eltern für den übrigen Teil der Nachfolgegeneration werden per Tournament Selection ausgewählt: es werden jeweils 2 Netzwerke zufällig gezogen, das bessere wird gewählt.

**(4) Erzeugen der nächsten Generation.** Die Netzwerke der nächsten Generation werden ausschließlich durch Mutation aus den in Schritt 3 bestimmten Eltern erzeugt. Dabei werden 2 verschiedene Arten der Mutation angewandt:

- Mutation der Netzstruktur: Es werden per Zufall Knoten und Kanten hinzugefügt oder entfernt (drastischere Veränderungen).
- Mutation der Reaktionsraten: Die Raten werden um eine Zufallszahl erhöht oder verringert (feinere Änderungen).

Danach werden die Netzwerke getrimmt, dh. alle Kanten, deren Entfernung aus dem Netz das Netzverhalten *nicht* ändern, werden entfernt.

**(5) Abbruchbedingung.** Wenn ein Netzwerk mit der gewünschten Fitness vorliegt, ist der Algorithmus beendet.

## 4.2 Ergebnisse

In [8] wurden für alle drei Netzwerkarchitekturen und für beide Zielfunktionsstypen erfolgreich Oszillatoren evolviert, wobei sich zwei Ergebniskategorien herausgestellt haben:

- Feedback-Oszillatoren, bei denen die Oszillationen ausschließlich durch (eventuell negative) Feedback-Loops entstehen, wobei die Phasenunterschiede zwischen den Oszillationen verschiedener Stoffkonzentrationen diese gegenseitig ständig aus dem Gleichgewicht treiben. Ein Beispiel für ein solches Netzwerk mit Massenwirkungskinetik ist in Abb. 11 gegeben, Ausgangskonzentrationen und Reaktionsparameter sind:

$s_1$	$s_2$	$s_3$	$s_4$	$s_5$	$s_6$	$s_7$	$s_8$
1.0	1.0	1.0	1.0	1.0	4.0	1.0	1.0
$k_1$	$k_2$	$k_3$	$k_4$	$k_5$	$k_6$	$k_7$	$k_8$
1.0	1.0	5.0	5.0	1.0	4.0	1.0	5.0

- Relaxationsoszillatoren. Hierbei entstehen die Oszillationen durch das exponentiell schnelle “Aufladen” einer Komponente, vergleichbar mit einem Kondensator. Ab einem bestimmten oberen Schwellwert wird der Kondensator wieder entladen, bis zu einem bestimmten unteren Schwellwert, und so weiter. Diese Oszillatoren benötigen zusätzlich zu Feedback-Loops auch autokatalytische Reaktionen. Ein Beispiel für ein solches Netzwerk mit Michaelis-Menten-Kinetik ist in Abb. 12 gegeben, die Reaktionsparameter sind:

			$s_1$	$s_2$	$s_3$	$s_4$	$s_5$	$s_6$				
			0.0	2.0	2.0	0.0	0.0	2.0				
$r1_1$	$r1_2$	$r2_1$	$r2_2$	$r3_1$	$r3_2$	$r4_1$	$r4_2$	$r5_1$	$r5_2$	$r6_1$	$r6_2$	
1.0	0.5	1.0	0.5	1.0	0.5	1.0	0.5	1.0	0.01	1.0	0.01	

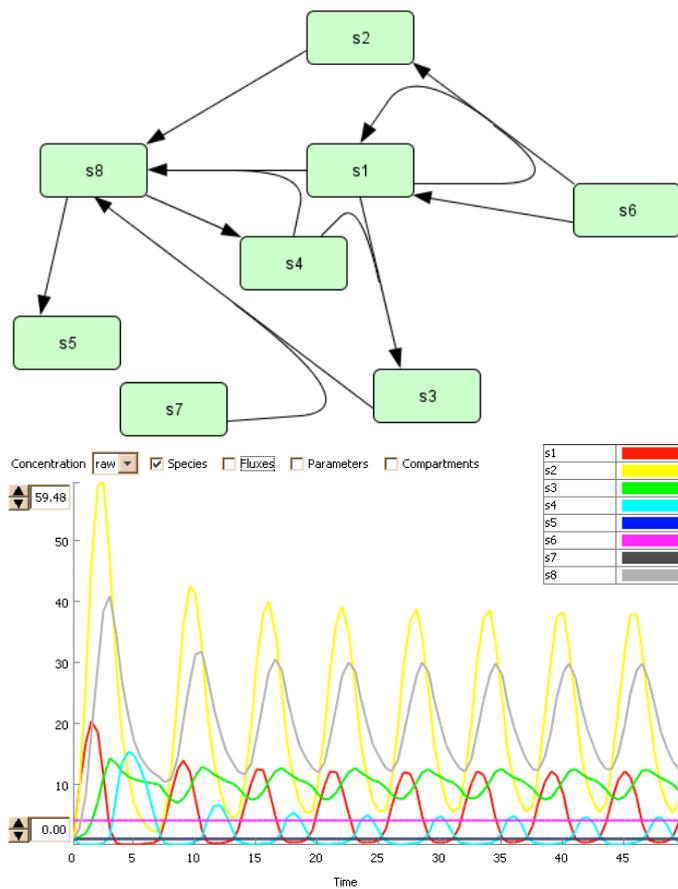


Abbildung 11: ein Feedback-Oszillator basierend auf Massenwirkungskinetik

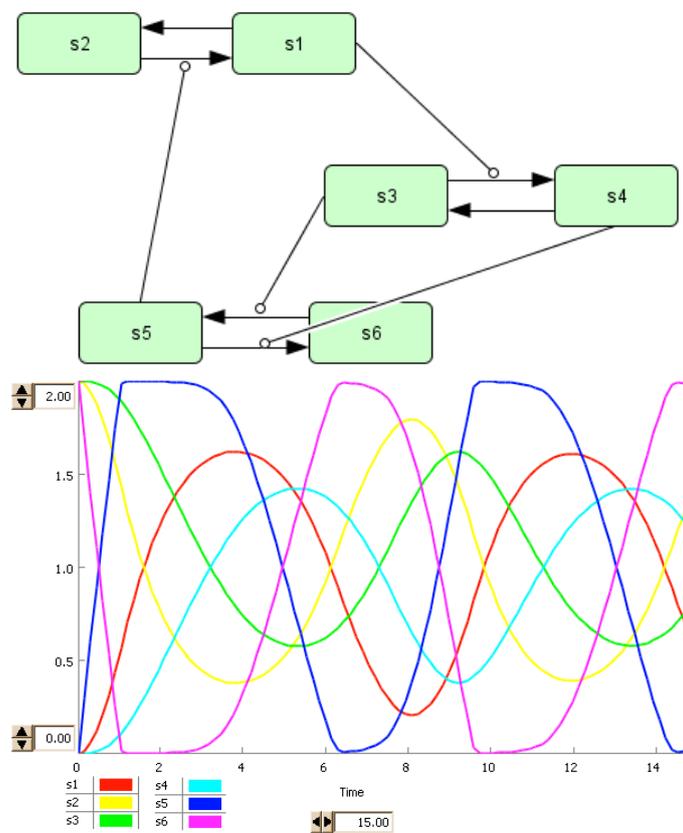


Abbildung 12: ein Relaxations-Oszillator basierend auf Michaelis-Menten-Kinetik

## 5 Diskussion der Ergebnisse

Im Rahmen der hier beschriebenen Versuche mit dem SBMLEvolver (Kapitel 3) konnten keine oszillationsfähigen CSNs evolviert werden, an anderer Stelle (Kapitel 4) war der Versuch jedoch erfolgreich. Eine mögliche Ursache dafür ist die Schwierigkeit, überhaupt oszillierende CSNs mit einem evolutionären Algorithmus zu “finden”.

Das Problem sind die Größe und Struktur des Suchraums, also des Raumes aller möglichen Netze. Jede strukturelle Änderung (Anzahl der Spezies und Reaktionen, Relationen zwischen den Spezies) verändert das Verhalten eines Netzes grundlegend und auch die Änderung der Parameter hat einen großen Einfluß auf die Eigenschaften des Netzes, wie zum Beispiel Oszillationsfähigkeit. Beschränkt man den unendlichen Raum aller Netze auf einen endlichen Teilraum von Netzen mit biologisch “sinnvollen” Spezies- und Reaktionszahlen, und betrachtet man die möglichen Parameter in einem Raster, so ergibt sich eine Suchraumgröße von mehreren Millionen Netzen. Die Fähigkeit zu oszillieren ist nun in diesem Suchraum nur punktuell zu finden, dh. nur ganz bestimmte Spezies-, Reaktions- und Parameterkombinationen ergeben ein oszillierendes Netz. Schon bei geringfügiger Veränderung der Netzstruktur bzw. der kinetischen Parameter geht diese Fähigkeit wieder verloren.

Um weiterführend diese Struktur des Suchraums zu überprüfen, könnte man an bekannten, oszillationsfähigen Netzen durch geringe Variation der Eigenschaften testen, wie schnell sie diese Fähigkeit verlieren. Dafür sprechen auch natürliche biochemische Netze. Die Menge der oszillationsfähigen Netze in der Natur macht nur einen sehr kleinen Anteil aller bekannten Netze aus. Diese Fähigkeit scheint auch hier nur unter ganz bestimmten Bedingungen aufzutreten.

Wenn man nun mit Hilfe eines evolutionären Algorithmus nach oszillierenden Netzen sucht, werden die meisten Individuen der Population sehr schlechte Fitnesswerte haben, da sie nicht oszillieren können. Auch im Suchraum nahe “neben” oszillationsfähigen Netzen liegende Netze haben keine bessere Fitness. Deshalb ähnelt die Suche des evolutionären Algorithmus eher einer Zufallssuche, welche erst erfolgreich ist, wenn durch Zufall eine der seltenen Spezies-, Reaktions- und Parameterkombinationen eines oszillationsfähigen Netzes gefunden wurden.

Ausgehend von der geschätzten Suchraumgröße und -struktur ist die Wahrscheinlichkeit, ein solches Netz zu finden,  $10^{-6}$  bis  $10^{-7}$ . Man kann also davon ausgehen, dass eine erfolgreiche heuristische Suche recht selten ist, es sei denn, man arbeitet tatsächlich den gesamten Suchraum ab.

## Literatur

- [1] [http://de.wikipedia.org/wiki/Circadianer\\_Rhythmus](http://de.wikipedia.org/wiki/Circadianer_Rhythmus), 09 2007.
- [2] <http://de.wikipedia.org/wiki/Oszillatorschaltung>, 09 2007.
- [3] <http://www.esignet.net/>, 09 2007.
- [4] <http://de.wikipedia.org/wiki/Lotka-Volterra-Gleichungen>, 03 2008.
- [5] J.C. Dunlap. Molecular Bases for Circadian Clocks Review. *Cell*, 96:271–290, 1999.
- [6] G. Jetschke. *Mathematik der Selbstorganisation: Qualitative Theorie nichtlinearer dynamischer Systeme und gleichgewichtsferner Strukturen in Physik, Chemie und Biologie*. Deutscher Verlag der Wissenschaften, 1989.
- [7] T. Lenser, T. Hinze, B. Ibrahim, and P. Dittrich. Towards Evolutionary Network Reconstruction Tools for Systems Biology. *Lecture Notes in Computer Science*, 4447:132–142, 2007.
- [8] SR Paladugu, V. Chickarmane, A. Deckard, JP Frumkin, M. McCormack, and HM Sauro. In silico evolution of functional modules in biochemical networks. *IEE Proceedings-Systems Biology*, 153:223–235, 2006.

- [9] N. Price-Lloyd, M. Elvin, and C. Heintzen. Synchronizing the *Neurospora crassa* circadian clock with the rhythmic environment. *Biochem. Soc. Trans*, 33:949–952, 2005.