

Methodology article

Open Access

## Transforming Boolean models to continuous models: methodology and application to T-cell receptor signaling

Dominik M Wittmann<sup>1,2</sup>, Jan Krumsiek<sup>1</sup>, Julio Saez-Rodriguez<sup>3,4</sup>, Douglas A Lauffenburger<sup>3</sup>, Steffen Klamt<sup>5</sup> and Fabian J Theis\*<sup>1,2,6</sup>

Address: <sup>1</sup>Institute for Bioinformatics and Systems Biology, Helmholtz Zentrum München - German Research Center for Environmental Health, 85764 Neuherberg, Germany, <sup>2</sup>Department of Mathematical Science, Technische Universität München, 85747 Garching, Germany, <sup>3</sup>Biological Engineering Department, M.I.T., Cambridge MA 02139, USA, <sup>4</sup>Department of Systems Biology, Harvard Medical School, Boston MA 02115, USA, <sup>5</sup>Max Planck Institute for Dynamics of Complex Technical Systems, 39106 Magdeburg, Germany and <sup>6</sup>Max Planck Institute for Dynamics and Self-Organisation, 37073 Göttingen, Germany

E-mail: Dominik M Wittmann - dominik.wittmann@helmholtz-muenchen.de; Jan Krumsiek - jan.krumsiek@helmholtz-muenchen.de; Julio Saez-Rodriguez - julio@hms.harvard.edu; Douglas A Lauffenburger - lauffen@mit.edu; Steffen Klamt - klamt@mpi-magdeburg.mpg.de; Fabian J Theis\* - fabian.theis@helmholtz-muenchen.de

\*Corresponding author

Published: 28 September 2009

Received: 19 January 2009

*BMC Systems Biology* 2009, **3**:98 doi: 10.1186/1752-0509-3-98

Accepted: 28 September 2009

This article is available from: <http://www.biomedcentral.com/1752-0509/3/98>

© 2009 Wittmann et al; licensee BioMed Central Ltd.

This is an Open Access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/2.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

### Abstract

**Background:** The understanding of regulatory and signaling networks has long been a core objective in Systems Biology. Knowledge about these networks is mainly of qualitative nature, which allows the construction of Boolean models, where the state of a component is either 'off' or 'on'. While often able to capture the essential behavior of a network, these models can never reproduce detailed time courses of concentration levels.

Nowadays however, experiments yield more and more quantitative data. An obvious question therefore is how qualitative models can be used to explain and predict the outcome of these experiments.

**Results:** In this contribution we present a canonical way of transforming Boolean into continuous models, where the use of multivariate polynomial interpolation allows transformation of logic operations into a system of ordinary differential equations (ODE). The method is standardized and can readily be applied to large networks. Other, more limited approaches to this task are briefly reviewed and compared. Moreover, we discuss and generalize existing theoretical results on the relation between Boolean and continuous models. As a test case a logical model is transformed into an extensive continuous ODE model describing the activation of T-cells. We discuss how parameters for this model can be determined such that quantitative experimental results are explained and predicted, including time-courses for multiple ligand concentrations and binding affinities of different ligands. This shows that from the continuous model we may obtain biological insights not evident from the discrete one.

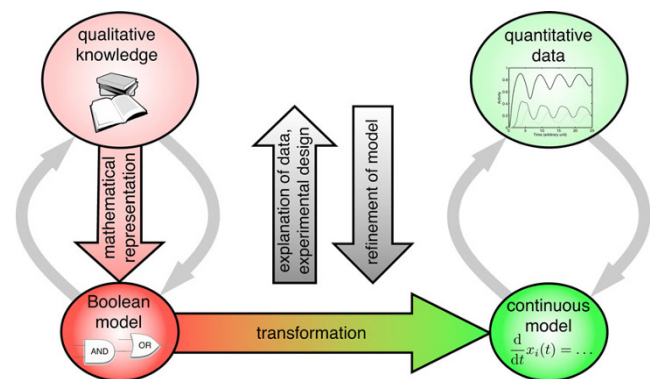
**Conclusion:** The presented approach will facilitate the interaction between modeling and experiments. Moreover, it provides a straightforward way to apply quantitative analysis methods to qualitatively described systems.

## Background

Close interaction between experiments and mathematical models has proven to be a powerful research approach in Systems Biology. Especially the modeling of regulatory and signaling networks, however, is typically hampered by a lack of information about mechanistic details, as often one can only determine the interactions of the involved species in a qualitative way. The current shift of focus in Systems Biology from single signal transduction pathways to networks of pathways exacerbates this lack of information even more. Therefore, the creation of mass action based models that accurately describe the underlying biochemistry is typically restricted to small well-studied subsystems.

Large-scale models of regulatory or signaling networks are often so-called *Boolean models* [1]. In fact, these models can be seen as the mathematically rigorous representation of qualitative biological knowledge. Their components, henceforth called *species*, can have only discrete states, typically two; these may be referred to as 0 and 1, 'off' and 'on', 'deactivated' and 'activated', etc. Time is discretized and the state of a species at time  $t + 1$  is a function of the states of the species at time  $t$ . Although being a crude simplification of biological reality, Boolean models are often able to reproduce the qualitative behavior of a system [2-8]. Naturally, Boolean models can neither describe continuous concentration levels nor realistic time scales. For this reason, they cannot be used to explain and predict the outcome of biological experiments that yield quantitative data. However, with increasing emphasis on these quantitative experiments the need for precisely this kind of model arises. In this contribution, we present and exemplify a practicable solution to this problem: a standardized method for accurately converting any Boolean model into a continuous model. This transformation fills the gap in the modeling process shown in Figure 1. It allows construction of a continuous model from qualitative knowledge by representing this knowledge as a Boolean model and then transforming this discrete model into a continuous one. The continuous model can now be used to explain experimental results and to design and optimize further experiments. The results of these experiments, in turn, help to refine the model.

Boolean models are a very coarse description of biochemical processes. They phenomenologically describe observed dependencies often leaving out still unknown players or intermediate steps. As our transformation requires no additional information the resulting continuous models are, of course, still phenomenological models. We can automatically create these continuous phenomenological models out of a Boolean



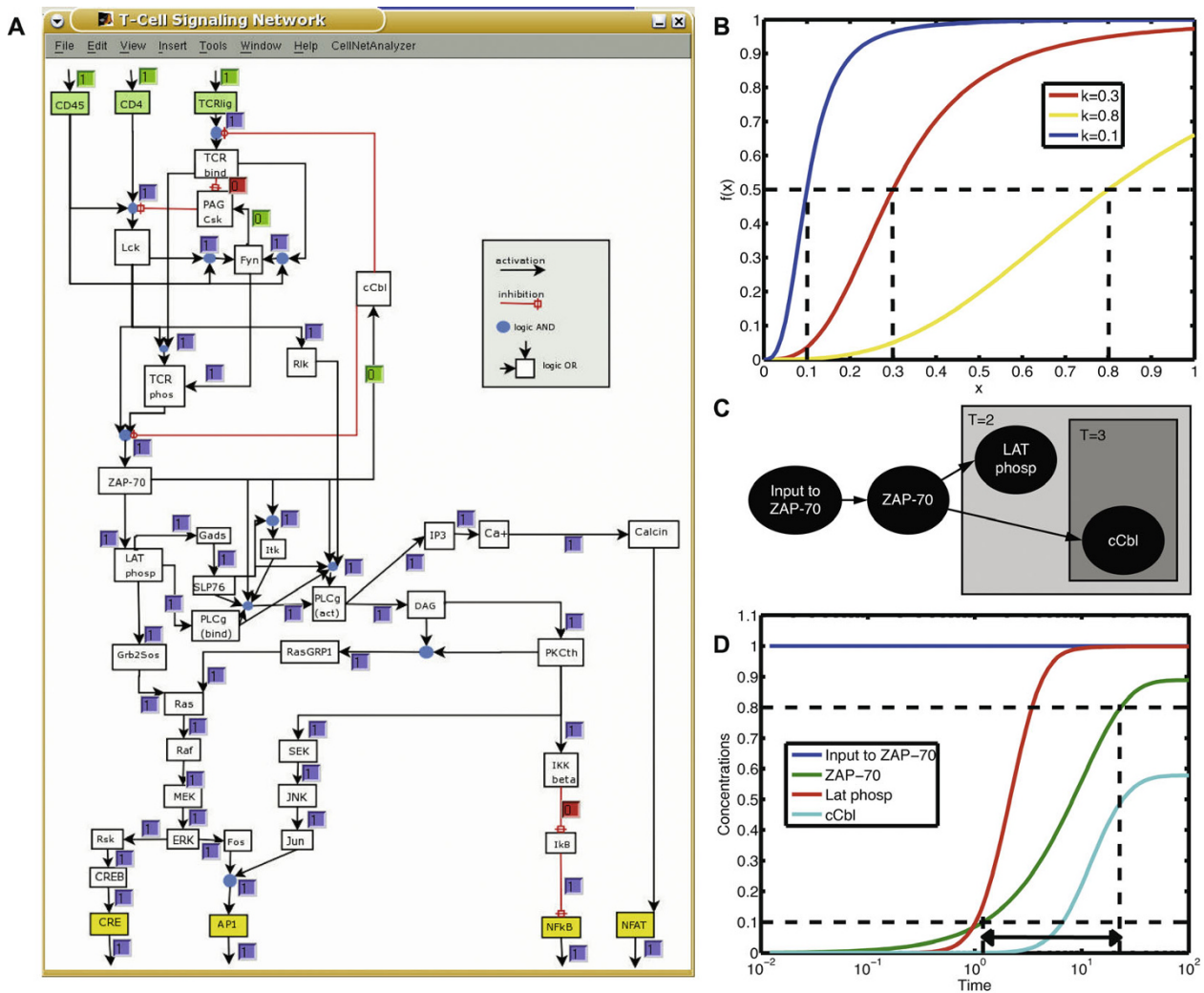
**Figure 1**  
**Overview of the modeling process.** The typically qualitative biological knowledge is mathematically rigorously represented as a Boolean model, that is then converted into a continuous model. This continuous model can be used to explain quantitative experimental results and to design and optimize further experiments; the results of these experiments, in turn, help to refine the model.

model, but we cannot create a mass action law without additional knowledge on the biochemistry. Our method is a top-down approach for (large) networks with incomplete mechanistic knowledge — derived e.g. from pathway databases — where predictive kinetic modeling is infeasible. The main point that we want to make in this contribution is that also these phenomenological models can be used in Systems Biology to explain and predict quantitative experimental results.

To this end, we focus on a large Boolean model of T-cell activation proposed by Klamt et al. [6]. These cells play a pivotal role in the immune system. When foreign antigens bind to their receptors, signaling cascades are triggered within the T-cell leading to an activation of several transcription factors. The logical structure of the model is shown in Figure 2A. There are three inputs: the T-cell receptor TCR, the coreceptor CD4 and an input for CD45; as well as four outputs: the transcription factors CRE, AP1, NFkB and NFAT. The rephosphorylation of PAG-Csk by Fyn and cCbl mediated degradation are known to be slow processes compared to the other interactions. This is modeled by activating the feedback loops  $\text{Fyn} \rightarrow \text{PAG-Csk}$  and  $\text{ZAP-70} \rightarrow \text{cCbl}$  only at a later stage. Therefore three scenarios are defined:

**scenario 1, resting state:** All inputs are set to 0, feedback loops are deactivated.

**scenario 2, early events:** Inputs are set to 1, feedback loops are still deactivated.



**Figure 2**  
**T-cell model.** (A) Structure of the Boolean model as shown in CellNetAnalyzer [24]. (B) Hill functions with parameter  $n = 3$  and different thresholds  $k = 0.3$ ,  $k_{fast} = 0.1$  and  $k_{slow} = 0.8$ . (C) Subnet of the T-cell model for scenario 2 (only activation) and scenario 3 (activation and feedback). (D) Numeric simulation of the subnet from (C). Time is plotted on a log-scale. The two dotted vertical lines indicate the time points when the concentration of ZAP-70 exceeds  $k_{fast}$  and  $k_{slow}$ , respectively.

**scenario 3, mid-time events:** Inputs are still 1, feedback loops are active.

A qualitative analysis of an expanded version of this Boolean model by Saez-Rodriguez et al. [8] yielded new and non-obvious signaling pathways. There are also quantitative models covering aspects of T-cell signaling in mechanistic detail, e.g. [9]. In contrast thereto, the continuous model we obtain in the following describes the T-cell signaling cascade on a larger scale yet in less mechanistic detail.

We apply the transformation method to the T-cell model. The resulting continuous model is able to fully

reproduce the behavior of the Boolean model. Moreover, we show that it can easily include and deal with the different time scales of interactions in the three scenarios. Hence the continuous model is indeed a generalization of the Boolean model with richer dynamic properties. This is in line with previous findings [10,11], indicating that the qualitative behavior of a discrete model is reproduced by its continuous homologue. We can further corroborate this hypothesis by generalizing existing theoretical results on the steady-states of discrete and continuous models.

The crucial question is, of course, whether the continuous model derived from a Boolean network is indeed

competent to explain an aspect of biological reality in a precise quantitative fashion. We answer this question in the affirmative by showing that our T-cell model reproduces time courses of concentration levels measured for three different ligand concentrations [12]. Moreover, it is able to predict binding affinities of different ligands from their induced signaling profiles. The fact that the model can differentiate between more than two different concentrations shows that we have definitely left the Boolean (binary) world behind.

## Results

### Representation of Boolean functions and models

A Boolean model consists of

- $N$  species  $X_1, X_2, \dots, X_N$ , e.g. genes, proteins, etc., each represented by a variable  $x_i$  taking values in  $\{0, 1\}$ ,
- for each species  $X_i$  a set of species  $R_i := \{X_{i1}, X_{i2}, \dots, X_{iN_i}\} \subset \{X_1, X_2, \dots, X_N\}$  that influence  $x_i$  and
- for each species  $X_i$  an update function  $B_i : \{0, 1\}^{N_i} \rightarrow \{0, 1\}$

giving the value of  $x_i$  at the next time step for every possible combination of  $(x_{i1}, x_{i2}, \dots, x_{iN_i}) \in \{0, 1\}^{N_i}$ .

In the following we think of  $B_i$  as a function on the vertices of the unit cube. Since vertex  $(\xi_{i1}, \xi_{i2}, \dots, \xi_{iN_i}) \in \{0, 1\}^{N_i}$  corresponds to the AND gate

$$\left( \bigwedge_{ij|\xi_{ij}=1} x_{ij} \right) \wedge \left( \bigwedge_{ij|\xi_{ij}=0} \neg x_{ij} \right)$$

we can write  $B_i$  in the form

$$B_i(x_{i1}, x_{i2}, \dots, x_{iN_i}) = \bigvee_{\substack{(\xi_{i1}, \xi_{i2}, \dots, \xi_{iN_i}) \\ B_i(\xi_{i1}, \xi_{i2}, \dots, \xi_{iN_i})=1}} \left[ \left( \bigwedge_{ij|\xi_{ij}=1} x_{ij} \right) \wedge \left( \bigwedge_{ij|\xi_{ij}=0} \neg x_{ij} \right) \right].$$

This is a so-called *sum-of-product representation* of  $B_i$ . These representations are especially convenient, as they allow to graphically represent our models using *interaction hypergraphs* [6]. The idea is to represent each product (AND gate) in the sum-of-product form of  $B_i$  by a directed hyperedge between a set of start nodes  $S \subset \{X_{i1}, X_{i2}, \dots, X_{iN_i}\}$  and the end node  $X_i$ . Each pair  $(s, X_i)$ ,  $s \in S$ , carries a sign — '+' or '-' — depending on whether there is a factor  $s$  or  $\neg s$  in the product. All incoming hyperedges at node  $X_i$  are then a graphical

representation of  $B_i$ . This is further illustrated in Additional data file 1.

### The general approach to making discrete models continuous

The first step for obtaining a continuous model from a Boolean one is to replace the discrete variables  $x_i$  by continuous variables  $\bar{x}_i$  taking values in  $[0, 1]$ , i.e. we normalize concentrations to the unit interval. Consequently, we have to 'extend' the functions  $B_i$  to functions  $\bar{B}_i : [0, 1]^{N_i} \rightarrow [0, 1]$ . We call the functions  $\bar{B}_i$  *continuous homologues* of the Boolean functions  $B_i$ . The crucial point is, how the transformation of a Boolean function into a continuous homologue is performed. We will address this issue in the next section and continue with the outline of the general approach.

In the second step we have to specify how to build the actual continuous model, for which there are two possibilities. The most straightforward is probably to proceed analogously to the Boolean model, i.e. to use discrete time steps and to compute the value of  $\bar{x}_i$  at time  $t + 1$  by

$$\bar{x}_i(t + 1) = \bar{B}_i(\bar{x}_{i1}(t), \bar{x}_{i2}(t), \dots, \bar{x}_{iN_i}(t)). \tag{1}$$

In numeric simulations the discretization of time is obviously irrelevant. It complicates, however, the detection of small-scale continuous effects and is a serious drawback in the further investigation of the model by analytical methods.

Another way to build the continuous model is to try to mimic biological reality more closely: mRNAs, proteins, etc. are produced at a certain rate and are at the same time degraded. We assume the production of  $X_i$  to be given by  $\bar{B}_i$ , and the degradation to be proportional to  $\bar{x}_i$ . Then the development of  $\bar{x}_i$  over time is governed by the ordinary differential equation (ODE)

$$\dot{\bar{x}}_i = \frac{1}{\tau_i} (\bar{B}_i(\bar{x}_{i1}, \bar{x}_{i2}, \dots, \bar{x}_{iN_i}) - \bar{x}_i), \tag{2}$$

where  $\tau_i$  can be interpreted as the life-time of species  $X_i$ . Note that due to the normalization of concentrations to the unit interval we have only one parameter for production as well as decay. To clarify this, assume that  $\tilde{x}_i$  denotes the non-normalized concentrations and  $0 \leq \tilde{B}_i(\tilde{x}_{i1}, \tilde{x}_{i2}, \dots, \tilde{x}_{iN_i}) \leq 1$  the corresponding production functions. Then a general system of ODEs is of the form  $\dot{\tilde{x}}_i = \alpha_i \tilde{B}_i(\tilde{x}_{i1}, \tilde{x}_{i2}, \dots, \tilde{x}_{iN_i}) - \gamma_i \tilde{x}_i$ , where  $\alpha_i$  is the production rate and  $\gamma_i$  the decay rate of species  $X_i$ . Since the

maximal concentration of  $X_i$  is  $\alpha_i/\gamma_i$  we have the relation  $\bar{x}_i = (\gamma_i / \alpha_i)\tilde{x}_i$ . It now follows that

$$\begin{aligned} \dot{\tilde{x}}_i &= \frac{\gamma_i}{\alpha_i} \dot{\tilde{x}} = \frac{\gamma_i}{\alpha_i} \left( \alpha_i \tilde{B}_i(\tilde{x}_{i1}, \tilde{x}_{i2}, \dots, \tilde{x}_{iN_i}) - \gamma_i \tilde{x}_i \right) \\ &= \gamma_i \left( \tilde{B}_i \left( \frac{\alpha_{i1}}{\gamma_{i1}} \tilde{x}_{i1}, \frac{\alpha_{i2}}{\gamma_{i2}} \tilde{x}_{i2}, \dots, \frac{\alpha_{iN_i}}{\gamma_{iN_i}} \tilde{x}_{iN_i} \right) - \tilde{x}_i \right) \end{aligned}$$

and by setting  $\tau_i = 1 / \gamma_i$  and  $\bar{B}_i(\bar{x}_{i1}, \dots, \bar{x}_{iN_i}) = \tilde{B}_i((\alpha_{i1} / \gamma_{i1})\bar{x}_{i1}, \dots, (\alpha_{iN_i} / \gamma_{iN_i})\bar{x}_{iN_i})$  we obtain the ODEs (2).

Herein we focus on model (2), as this model can be further analyzed using the rich and mathematically rigorous theory of ODEs. Note furthermore that model (1) can be considered a special case of model (2) after numeric integration.

**Continuous homologues of Boolean functions**

As already mentioned, the key point is how a continuous homologue  $\bar{B}_i$  can be obtained from a Boolean function  $B_i$  in a computationally efficient manner. A suitable transformation has to satisfy three conditions:

**Accuracy:** It has to be accurate, which means that  $\bar{B}_i$  and  $B_i$  must agree on the vertices of the unit cube, i.e. for values in  $\{0,1\}^{N_i}$ . Functions  $\bar{B}_i$  satisfying this condition are called *perfect* continuous homologues. As will be shown, for perfect continuous homologues discrete and continuous models exhibit a similar steady-state behavior.

**Good analytical properties:** The functions  $\bar{B}_i$  should have good analytical properties such as smoothness in order to allow and facilitate a mathematical analysis of the system of ODEs.

**Minimality and uniqueness:** The functions  $\bar{B}_i$  should be the unique minimal solution in their interpolation class.

The three transformations we propose in the following are all based on multivariate polynomial interpolation [13,14] (see Methods). Here  $\bar{B}_i$  is defined as a polynomial in the variables  $\bar{x}_{i1}, \bar{x}_{i2}, \dots, \bar{x}_{iN_i}$  that agrees with  $B_i$  on the vertices of the unit cube. As will be shown in the Methods section, this technique satisfies all three of the above requirements. There are other approaches which we shortly review and compare in the Discussion section.

**BooleCube**

In a first step, we define the functions

$$\bar{B}_i^1(\bar{x}_{i1}, \bar{x}_{i2}, \dots, \bar{x}_{iN_i}) \tag{3}$$

by linearly interpolating the functions  $B_i$  using the technique of multivariate polynomial interpolation as explained in the Methods section. These functions are called *BooleCubes*. By substituting  $\bar{B}_i^1$  for  $B_i$  in equation (2), we can then define a system of ODEs that describes the temporal development of the  $\bar{x}_i$ .

**HillCube**

The functions  $\bar{B}_i^1$  are affine multilinear, i.e. for each  $1 \leq j \leq N_i$  and fixed  $\bar{x}_{ik}, k \neq j$ , there exist constants  $a, b \in \mathbb{R}$  such that  $\bar{B}_i^1(\bar{x}_{i1}, \bar{x}_{i2}, \dots, \bar{x}_{iN_i}) = a + b\bar{x}_{ij}$ . Molecular interactions, however, are known to show a switch-like behavior, which can be modeled using sigmoid shaped *Hill functions*  $f(\bar{x}) = \bar{x}^n / (\bar{x}^n + k^n)$  [15] (see Additional data file 2). The two parameters  $n$  and  $k$  have a clear biological meaning. The Hill coefficient  $n$  determines the slope of the curve and is a measure of the cooperativity of the interaction. The parameter  $k$  corresponds to the threshold in the Boolean model, above which one defines the state of a species as ‘on’. Mathematically speaking, it is the value at which the activation is half maximal.

We now define a Hill function  $f_{ij}$  with parameters  $n_{ij}$  and  $k_{ij}$  for every interaction and define new functions

$$\bar{B}_i^H(\bar{x}_{i1}, \bar{x}_{i2}, \dots, \bar{x}_{iN_i}) := \bar{B}_i^1(f_{i1}(\bar{x}_{i1}), f_{i2}(\bar{x}_{i2}), \dots, f_{iN_i}(\bar{x}_{iN_i})), \tag{4}$$

which we call *HillCubes*. Plots of the HillCubes of all 16 two-variable Boolean gates can be found in Additional data file 3. Now a new system of ODEs can be defined by replacing the  $\bar{B}_i$  in equation (2) by the HillCubes  $\bar{B}_i^H$ .

**normalized HillCube**

Note that Hill functions never assume the value 1, but approach it asymptotically. Hence, the HillCubes are not perfect homologues of the Boolean update functions  $B_i$ . If this is desired a simple solution is to normalize the Hill functions to the unit interval. This yields another (perfect) continuous homologue of the

Boolean functions  $B_i$

$$\bar{B}_i^{Hn}(\bar{x}_{i1}, \bar{x}_{i2}, \dots, \bar{x}_{iN_i}) := \bar{B}_i^1 \left( \frac{f_{i1}(\bar{x}_{i1})}{f_{i1}(1)}, \frac{f_{i2}(\bar{x}_{i2})}{f_{i2}(1)}, \dots, \frac{f_{iN_i}(\bar{x}_{iN_i})}{f_{iN_i}(1)} \right), \tag{5}$$

which we call *normalized HillCubes*, and thus defines also a new continuous ODE model.

### **Theoretical results about the relation between discrete and continuous models**

A natural and interesting question is how similar the discrete and the continuous model are. It can easily be shown (see Methods and [11]) that, whenever the  $\bar{B}_i$  in equation (2) are perfect homologues of the Boolean update functions  $B_i$ , a steady-state of the Boolean model will also be a steady-state of the continuous system. This result can be applied to the BooleCube model as well as the normalized HillCube model, but not to the non-normalized HillCube model. Therefore, Boolean steady-states will in general not be steady-states of the non-normalized HillCube system. The question is if Boolean steady-states still correspond to 'similar' steady-states of this continuous model, at least for certain parameters. Using the implicit function theorem we were able to show that this is indeed the case (see Methods). The reverse statement is, of course, not true, as in the continuous model additional (stable) steady-states may arise. Besides the steady-state behavior, monotony properties are a further important characteristic of a dynamical system. These properties determine the effect of a down- or up-regulation of a certain species on the other species. Due to its accuracy the presented transformation method preserves the monotony properties of a Boolean network. In the Discussion section we illustrate this further using the T-cell model as an example.

We show these results as they justify our transformation approach by demonstrating that it preserves essential properties of the Boolean model. For a deeper mathematical investigation of the relation between discrete and continuous models we refer the interested reader to the rich literature on this subject, e.g. [11,16,17]

### **Simulation of the Boolean T-cell model**

Figure 3A shows a simulation of the Boolean T-cell model using synchronous updates. In the beginning the three inputs were 0 and the system was in its resting state (scenario 1). At  $t = 10$  the inputs were manually switched on and the signaling cascade was triggered off, which at  $t = 24$  led to the activation of all four transcription factors CRE, AP1, NFkB and NFAT (scenario 2). At  $t = 57$  we activated the feedback loops Fyn  $\rightarrow$  PAG-Csk and ZAP-70  $\rightarrow$  cCbl. Consequently the cascade was blocked and at  $t = 67$  all transcription factors were again deactivated. So, essentially the simulation used three different Boolean models at times  $[0, 10)$ ,  $[10, 57)$  and  $[57, 100]$ . Note that in the Boolean model the time point for the activation of the feedback loops could be chosen arbitrarily. As is explained below, in the continuous model this time point was determined by our choice of the kinetic parameters. We chose  $t = 57$  since then

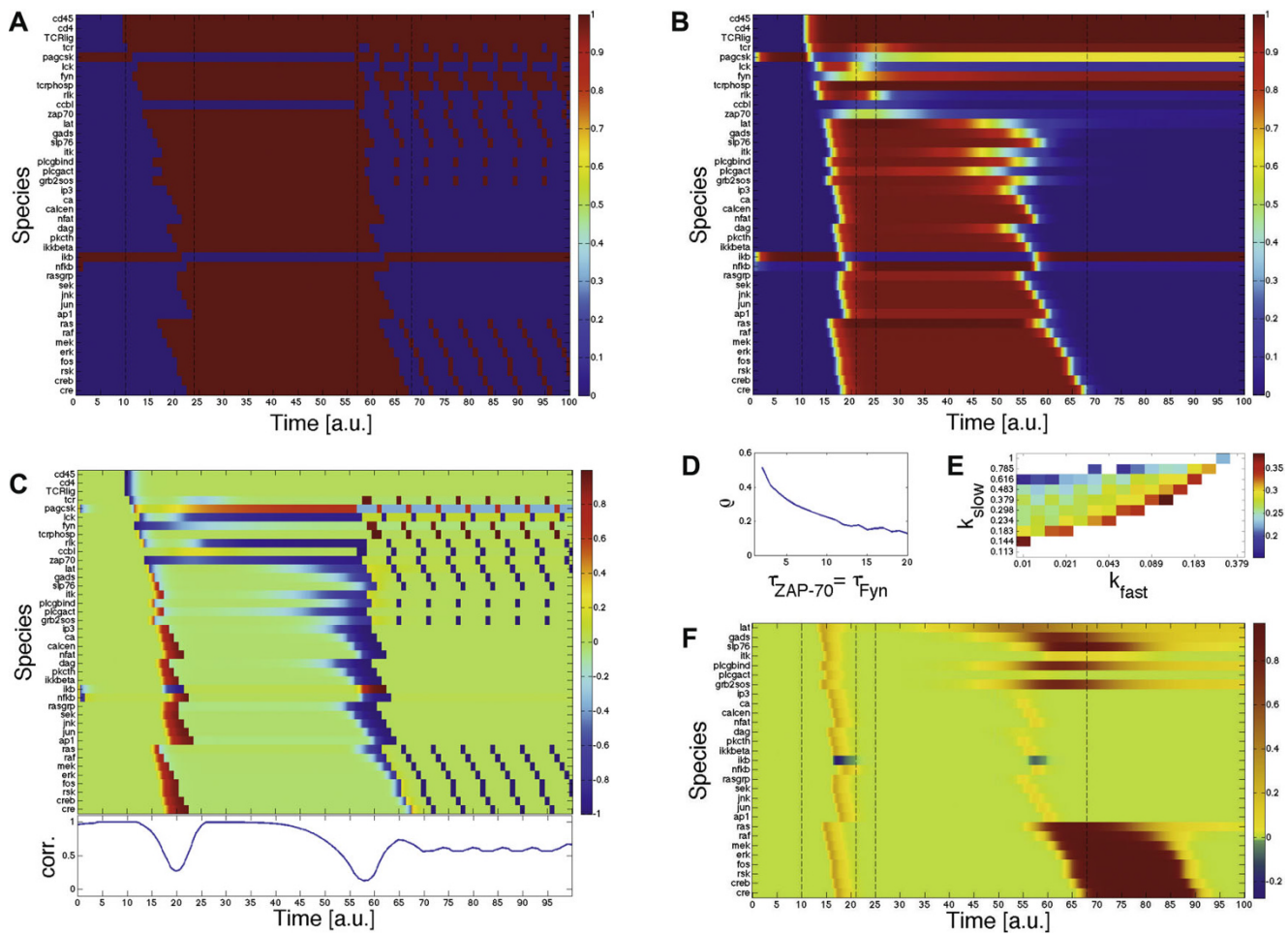
deactivation of all transcription factors occurred at around the same time point in both models.

After deactivation oscillations occurred, artefacts of the synchronous updating. When the species were updated asynchronously according to some permutation, these oscillations could be observed for 3078 out of 10000 randomly sampled permutations; in the other cases a steady-state was reached.

### **Continuous T-cell model**

We applied the transformation technique to the Boolean T-cell model. Using non-normalized HillCubes we obtained a large quantitative model of T-cell activation (see Methods). In a first step, we manually determined approximate parameters for which the continuous model reproduces the behavior of the Boolean model during all three scenarios. The information about fast and slow interactions was thereby encoded in the values of specific parameters. Consequently, the continuous model was able to explain both, the activation as well as the deactivation of T-cells, without any alteration of the network topology between different scenarios. Note that this was necessary in the Boolean case. We set all Hill coefficients to  $n = 3$ , the thresholds to  $k = 0.3$  and the life-times to  $\tau = 1$ , with the exception of Fyn and ZAP-70. Here we knew that the interactions Fyn  $\rightarrow$  PAG-Csk and ZAP-70  $\rightarrow$  cCbl operate on a slower time scale. Therefore the thresholds for these interactions were set to  $k_{\text{slow}} = 0.8$  whereas the thresholds for the interactions Fyn  $\rightarrow$  TCR-phos, ZAP-70  $\rightarrow$  LAT-phosp, ZAP-70  $\rightarrow$  PLCg (act) and ZAP-70  $\rightarrow$  Itk were set to  $k_{\text{fast}} = 0.1$ . The Hill functions for the three thresholds  $k$ ,  $k_{\text{slow}}$  and  $k_{\text{fast}}$  are displayed in Figure 2B. Finally, we set the life-times of Fyn and ZAP-70 to  $\tau_{\text{Fyn}} = \tau_{\text{ZAP-70}} = 10$  to enlarge the time gap between the two switching points.

The numeric simulation of the continuous T-cell model with the manually determined parameters is shown in Figure 3B. At first, the cell was again in its resting state with all three inputs turned off. Then at  $t = 10$  we manually switched on all inputs and the signaling cascade was triggered off showing an expression profile very similar to the Boolean simulation (Figure 3A). We observed a total activation of the four transcription factors at around  $t = 21$  just like in the Boolean simulation. One can clearly see that Fyn and ZAP-70 were activated more slowly. Nonetheless TCR-phos, LAT-phosp, PLCg (act) and Itk were instantly turned on due to the low threshold  $k_{\text{fast}}$ . Only when at around  $t = 25$  the concentrations of Fyn and ZAP-70 were high enough, the feedback loops Fyn  $\rightarrow$  PAG-Csk and ZAP-70  $\rightarrow$  cCbl became active and began to switch on PAG-Csk and cCbl. While PAG-Csk reached a constant medium



**Figure 3**  
**Results from the Boolean and continuous simulations of the T-cell model using the manually determined parameter set.** The vertical dashed lines in (A, B, F) mark (from left to right): the switching on of the inputs, the total activation of the transcription factors, the activation of the feedback loops and the total deactivation of the transcription factors. **(A)** Boolean simulation using synchronous updates. **(B)** Continuous simulation using HillCubes. **(C)** Upper figure: difference between the time course from (A) and the time course from (B). Lower figure: correlation between the discrete and the continuous expression pattern of the lower 29 species at each time point  $0 < t < 100$ . The curve was smoothed using a moving-average filter with 5 time-units window length. **(D)** Ratio of activation and deactivation time,  $\rho$ , for  $\tau_{Fyn} = \tau_{ZAP-70} = 1, 2, \dots, 20$ . Other parameters are set to  $n = 3, k = 0.3, \tau = 1, k_{fast} = 0.1$  and  $k_{slow} = 0.8$ . For  $\tau_{Fyn} = \tau_{ZAP-70} = 1$  the cascade was not activated properly. **(E)** Ratio of activation and deactivation time,  $\rho$ , for 20 different  $k_{fast}$  and  $k_{slow}$ .  $\log_{10}(k_{fast})$  and  $\log_{10}(k_{slow})$  are uniform from  $[-2, 0]$ . Other parameters are set to  $n = 3, k = 0.3, \tau = 1, \tau_{Fyn} = \tau_{ZAP-70} = 10$ . In the white areas the cascade was not activated properly. **(F)** Comparison between time course  $y(t)$  with the manually determined parameters and time course  $y^3(t)$  with the decreased threshold in the activation of LAT-phosph by ZAP-70 as described in the section about monotony properties. Plotted is the difference  $y^3(t) - y(t)$ .

expression level, cCbl was only weakly and transiently expressed. This, however, sufficed to switch off the cascade and ultimately at around  $t = 67$  all transcription factors were again deactivated. In contrast to the Boolean model, the continuous model did not exhibit an oscillatory behavior but reached a steady-state after deactivation of the signaling cascade. The species with long expression periods in the Boolean oscillation (Tcr

bind, PAG Csk, Fyn, TCR phos and Ikb) were expressed at high or medium levels whereas the species with short expression periods were not expressed at all.

**Comparison of the discrete and the continuous T-cell model**

Although one can argue that there is no real time scale in Boolean networks, we compared both models by

subtracting the discrete from the continuous time course (Figure 3C). The activation of the feedback loops in the Boolean model was conveniently chosen such that we have a total deactivation of the transcription factors at around  $t = 67$  in both models. While we observed an almost perfect agreement of the time courses of the three inputs, there was a huge difference in the time courses of the species involved in the two feedback loops. This was not surprising, considering that these loops are regulated differently in both models. The species downstream of the regulatory loops (from LAT phospho to CRE in Figure 3C) showed again a similar expression pattern.

We then analyzed this part of the cascade more deeply. Figure 3C also shows the correlation between the discrete and the continuous expression pattern of the species downstream of the feedback loops at each time point  $0 < t < 100$ . We observed a high correlation in the more stationary phases (resting state, activated state and deactivated state) and a significant drop of correlation during the transitions between these phases. This met our expectation that the two models show the same qualitative but a different dynamic behavior.

#### Ratio of activation and deactivation time

When looking at Figures 3A and 3B, a striking difference between the dynamics of both models is that in the discrete model activation and deactivation took approximately the same time, whereas in the continuous model activation was a much faster process than deactivation. This can also be seen from the red and blue 'steps' in upper Figure 3C during the activation and deactivation phases. To confirm that this was not merely an artefact of our choice of parameters, we calculated and analyzed the ratio between activation and deactivation time for different parameters. We defined

- the beginning of activation  $t_{\text{ActBeg}}$  as the time when the first species (of the lower 29) reaches 5% of its maximal value,
- the end of activation  $t_{\text{ActEnd}}$  as the time when the last species reaches 95% of its maximal value,
- the beginning of deactivation  $t_{\text{DeactBeg}}$  as the time after  $t_{\text{ActEnd}}$  when the first species drops under 95% of its maximal value,
- the end of deactivation  $t_{\text{DeactEnd}}$  as the time when the last species drops under 5% of its maximal value,
- and, finally, the ratio of interest

$$\rho = \frac{t_{\text{ActEnd}} - t_{\text{ActBeg}}}{t_{\text{DeactEnd}} - t_{\text{DeactBeg}}}.$$

In the Boolean model we could easily compute  $\rho_b = 1$  implying equally fast activation and deactivation. In the

continuous model the crucial parameters were the life-times  $\tau_{\text{Fyn}}$  and  $\tau_{\text{ZAP-70}}$  on the one hand and the concentration thresholds  $k_{\text{fast}}$  and  $k_{\text{slow}}$  on the other hand. The remaining parameters were set to  $n = 3$ ,  $k = 0.3$  and  $\tau = 1$ .

First, we computed  $\rho$  for fixed  $k_{\text{fast}}$  and  $k_{\text{slow}}$  and different  $\tau_{\text{Fyn}} = \tau_{\text{ZAP-70}}$ . The result is shown in Figure 3D. For  $\tau_{\text{Fyn}} = \tau_{\text{ZAP-70}} = 1$  the cascade was not activated properly. For larger values we observed a decrease in  $\rho$  implying that an increase of  $\tau_{\text{Fyn}}$  and  $\tau_{\text{ZAP-70}}$  prolonged the deactivation phase. This was to be expected — longer life-times resulted in a lessened increase of the decisive elements Fyn and ZAP-70 in the regulatory loop.

Second, we analyzed the effect of  $k_{\text{fast}}$  and  $k_{\text{slow}}$  for fixed  $\tau_{\text{Fyn}} = \tau_{\text{ZAP-70}} = 10$ . The result is shown in Figure 3E. Only for parameters  $k_{\text{fast}} \ll k_{\text{slow}} < 1$  the cascade was activated properly. This agrees well with the fact that the difference between these two parameters is responsible for the delayed activation of the feedback loops. If it was not big enough the cascade was being deactivated before it had been fully activated. The greater this difference was, i.e. the farther we go away from the diagonal in Figure 3E, the smaller  $\rho$  got, implying that a later activation of the feedback loops prolonged the deactivation phase. However, despite all these influences of the parameters, we observed much smaller ratios  $\rho$  in the continuous model than in the Boolean model. The average  $\rho$ 's in Figures 3D and 3E were  $\bar{\rho}_c^{(1)} = 0.24$  and  $\bar{\rho}_c^{(2)} = 0.27$ , respectively, which were both significantly smaller than  $\rho_b = 1$ . This suggests that  $\rho \ll 1$  may be an invariant of the dynamical system of biological importance.

#### Explanation of experimental data using the continuous T-cell model

The crucial question is if our model can reproduce real data. To test this we used the data set presented in [12] (see Methods). It describes the dynamics of the activation of key signaling elements upon activation of the TCR by three different ligands with varying affinity, Q144, Y144 and L144. We considered, in particular, the ligand L144 for which experiments were performed with three different ligand concentrations. Using global minimization of the model fit with respect to the data, we were able to determine a set of parameters for which our model reasonably reproduced the experimental data (see Methods and Table 1). Figures 4A-D show the corresponding simulated time courses. We see that the model was able to approximate the time courses of ERK and IKK well. The fit of JNK was also acceptable, although the high measured concentrations at time  $t = 0$  constituted a problem, as the model was naturally unable to reproduce them due to the delaying effect of the signaling cascade. In the case of NFAT, the model was



**Table 1: Best fit parameter set for ligand L144**

Species	Input	<i>n</i>	<i>k</i>	Input	<i>n</i>	<i>k</i>	Input	<i>n</i>	<i>k</i>	$\tau$
zap70	tcraphosp	2	0.29	lck	1	0.33	ccbl	1	0.30	2.77
tcraphosp	tcr	1	0.34	lck	1	0.34	fyn	1	0.13	0.03
tcr	TCRlig	1	0.20	ccbl	1	0.41				0.59
slp76	gads	1	0.35							0.24
sek	pkcth	1	0.26							0.01
rsk	erk	<b>3</b>	<b>0.30</b>							<b>1.00</b>
rlk	lck	1	0.36							1.02
rasgrp	pkcth	1	0.40	dag	1	0.39				2.76
ras	rasgrp	3	0.33	grb2sos	17	0.39				7.50
raf	ras	3	0.31							11.80
plcgbind	lat	5	0.36							0.01
plcgact	zap70	1	0.10	slp76	1	0.29	rlk	1	0.24	0.01
	plcgbind	1	0.31	itk	2	0.34				
pkcth	dag	2	0.31							8.66
pagcsk	tcr	1	0.32	fyn	1	0.69				1.33
nfbk	ikb	<b>3</b>	<b>0.30</b>							<b>1.00</b>
nfat	calcen	1	0.37							5.03
mek	raf	1	0.28							0.01
TCRlig	ext. sig.	1	0.12							1.28
lck	pagcsk	2	0.30	cd4	2	0.37	cd45	1	0.33	0.15
lat	zap70	3	0.14							0.01
jun	jnk	<b>3</b>	<b>0.30</b>							<b>1.00</b>
jnk	sek	4	0.57							0.08
itk	zap70	2	0.10	slp76	1	0.31				0.01
ip3	plcgact	2	0.38							0.01
ikkbeta	pkcth	18	0.33							10.55
ikb	ikkbeta	<b>3</b>	<b>0.30</b>							<b>1.00</b>
grb2sos	lat	8	0.46							5.26
gads	lat	1	0.36							0.12
fyn	tcr	6	0.27	lck	2	0.30	cd45	4	0.37	0.46
fos	erk	<b>3</b>	<b>0.30</b>							<b>1.00</b>
erk	mek	2	0.28							0.10
dag	plcgact	3	0.28							0.01
creb	rsk	<b>3</b>	<b>0.30</b>							<b>1.00</b>

**Table 1: Best fit parameter set for ligand L144** (Continued)

cre	creb	<b>3 0.30</b>			<b>1.00</b>
cd4	ext. sig.	0.32			2.25
cd45	ext. sig.	0.35			1.64
ccbl	zap70	0.70			0.29
calcen	ca	0.30			4.53
ca	ip3	3 0.33			0.20
ap1	jun	<b>3 0.30</b>	fos	<b>3 0.30</b>	<b>1.00</b>

The bold entries are the parameters downstream of all measured species. They were not affected by the optimization process.

unable to reproduce the non-monotone dependence on the ligand concentration. This suggests that the network structure cannot be reconciled with the non-monotone dependence of the two key signaling molecules JNK and especially NFAT with respect to the ligand concentration. A non-monotone response of T-cell signals has been reported in other contexts [18], and is consistent with the role of T-cells: their response has to be exquisitely regulated so as to reply only to a particular stimulus; an uncoupled response between the JNK (and p38) and ERK MAP Kinases has also been observed [19]. The regulation of NFAT and JNK, and more generally of TCR-induced signaling, is complex and not yet fully understood. To mention just two examples, processes such as Ras localization [20] have been shown to play a key role, but are not included in the model, and the regulation of calcium, which governs NFAT behavior, is more complex than described in the model. It is out of the scope of this paper to investigate this intriguing behavior in detail, but this result illustrates the power of our approach to gain new biological insight: taking exactly the same knowledge encoded in the discrete model and fitting it to quantitative data, we were able to identify the incompleteness of our model in an aspect that we could not have explored with a discrete model.

#### Best fit parameter set

Subsequently, we analyzed the distribution of the Hill parameters within the best fit parameter set for ligand L144 (Table 1). When looking at the distribution of the exponents  $n$  upstream of the measured species (Figure 4F) we found that after the optimization 36 out of 49 (73%) were below the ad hoc estimate (3) and only 8 (16%) were above it. This was to be expected, as in order to fit three different concentration levels the model had to contain mainly slow switches (low  $n$ ) and only few Boolean-like switches ( $n \rightarrow \infty$ ). Figure 4E shows the distribution of the threshold parameters  $k$  upstream of the measured species. As explained above, we had manually set these parameters to 0.3, with the exception

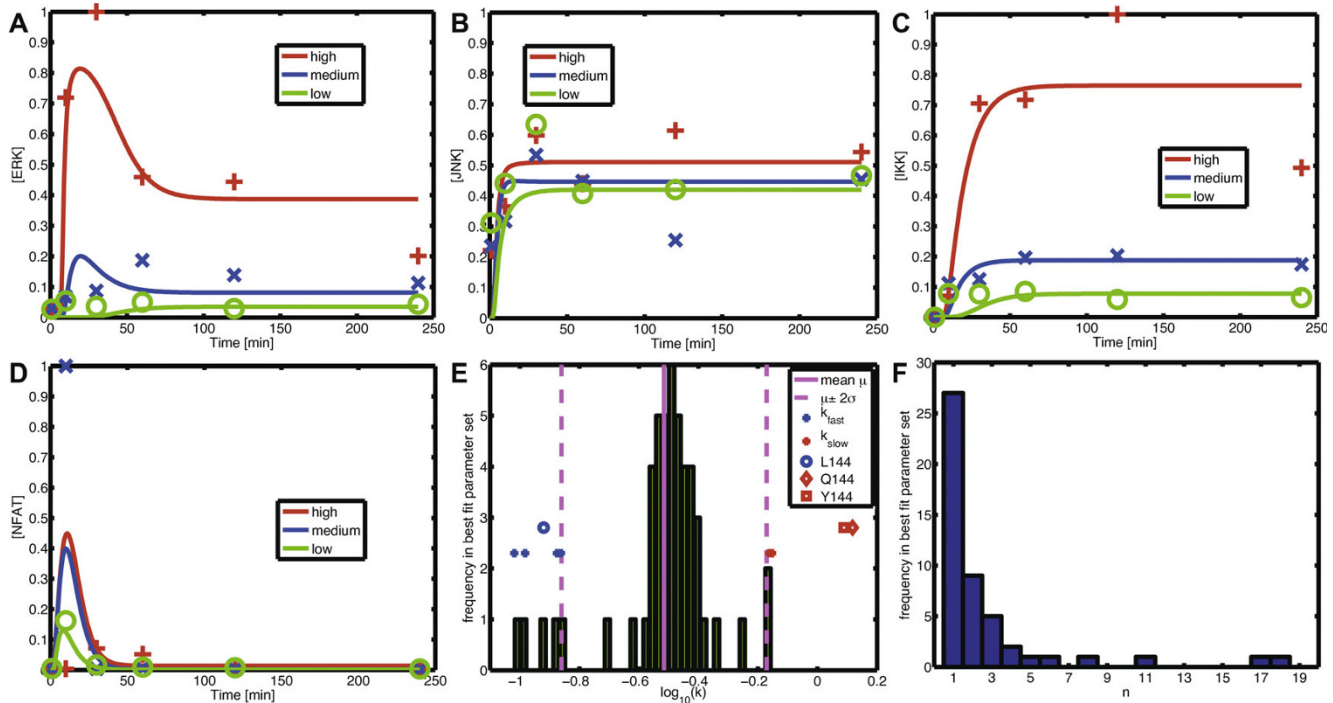
of four  $k$ 's which had been set to  $k_{\text{slow}} = 0.8$  and two  $k$ 's which had been set to  $k_{\text{fast}} = 0.1$ . Interestingly, this structure had been preserved during the optimization. The mean of the thresholds  $k$  was 0.298 and hence well agreed with the ad hoc estimate. Also the high and low thresholds were still at least two standard deviations away from the mean, cf. the red and blue markers '+' in Figure 4E. Only one other parameter  $k$  also had a Z-score above 2: the threshold  $k_{L144}$  for the stimulation of the TCR by the ligand L144, cf. marker 'o'. Possible implications hereof are discussed below.

Due to the large number of parameters the obtained parameter set was, of course, far from being unique. But we showed that a continuous model inferred from a Boolean model is able to reproduce experimental data in a quantitative way. Moreover, this transformation could enhance the explanatory power of the model in the sense that it was enabled to differentiate between more than two states.

In our example, the threshold parameters  $k$  are rather tightly centered around their mean. In principle, however, we could also have extreme outliers, i.e. very large or very small ( $\approx 0$ ) values in the distribution. Mapped back to the Boolean model, this would imply a change of the network topology, as the corresponding reactions are then quasi-constant, either 'off' (for very large  $k$ ) or 'on' (for very small  $k$ ). Thus, a fitting of the continuous model to experimental data may also yield information about the network structure of the Boolean model.

#### Prediction of binding affinities of different ligands

As already mentioned, the fitted threshold  $k_{L144}$  for the stimulation of the TCR by the ligand L144 was significantly below the mean of the other thresholds  $k$ , cf. marker 'o' in Figure 4E. This gave rise to the question of which affinities the model predicts for the other two ligands. To this end, we fitted the affinities of Q144 and Y144, mapped to the inverse of the Hill function's



**Figure 4**  
**Results of the parameter fit. (A-D)** Simulation of the continuous model (solid lines) for high (red), medium (blue) and low (green) concentrations of ligand L144 and experimentally measured concentrations, cf. markers '+', 'x' and 'o'. **(A)** ERK. **(B)** JNK. **(C)** NFAT. **(D)** IKK. **(E, F)** Distribution of the Hill parameters in the best fit parameter set for ligand L144 (Table 1). **(E)** Distribution of the thresholds  $k$ . The markers '+' indicate the position of the parameters which we set to  $k_{fast}$  (blue) and  $k_{slow}$  (red) at the beginning of the optimization. The marker 'o' indicates the position of  $k_{L144}$ ; for comparability the thresholds  $k_{Q144}$  and  $k_{Y144}$  are also indicated, cf. the markers '◇' and '□', respectively. **(F)** Distribution of the Hill exponents  $n$ .

threshold parameters  $k_{Q144}$  and  $k_{Y144}$ , respectively, keeping the rest of the parameters constant. As expected, the fits themselves were far from perfect, due to parameter indeterminacies. Surprisingly however, the values of  $k_{Q144}$  and  $k_{Y144}$  we obtained were significantly above the mean of the other thresholds  $k$ , cf. markers '◇' and '□' in Figure 4E. This suggests that the predicted relation between the parameters  $k_{L144} \ll k_{Y144} < k_{Q144}$  is not simply an artefact of the optimization process. And indeed, it agrees well with experimental data [21].

**Discussion**

We now further discuss the presented transformation method and compare it to various other approaches. Also the relation between discrete and continuous models is discussed, especially with respect to their steady-state behavior and monotony properties.

**Comparison of different transformation approaches**

The relation between discrete and continuous models has already been investigated and various approaches to the problem of constructing the continuous homologues

of the Boolean update rules have been proposed. In the following we shortly review previous work and compare the different approaches.

*Piecewise linear differential equations*

The idea to compare continuous and discrete models is almost as old as Boolean modeling itself. In 1973, Glass et al. [11] studied the relation between discrete models and ODE models of the form (2). Their motivation, however, was quite the opposite of ours. While we intend to enrich the dynamic behavior of discrete models, Glass et al. wanted to investigate the qualitative properties of continuous networks by studying corresponding simpler discrete models. They propose Hill functions as a suitable continuous homologue of one-variable Boolean step functions. In the case of multi-variable Boolean functions  $B_i$  a (perfect) continuous homologue  $\bar{B}_i$  is constructed as follows: Note that, when building a Boolean model, one implicitly introduces a threshold  $0 < \theta_i < 1$  for each species  $X_i$  and defines its state as 'on' if its concentration is above this threshold. The hyperplanes  $\bar{x}_i = \theta_i$ ,  $i = 1, 2, \dots, N$

decompose the cubes  $[0,1]^{N_i}$  into  $2^{N_i}$  rectangular regions called *domains*. Each of these domains contains exactly one vertex  $a \in \{0,1\}^{N_i}$  and is denoted by  $D_i^a$  accordingly (see Additional data file 4). A simple way of defining the functions  $B_i$  is now to set

$$\bar{B}_i^{PW}(\bar{x}_{i1}, \bar{x}_{i2}, \dots, \bar{x}_{iN_i}) = B_i(a), \text{ where } (\bar{x}_{i1}, \bar{x}_{i2}, \dots, \bar{x}_{iN_i}) \in D_i^a. \tag{6}$$

With this definition model (2) is a so-called *piecewise linear ODE model* which means that within each domain  $D_i^a$  equation (2) is a linear ODE of the form  $\dot{\bar{x}}_i = (\alpha_i - \bar{x}_i) / \tau_i$ . This kind of equation is very well understood and can be solved analytically. The functions  $\bar{B}_i^{PW}$  defined in (6) perfectly agree with  $B_i$  on the vertices of the unit cube. Using piecewise linear ODEs Glass et al. could prove some theoretical results on the relation between discrete and continuous models, e.g. that Boolean steady-states are also steady-states of the continuous model. Some of these results are restated and generalized in the Methods section. Piecewise linear models are typically not used for quantitative simulations, as the step-like transition between the different domains is often unrealistic. Rather they are analyzed in a qualitative and semi-qualitative way, where their trajectories between the different domains are treated analogously to the state transition graphs of Boolean models [22].

**Fuzzy logic**

Another well studied way of generalizing Boolean models is *fuzzy logic* [23]. Recall that in a Boolean model one defines the state of a species as ‘on’ if its concentration is above a certain threshold. In fuzzy logic this concept is relaxed and a so-called *degree of membership* (DOM) function  $0 \leq \mu_{X_i}(\bar{x}_i) \leq 1$  is introduced for each species  $X_i$ . For concentrations  $0 = \bar{x}_i = 1$  this function gives the degree with which we say that  $X_i$  is ‘on’. There are two standard ways of generalizing the Boolean operators AND, OR, NOT:

(i) min-max logic

$$\begin{aligned} x_i \wedge x_j &\mapsto \min(\mu_{X_i}(\bar{x}_i), \mu_{X_j}(\bar{x}_j)) \\ x_i \vee x_j &\mapsto \max(\mu_{X_i}(\bar{x}_i), \mu_{X_j}(\bar{x}_j)) \\ \neg x_i &\mapsto 1 - \mu_{X_i}(\bar{x}_i) \end{aligned} \tag{7}$$

(ii) product-sum logic

$$\begin{aligned} x_i \wedge x_j &\mapsto \mu_{X_i}(\bar{x}_i) \cdot \mu_{X_j}(\bar{x}_j) \\ x_i \vee x_j &\mapsto \mu_{X_i}(\bar{x}_i) + \mu_{X_j}(\bar{x}_j) \\ \neg x_i &\mapsto 1 - \mu_{X_i}(\bar{x}_i) \end{aligned} \tag{8}$$

When using product-sum logic, a normalization to the unit interval is necessary, since  $\mu_{X_i}(\bar{x}_i) + \mu_{X_j}(\bar{x}_j)$  can assume values greater than 1. Both, (i) as well as (ii) are ways to construct the functions  $\bar{B}_i$  from the Boolean functions  $B_i$ . If the DOM functions satisfy  $\mu(0) = 0$  and  $\mu(1) = 1$ , the functions  $\bar{B}_i^{MM}$  obtained by min-max logic agree with  $B_i$  on the vertices of the unit cube and hence the steady-states of the Boolean model are also steady-states of the continuous model. The major drawback of this method is that the functions  $\bar{B}_i^{MM}$  are in general not differentiable, and do not have a ‘nice’ analytic representation (Figure 5B). Hence most analysis methods are not applicable to the resulting ODE systems. When we use product-sum logic, we encounter the problem that the resulting functions  $\bar{B}_i^{PS}$  do not necessarily agree with  $B_i$  (Figure 5C).

**Standardized qualitative dynamical systems**

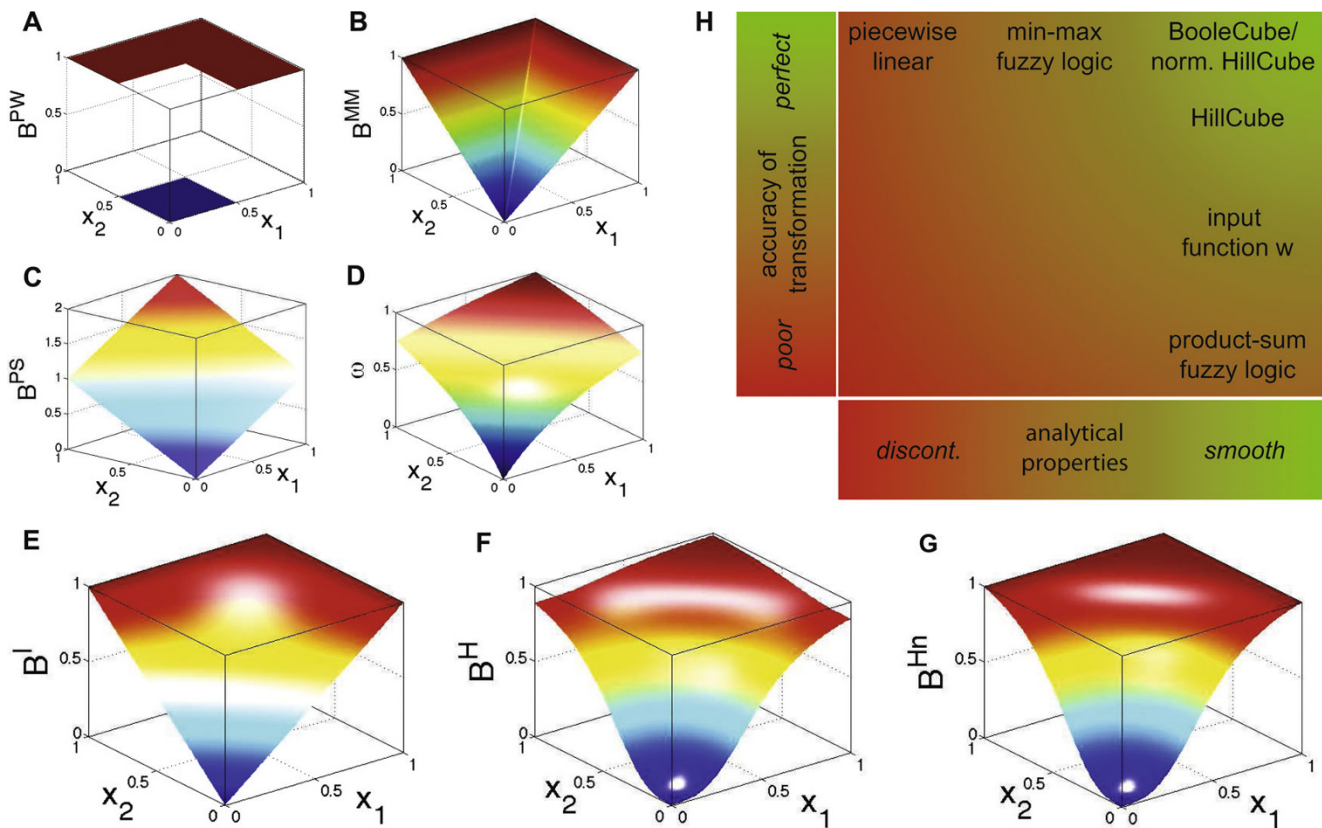
Mendoza et al. [10] put forward a method to transform a Boolean model into a system of ODEs similar to (2) called *standardized qualitative dynamical system*. Their approach, however, is applicable only to a subclass of Boolean models: For each species  $X_i$  we have a set of activators  $A_i$  and a set of inhibitors  $I_i$  of  $X_i$ . Then  $x_i$  is set to 1 at the next time step if at this time any of its activators and none of its inhibitors are acting upon it, otherwise  $x_i$  is set to 0. This corresponds to the Boolean logic  $(\bigvee_{a \in A_i} a) \wedge \neg(\bigvee_{r \in I_i} r)$ . Clearly, many logic functions, like XOR gates for example, cannot be represented that way. From this Boolean model a continuous model is built up consisting of ODEs of the form

$$\dot{\bar{x}}_i = \frac{-e^{0.5h} + e^{-h(\omega_i - 0.5)}}{(1 - e^{0.5h})(1 + e^{-h(\omega_i - 0.5)})} - \gamma_i \bar{x}_i,$$

Where

$$\omega_i = \frac{1 + |A_i|}{|A_i|} \frac{\sum_{X_j \in A_i} \bar{x}_j}{\sum_{X_j \in A_i} \bar{x}_j} \left( 1 - \frac{1 + |I_i|}{|I_i|} \frac{\sum_{X_j \in I_i} \bar{x}_j}{1 + \sum_{X_j \in I_i} \bar{x}_j} \right). \tag{9}$$

The right-hand side of the above ODE consists of two parts: an activation function and a term for decay as in (2). The activation is given by a sigmoid shaped function of  $\omega_i$ , where  $\omega_i$  represents the total input to node  $X_i$ . The steepness of the activation function is determined by the parameter  $h$ . Decay is assumed to be proportional to  $\bar{x}_i$ . Actually, a more general form of  $\omega_i$  is introduced in [10], where the influence of the activators and inhibitors can be differently weighted. For the sake of better comparability we set all these weights equally to 1, as suggested in [10]. The more activators and the less inhibitors of a node are 1, the greater  $\omega_i$  is. It takes its minimum (0), iff



**Figure 5**  
**Comparison of different transformation techniques.** Different continuous homologues of a Boolean OR gate. **(A)** Piecewise linear function  $\bar{B}^{PW}$ . **(B)** Function  $\bar{B}^{MM}$  obtained by min-max fuzzy logic and linear DOM functions. **(C)** Function  $\bar{B}^{PS}$  obtained by product-sum fuzzy logic and linear DOM functions. **(D)** Input function  $\omega$  introduced by Mendoza et al. [10]. **(E)** BooleCube  $\bar{B}^I$  obtained by multivariate polynomial interpolation. **(F)** HillCube  $\bar{B}^H$ . **(G)** normalized HillCube  $\bar{B}^{Hn}$ . In the last two figures parameters  $n = 3$  and  $k = 0.5$  were chosen for both inputs. **(H)** Overview of the different transformation techniques with respect to their analytical properties and transformation accuracy.

all activators are 0 or all inhibitors are 1, and its maximum (1), iff all activators are 1 and all inhibitors are 0. This, however, does not exactly correspond to the Boolean rule from above. In consequence, steady-states of the Boolean model are not necessarily steady-states of the continuous model. This will be further discussed below.

*Multivariate polynomial interpolation*

The aforementioned attempts only lead to functions which are either not differentiable or do not precisely generalize the Boolean logic. A straightforward approach to eliminate these drawbacks is to use multivariate polynomial interpolation [13] for the construction of the functions  $\bar{B}_i$  (see Methods). This technique can be applied to any Boolean function  $B_i$ . The resulting BooleCubes  $\bar{B}_i^I$  are smooth and can easily be analytically differentiated and integrated. They agree with the Boolean functions  $B_i$  on the vertices of the unit cube

and hence the steady-states from the Boolean model are also steady-states in the continuous model.

We define Hill functions  $f_{ij}$  for all interactions and consider the HillCubes  $\bar{B}_i^H$  from equation (4). The idea behind this is that each interaction is described by its own Hill function with specific parameters and the different interactions are coupled by the BooleCubes. Since they are affine multilinear, the latter preserve the shape of the individual Hill functions. Thus, we can mimic single-component non-linearities which are common in switch-like regulatory systems. Additional data file 3 shows the HillCubes derived from all 16 two-variable Boolean gates. A mathematically rigorous treatment of this kind of dynamical systems can be found in [14]. If the Hill coefficient  $n$  goes to infinity the Hill function becomes more and more like a Boolean step function (see Additional data file 2). Hence for large exponents the

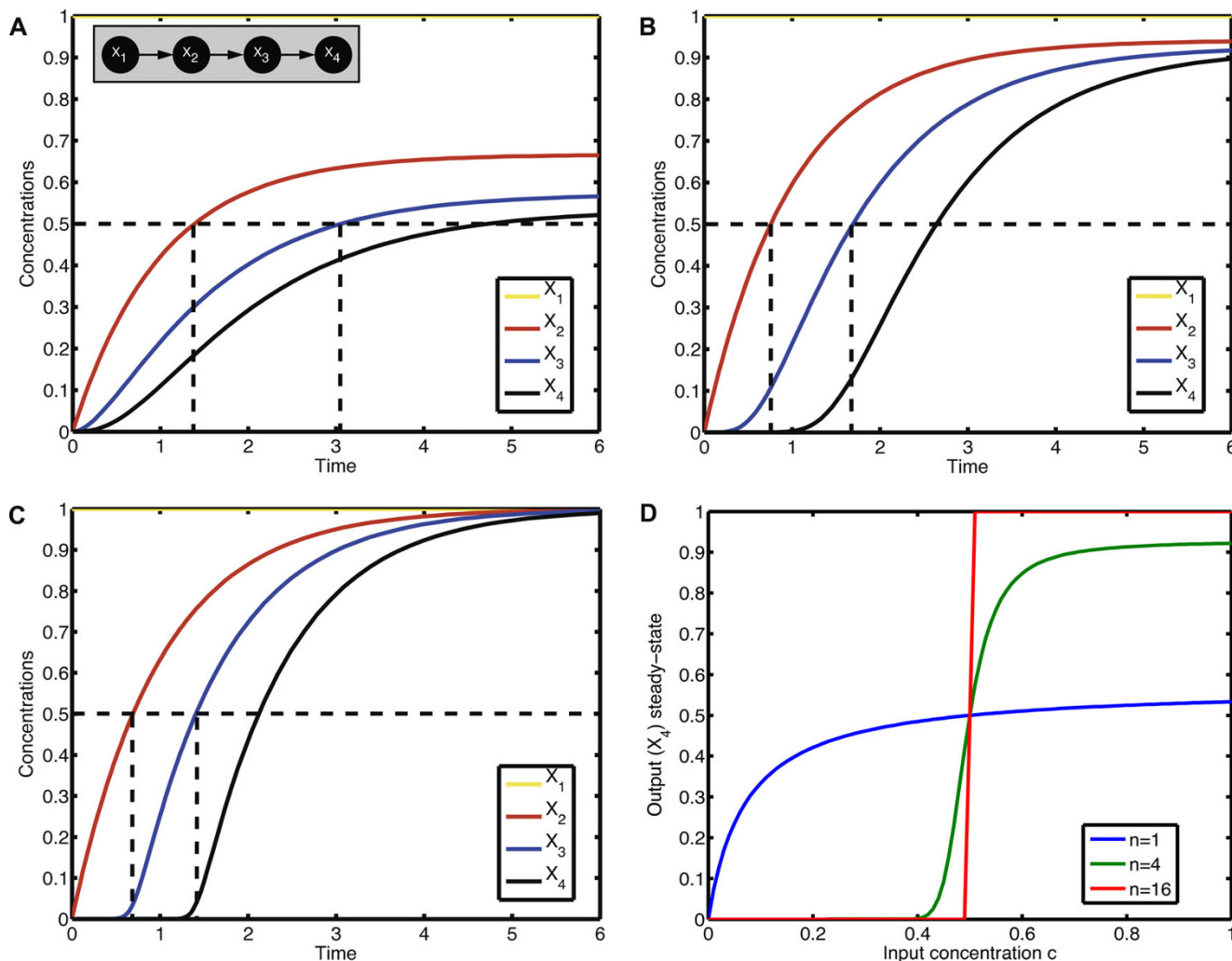
HillCubes are very similar to the Boolean functions  $B_i$  and the continuous system will likely show an almost Boolean behavior. This is further illustrated in Figure 6.

The HillCubes do not perfectly agree with the Boolean update functions due to the asymptotic behavior of the Hill functions. By a suitable choice of the Hill parameters the difference can be reduced but not fully eliminated. An

easy way to achieve a perfect agreement is to normalize the Hill functions to the unit interval, as is done in the normalized HillCubes  $\bar{B}_i^{\text{Hn}}$  from equation (5).

**Comparison**

To conclude, we illustrate the above methods applied to a simple OR gate between two species  $X_1$  and  $X_2$ . We compute



**Figure 6**

**Effect of increasing Hill exponents.** We consider a simple cascade between the four species  $X_1, X_2, X_3, X_4$  as shown in the inset in (A). Each activation is modeled using a Hill function with threshold  $k = 0.5$  and Hill coefficient  $n$ . The life-times  $\tau_i$  are set to 1. As initial conditions we take  $\bar{x}_1 = c > 0, \bar{x}_2 = 0, \bar{x}_3 = 0, \bar{x}_4 = 0$ , for some constant input concentration  $c$ . The input node  $X_1$  remains constant and the other concentrations  $\bar{x}_i$  change accordingly to the ODE  $\dot{\bar{x}}_i = \bar{x}_{i-1}^n / (\bar{x}_{i-1}^n + k^n) - \bar{x}_i, i = 2, 3, 4$ . We simulate the model for different Hill coefficients  $n = 1, 4, 16$  and input level  $c = 1$ ; the results are shown in (A), (B) and (C). All three time courses show qualitatively the same cascade-like pattern. With growing  $n$ , however, the onset of activation of  $X_3$  and  $X_4$  comes closer and closer to the time point at which their activators  $X_2$  and  $X_3$ , respectively, cross the threshold  $k$ . (D) shows the input-output curve. Plotted is the (constant) input concentration  $c$  of node  $X_1$  against the steady-state concentration of node  $X_4$ . For  $n > 1$ , we observe the typical sigmoid stimulus-response behavior of signaling cascades, see e.g. [28]. With increasing  $n$  the steepness of the input-output curve increases, leading to an almost discrete (Boolean) output in the case  $n = 16$ .

- the piecewise linear function

$$\bar{B}^{\text{PW}} = \begin{cases} 0 & \text{where } \bar{x}_1 < \theta_1 := 0.5 \text{ and } \bar{x}_2 < \theta_2 := 0.5 \\ 1 & \text{elsewhere} \end{cases}$$

from equation (6),

- $\bar{B}^{\text{MM}}(\bar{x}_1, \bar{x}_2) = \max(\bar{x}_1, \bar{x}_2)$  obtained by fuzzy logic (with linear DOM functions) following (7),
- $\bar{B}^{\text{PS}}(\bar{x}_1, \bar{x}_2) = (\bar{x}_1 + \bar{x}_2)$  obtained by fuzzy logic (with linear DOM functions) following (8),
- the input function from equation (9) introduced by Mendoza et al.

$$\omega(\bar{x}_1, \bar{x}_2) = \frac{3}{2} \frac{\bar{x}_1 + \bar{x}_2}{1 + \bar{x}_1 + \bar{x}_2},$$

- the BooleCube  $\bar{B}^{\text{I}}(\bar{x}_1, \bar{x}_2) = -\bar{x}_1\bar{x}_2 + \bar{x}_1 + \bar{x}_2$  from equation (3) obtained by the interpolation technique,
- the HillCube  $\bar{B}^{\text{H}}(\bar{x}_1, \bar{x}_2) = \bar{B}^{\text{I}}(f_1(\bar{x}_1), f_2(\bar{x}_2))$  from equation (4) for Hill functions  $f_1$  and  $f_2$  with parameters  $n = 3, k = 0.5$ ,
- and, finally, the normalization

$$\bar{B}^{\text{Hn}}(\bar{x}_1, \bar{x}_2) := \bar{B}^{\text{I}}\left(\frac{f_1(\bar{x}_1)}{f_1(1)}, \frac{f_2(\bar{x}_2)}{f_2(1)}\right)$$

of  $\bar{B}^{\text{H}}$  from equation (5).

Figures 5C and 5D show the product-sum fuzzy logic function  $\bar{B}^{\text{PS}}$  and the input function  $\omega$ . One can clearly see that they do not represent a pure OR gate, where the values at  $(x_1, x_2) = (1, 0)$  and  $(x_1, x_2) = (0, 1)$  should already be maximal. This is the case in Figures 5A and 5B which show the piecewise linear  $\bar{B}^{\text{PW}}$  and the min-max fuzzy logic function  $\bar{B}^{\text{MM}}$ . Here however, the problem is that the functions are not differentiable, as can easily be seen from their plots. The BooleCube  $\bar{B}^{\text{I}}$  shown in Figure 5E is both, smooth and maximal as soon as any concentration is equal to 1. Finally, Figures 5F and 5G show the (normalized) HillCubes  $\bar{B}^{\text{H}}$  and  $\bar{B}^{\text{Hn}}$ , respectively, which are also smooth and can be considered good transformations of the Boolean OR gate. An overview about the discussed advantages and disadvantages of the different transformation techniques is provided in Figure 5H.

### Theoretical results about steady-states

A fundamental principle of biological modeling is that steady-states of a model typically correspond to the different operating modes or states of the biological system under study. This correspondence was also the motivation for Kauffman's seminal study [1], where Boolean models were introduced for the first time in biology. A critical step in the justification of any

transformation method therefore is to ensure that at least the steady-states of the Boolean model are still steady-states in the homologue continuous system. In the case of a perfect agreement of the Boolean update rules  $B_i$  with their continuous homologues  $\bar{B}_i$  this can easily be shown (see Methods and [11]). This perfect agreement, however, is not a biologically plausible assumption; biological interactions, such as enzyme kinetics for example, are known to asymptotically approach but never fully reach a saturation level. Empiric evidence that also in real-world examples, the steady-states of a Boolean model correspond to steady-states of a homologue continuous model, is given by Mendoza et al. [10]. The method of transformation used therein has already been described above and we also mentioned that it does not accurately transform the Boolean update rule into a continuous activation function. This inaccuracy is due to a systematic difference between the Boolean logic and the analytic form of the activation function. It is not the result of an asymptotic sigmoid function; in fact, the used sigmoid function assumes its maximal values 0 and 1. One can easily construct an example where due to this systematic difference Boolean steady-states are not conserved under the transformation (see Additional data file 5).

In the case of the HillCube model, it is the other way round. There is no systematic difference between the Boolean update rules and the HillCube functions, the imperfect agreement is caused by the asymptotic behavior of the Hill functions. Therefore, the difference between both can be made arbitrarily small — albeit not zero — by a suitable choice of parameters. In this situation, we can show that for certain parameters, more precisely for sufficiently large exponents, there will be a steady-state of the continuous system in the neighborhood of each Boolean steady-state (see Methods). This theoretical result further justifies the presented transformation method.

### Monotony properties

A nice feature of our method for converting Boolean into continuous models is that monotony properties, typically captured in the underlying interaction graph of the system, are preserved. Interaction graphs are signed directed graphs where each directed edge reflects a causal dependency, which can either be positive or negative, between its start and end node. Boolean models represented as interaction hypergraphs have a unique underlying interaction graph which can easily be derived from the logical model (by splitting the ANDs, see Klamt et al. [6]). For example, the Boolean function  $A = (\neg B \wedge C) \vee D$  would be translated into two positive arcs ( $C \rightarrow A, D \rightarrow A$ ) and one negative arc ( $B \dashv A$ ). In the interaction

graph one may then compute the recently introduced dependency matrix [6,24], which determines for each pair (X, Y) of species the global effect of X on Y. This effect can — in some cases only initially — be positive, negative, ambivalent or vanishing.

For example, the dependency matrix of the T-cell model tells us, that LAT-phosp exerts purely positive effects on ERK, JNK, IKK, and NFAT, because there are only positive paths from LAT-phosp to these species and no negative feedback is involved. If we simulate a scenario with the logical model and repeat it then with e.g. fixing LAT-phosp to 1 (i.e. to the highest possible value), the resulting Boolean values in the four species mentioned above cannot decrease. In continuous systems, the interaction graph is encoded in the sign structure of the Jacobian matrix. In fact, in a continuous system obtained from a Boolean model the interaction graph is up to negative self-loops identical and monotony properties are therefore preserved. Accordingly, a positive perturbation in LAT-phosp, e.g. by permanently decreasing the threshold of ZAP-70 in the interaction activating LAT-phosp, results in a trajectory that is always above the trajectory of the non-perturbed system (Figure 3F). In fact, in accordance with the interaction graph of the Boolean model, we observe purely positive effects on all species downstream of LAT-phosp with the exception of ikB. Hence, important qualitative properties of the dynamics derived from the logical model are reflected in the dynamics of the continuous system.

### Conclusion

With increasing amounts of quantitative data being available, the challenge arises how we can use our typically qualitative knowledge about biological systems to explain this data. For this purpose, we presented a canonical and fully standardized way of transforming qualitative discrete into continuous models. The transformation is accurate and we can show that it preserves the steady-state behavior as well as the monotony properties of the discrete model. The feasibility of the presented approach was substantiated by applying it to a logical model of T-cell receptor signaling. The resulting model is an extensive continuous model of T-cell activation. In contrast to the Boolean model it allowed to accommodate different time scales by adjusting kinetic parameters. It was competent to reproduce time courses of key signaling molecules measured for three different ligand concentration levels. Moreover, the model was able to predict the binding affinities of different ligands.

Being fully automatized [Krumsiek et al.: Odefy — From discrete to continuous models. In preparation (2009)]

the presented method recommends itself to be applied to further biological systems. Future work could also aim at generalizing the approach from Boolean (binary) to *s*-state systems, where one no longer differentiates between two but *s* > 2 discrete states, e.g. ‘low’, ‘medium’ and ‘high’. Finally, the relation between a discrete model and its continuous homologue needs to be further investigated, especially with respect to more complex behaviors like oscillations, which are of importance in many biological systems, such as cell cycle.

### Methods

#### Multivariate polynomial interpolation

We now explain the technique of multivariate polynomial interpolation of a single Boolean function  $B_i$ . Therefore  $i \in \{1, 2, \dots, N\}$  is fixed and for the sake of simplicity the subscript *i* is omitted. We remark that here  $B$  can be any real-valued function on the vertices of the unit cube  $\{0, 1\}^N$ , i.e. does not necessarily have to be a Boolean function. The idea is, to find a polynomial  $\bar{B}^1: \mathbb{R}^N \rightarrow \mathbb{R}$  that is a continuation of  $B_i$  in the sense that  $\bar{B}^1(\bar{x}_1, \bar{x}_2, \dots, \bar{x}_N) = B(\bar{x}_1, \bar{x}_2, \dots, \bar{x}_N)$  for all  $\bar{x}_1, \bar{x}_2, \dots, \bar{x}_N \in \{0, 1\}$ .

One can easily see, that there is no unique solution to this problem. Therefore, we additionally require that the degree of the polynomial be minimal, where the degree of some polynomial

$$f(\gamma_1, \gamma_2, \dots, \gamma_N) = \sum_{(m_1, m_2, \dots, m_N) \in \mathbb{N}^N} a_{m_1 m_2 \dots m_N} \gamma_1^{m_1} \gamma_2^{m_2} \dots \gamma_N^{m_N},$$

$a_{m_1 m_2 \dots m_N} = 0$  for almost all  $(m_1, m_2, \dots, m_N) \in \mathbb{N}^N$ , is defined as

$$\text{deg}(f) = \max_{\substack{(m_1, m_2, \dots, m_N) \in \mathbb{N}^N \\ a_{m_1 m_2 \dots m_N} \neq 0}} \sum_{i=1}^N m_i.$$

For  $\bar{x}_1, \bar{x}_2, \dots, \bar{x}_N \in \mathbb{R}$  we define

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

We now show that  $\bar{B}^1$  indeed satisfies the three requirements for interpolation functions that we set out at the beginning (see section on continuous homologues of Boolean functions).

**Theorem.** *The function  $\bar{B}^1$  has the following properties:*

$$(i) \quad \bar{B}^1(\bar{x}_1, \bar{x}_2, \dots, \bar{x}_N) = B(\bar{x}_1, \bar{x}_2, \dots, \bar{x}_N) \text{ for all } \bar{x}_1, \bar{x}_2, \dots, \bar{x}_N \in \{0, 1\}.$$



(ii)  $\bar{B}^1$  is the unique minimal degree polynomial interpolating  $B$ .

(iii) Let  $s$  denote the number of symmetry hyperplanes of  $B$ , i.e. the number of variables  $x_i$  satisfying

$$B(x_1, \dots, x_{i-1}, 0, x_{i+1}, \dots, x_N) = B(x_1, \dots, x_{i-1}, 1, x_{i+1}, \dots, x_N), \text{ for all } (x_1, \dots, \hat{x}_i, \dots, x_N) \in \{0, 1\}^{N-1}. \text{ Then } \deg(\bar{B}^1) = N - s.$$

(iv)  $\bar{B}^1$  is affine multilinear. It is multilinear iff it corresponds to an AND gate, i.e.

$$B(x_1, x_2, \dots, x_N) = \bigwedge_{i=1}^N x_i.$$

(v) Let  $0 \leq \bar{x}_1, \bar{x}_2, \dots, \bar{x}_N \leq 1$ . Then it holds  $\min_{\{0,1\}^N} B \leq \bar{B}^1(\bar{x}_1, \bar{x}_2, \dots, \bar{x}_N) \leq \max_{\{0,1\}^N} B$ .

In particular,  $\bar{B}^1(\bar{x}_1, \bar{x}_2, \dots, \bar{x}_N) \in [0, 1]$  if  $B$  is a Boolean function.

Proof. (i) Note that for  $\bar{x}_i \in \{0, 1\}$  we have

$$x_i \bar{x}_i + (1 - x_i)(1 - \bar{x}_i) = \begin{cases} 1 & \text{if } \bar{x}_i = x_i, \\ 0 & \text{otherwise.} \end{cases}$$

(ii) A minimal degree interpolation polynomial is of the form

$$f(\bar{x}_1, \bar{x}_2, \dots, \bar{x}_N) = \sum_{(m_1, m_2, \dots, m_N) \in \{0,1\}^N} a_{m_1, m_2, \dots, m_N} \bar{x}_1^{m_1} \bar{x}_2^{m_2} \dots \bar{x}_N^{m_N} \quad (*)$$

since exponents greater than 1 can be replaced by 1 without changing the values of  $f$  for  $\bar{x}_1, \bar{x}_2, \dots, \bar{x}_N \in \{0, 1\}$ . We order the vertices of the unit cube such that the number of 1's in the coordinates is not decreasing and denote the reordered sequence by  $(V_k)_{k=1}^{2^N}$ . Then we consider the sequence of equations  $f(V_k) = B(V_k)$ ,  $k = 1, 2, \dots, 2^N$ . For all  $k \in \{1, 2, \dots, 2^N\}$  there is exactly one coefficient  $a_{m_1, m_2, \dots, m_N}$  whose monomial  $g(\bar{x}_1, \bar{x}_2, \dots, \bar{x}_N) = \bar{x}_1^{m_1} \bar{x}_2^{m_2} \dots \bar{x}_N^{m_N}$  satisfies  $g(V_k) = 0$ ,  $k = 1, 2, \dots, k - 1$  and  $g(V_k) = 1$ . This allows to uniquely determine the coefficients  $a_{m_1, m_2, \dots, m_N}$  and that way also the polynomial  $f$ . Since  $\bar{B}^1$  is of the form  $(*)$ , it is the unique minimal degree interpolation polynomial.

(iii) The degree of  $\bar{B}^1$  is clearly  $\deg(\bar{B}^1) \leq N$ . If  $B$  is symmetric with respect to the hyperplane  $x_i = 0.5$ , w.l.o.g.  $i = 1$ , we have  $B(0, x_2, \dots, x_N) = B(1, x_2, \dots, x_N)$  for all  $(x_2, \dots, x_N) \in \{0, 1\}^{N-1}$  and consequently

$$\begin{aligned} & \bar{B}^1(\bar{x}_1, \bar{x}_2, \dots, \bar{x}_N) \\ &= (1 - \bar{x}_1) \sum_{x_2=0}^1 \dots \sum_{x_N=0}^1 B(0, x_2, \dots, x_N) \prod_{i=2}^N (x_i \bar{x}_i + (1 - x_i)(1 - \bar{x}_i)) + \\ & \bar{x}_1 \sum_{x_2=0}^1 \dots \sum_{x_N=0}^1 B(1, x_2, \dots, x_N) \prod_{i=2}^N (x_i \bar{x}_i + (1 - x_i)(1 - \bar{x}_i)) \\ &= \sum_{x_2=0}^1 \dots \sum_{x_N=0}^1 B(\bullet, x_2, \dots, x_N) \prod_{i=2}^N (x_i \bar{x}_i + (1 - x_i)(1 - \bar{x}_i)). \end{aligned}$$

Hence,  $\deg(\bar{B}^1) \leq N - 1$ . Inductively this proves (iii).

(iv) Let  $i \in \{1, 2, \dots, N\}$ . Then at fixed  $\bar{x}_1, \dots, \bar{x}_{i-1}, \dots, \bar{x}_N$  the derivative  $\bar{x}_i \mapsto \partial \bar{B}^1 / \partial \bar{x}_i$  is constant so  $\bar{B}^1$  is affine linear in  $\bar{x}_i$ . Moreover, we have

$$\bar{B}^1 \text{ comes from an AND gate. } \Leftrightarrow \bar{B}^1 = \prod_{i=1}^N \bar{x}_i \Leftrightarrow \bar{B}^1 \text{ is multilinear.}$$

For the last equivalence note that if  $\bar{B}^1$  is multilinear, then  $\bar{B}^1(\bar{x}_1, \bar{x}_2, \dots, \bar{x}_N) = 0$  if any  $\bar{x}_i = 0$ .

(v) Assume  $\bar{B}^1(\bar{x}_1, \bar{x}_2, \dots, \bar{x}_N) > \max_{\{0,1\}^N} B$  and w.l.o.g.  $\bar{x}_1 \notin \{0, 1\}$ . Then it follows from (iv) that  $\bar{B}^1(0, \bar{x}_2, \dots, \bar{x}_N) > \max_{\{0,1\}^N} B$  or  $\bar{B}^1(1, \bar{x}_2, \dots, \bar{x}_N) > \max_{\{0,1\}^N} B$ . Inductively, we obtain  $x \in \{0, 1\}^N$  such that  $\bar{B}^1(x) > \max_{\{0,1\}^N} B$ , a contradiction to (i). Analogously, one proves that  $\bar{B}^1(\bar{x}_1, \bar{x}_2, \dots, \bar{x}_N) \geq \min_{\{0,1\}^N} B$ .  $\square$

Note that in general  $\deg(\bar{B}^1) = N - s$  does not hold, as can be seen from the Boolean function in three variables  $x_1, x_2, x_3$

$$\begin{matrix} B(0, 0, 0) = 0 & B(0, 0, 1) = 1 & B(0, 1, 0) = 1 & B(0, 1, 1) = 1 \\ B(1, 0, 0) = 1 & B(1, 0, 1) = 1 & B(1, 1, 0) = 1 & B(1, 1, 1) = 0 \end{matrix}$$

This function does not have any symmetry hyperplanes, but its interpolation  $\bar{B}^1 = \bar{x}_1 + \bar{x}_2 + \bar{x}_3 - \bar{x}_1 \bar{x}_2 - \bar{x}_1 \bar{x}_3 - \bar{x}_2 \bar{x}_3$  has degree 2.

### Theoretical results about steady-states

The following theorem investigates the steady-state behavior of discrete and continuous models.

**Theorem.** Assume we are given a Boolean model and perfect continuous homologues  $\bar{B}_i$  of the Boolean update functions. Then for any state vector  $\bar{x} = (\bar{x}_1, \bar{x}_2, \dots, \bar{x}_N) \in \{0, 1\}^N$  the following are equivalent:

- (i)  $\bar{x}$  is a steady-state of the Boolean model.
- (ii)  $\bar{x}$  is a steady-state of the model (1).
- (iii)  $\bar{x}$  is a steady-state of the ODE model (2).

Proof. The steady-state conditions are

$$\bar{x}_i(t) = \bar{x}_i(t + 1) = B_i(\bar{x}_{i1}(t), \bar{x}_{i2}(t), \dots, \bar{x}_{iN_i}(t)), \quad i = 1, 2, \dots, N$$

for the Boolean model,

$$\bar{x}_i(t) = \bar{x}_i(t + 1) = \bar{B}_i(\bar{x}_{i1}(t), \bar{x}_{i2}(t), \dots, \bar{x}_{iN_i}(t)), \quad i = 1, 2, \dots, N$$

for model (1) and

$$0 = \dot{\bar{x}}_i(t) = \frac{1}{\tau_i} (\bar{B}_i(\bar{x}_{i1}(t), \bar{x}_{i2}(t), \dots, \bar{x}_{iN_i}(t)) - \bar{x}_i(t)), \quad i = 1, 2, \dots, N$$

for model (2). Considering that the  $\bar{B}_i$  are perfect homologues of the  $B_i$ , these conditions are clearly equivalent.  $\square$

*Remark.* Note that the above theorem generally applies to transformations using perfect continuous homologues of Boolean functions. In the special case of piecewise linear ODEs, i.e.  $\bar{B}_i = \bar{B}_i^{PW}$  from equation (6), Glass et al. [11] could show that the above theorem also holds when (iii) is replaced by ' $\bar{x}$  is a stable steady-state of the ODE model (2)'.  
 We now extend the above theorem to HillCube models. The problem is that we can no longer assume a perfect agreement between  $\bar{B}_i^H$  and  $B_i$ . The main idea is to use the implicit function theorem to prove the existence of a steady-state of the continuous model in a neighborhood of a Boolean steady-state.

**Theorem.** Assume we are given a Boolean model and the corresponding HillCube model. Let  $x^\infty \in \{0, 1\}^N$  be a steady-state of the Boolean model. We fix the thresholds  $k_{ij}$  at values in  $(0, 1)$  and the life-times  $\tau_i$  at values in  $(0, \infty)$ . Then there exists a neighborhood  $U$  of  $x^\infty$  such that for all sufficiently large Hill exponents  $n = (n_{ij})$  the HillCube model has a stable steady-state  $\bar{x}^n$  in  $U$ . It holds

$$\lim_{\text{all } n_{ij} \rightarrow \infty} \bar{x}^{(n)} = x^\infty.$$

*Proof.* For a single Hill exponent  $n$  we write  $n = m^{-2}$ ,  $m \neq 0$ , fix some  $0 < k < 1$  and consider the Hill function

$$f(\bar{x}, m) = \frac{\bar{x}^{m^{-2}}}{k^{m^{-2}} + \bar{x}^{m^{-2}}}.$$

We extend this function at  $m = 0$  by setting

$$f(\bar{x}, 0) = \begin{cases} 0, & \bar{x} \leq k \\ 1, & \bar{x} > k \end{cases}.$$

Then there are open neighborhoods  $U_1, U_2$  of 0 and 1, respectively, such that  $f(\bar{x}, m)$  is continuous on  $U_{1/2} \times \mathbb{R}$  and it can be easily shown that  $f$  is even continuously differentiable on  $U_{1/2} \times \mathbb{R}$ . Now consider the HillCube model. It depends only on the Hill exponents  $n$  and we define  $m = (m_{ij})$  by  $n_{ij} = m_{ij}^{-2}$ . For concentrations  $\bar{x} = (\bar{x}_1, \bar{x}_2, \dots, \bar{x}_N)$  let  $\Phi(m, \bar{x})$  denote the right hand side of the HillCube ODE system. As explained above, we continuously extend the Hill functions and hence also  $\Phi$  at  $m = 0$ . Then there exists an open neighborhood  $U \subset \mathbb{R}^N$  of  $x^\infty$  such that  $\Phi$  is continuously differentiable on  $\mathbb{R}^{|m|} \times U$  (as the composition of continuously differentiable functions).

For  $m = 0$  the Hill functions become Boolean step functions and hence the HillCubes perfectly agree with the Boolean update rules. Therefore, we have  $\Phi(0, x^\infty) = 0$ . Now, let us compute the Jacobian  $D\Phi$  of  $\Phi$  in  $(0, x^\infty)$ .

$$\begin{aligned} \frac{\partial \Phi_i}{\partial \bar{x}_j}(0, x^\infty) &= \frac{\partial}{\partial \bar{x}_j} \bigg|_{(0, x^\infty)} \left[ \frac{1}{\tau_i} (\bar{B}_i^H(\bar{x}_{i1}, \bar{x}_{i2}, \dots, \bar{x}_{iN_i}) - \bar{x}_i) \right] \\ &= \frac{1}{\tau_i} \left[ \frac{\partial}{\partial \bar{x}_j} \bigg|_{(0, x^\infty)} \bar{B}_i^H(\bar{x}_{i1}, \bar{x}_{i2}, \dots, \bar{x}_{iN_i}) - \frac{\partial}{\partial \bar{x}_j} \bigg|_{(0, x^\infty)} \bar{x}_i \right] = -\frac{1}{\tau_i} \delta_{ij}, \end{aligned}$$

where  $\delta_{ij}$  is the Kronecker delta. Note that for  $m = 0$  the Hill functions in the HillCubes are step functions, i.e. constant in a neighborhood of  $x^\infty$ . Hence the derivative of the HillCubes vanishes. This shows that  $D\Phi$  is diagonal negative definite and hence, in particular, invertible. Therefore, the implicit function theorem guarantees that there are open neighborhoods  $U$  of  $x^\infty$  and  $V \subset \mathbb{R}^{|m|}$  of 0 such that for each  $m \in V$ , i.e. for large exponents  $n$ , there is a  $\bar{x}^n \in U$  such that  $\Phi(m, \bar{x}^n) = 0$ , i.e.  $\bar{x}^n$  is a steady-state of the model. Since the mapping  $m \mapsto \bar{x}^n$  is continuous we have  $\bar{x}^n \rightarrow x^\infty$  as  $m \rightarrow 0$  with respect to the euclidean norm, i.e. all  $m_{ij} \rightarrow 0$  or equivalently  $n_{ij} \rightarrow \infty$ . Moreover, it follows that on subsets  $V \subseteq V'$  and  $U \subseteq U'$  the Jacobian of  $\Phi$  is still negative definite and, consequently, the  $\bar{x}^n$  are stable steady-states.  $\square$

**Manual parameter determination for the continuous T-cell model**

The question is how to encode the information about slow and fast interactions in the numeric values of the parameters. We illustrate our approach using the subnetwork shown in Figure 2C, where we focus on ZAP-70, LAT phosp and cCbl. The activation of LAT phosp stands for the interactions of scenario 2, whereas the activation of cCbl represents scenario 3. The ODE system for this network is given by

$$\begin{aligned} \frac{d[\text{ZAP-70}]}{dt} &= \frac{1}{\tau_{\text{ZAP-70}}} \left( \frac{[\text{Input}]^n}{[\text{Input}]^n + k^n} - [\text{ZAP-70}] \right) \\ \frac{d[\text{LAT phosp}]}{dt} &= \frac{1}{\tau} \left( \frac{[\text{ZAP-70}]^n}{[\text{ZAP-70}]^n + k_{\text{fast}}^n} - [\text{LAT phosp}] \right) \\ \frac{d[\text{cCbl}]}{dt} &= \frac{1}{\tau} \left( \frac{[\text{ZAP-70}]^n}{[\text{ZAP-70}]^n + k_{\text{slow}}^n} - [\text{cCbl}] \right). \end{aligned}$$

For the activation of ZAP-70  $k = 0.3$  and  $\tau_{\text{ZAP-70}} = 10$  are used. LAT phosp and cCbl are activated at thresholds  $k_{\text{fast}} = 0.1$  and  $k_{\text{slow}} = 0.8$ , respectively, and both their life-times are set to  $\tau = 1$ . Initial condition for all species is 0 and the inputs for ZAP-70 are fixed at 1. The result of

the numeric simulation is shown in Figure 2D. From the beginning on ZAP-70 is activated but rather slowly due to the increased life-time  $\tau_{ZAP-70}$ . The activations of LAT-phosph and cCbl occur at around the time points when the concentration of ZAP-70 crosses the thresholds  $k_{fast}$  and  $k_{slow}$ , respectively. The high threshold  $k_{slow}$  also leads to a lower total activation level of cCbl. One can clearly see the time lag between the activations of LAT-phosph and cCbl, i.e. between the interactions in scenario 2 and the interactions not occurring until scenario 3.

**Model transformation and simulation**

For the transformation of the Boolean T-cell model we choose HillCube ODEs. HillCubes are better suited to describe signaling cascades than BooleCubes. Normalization is not necessary as we will choose thresholds  $k \ll 1$  and consequently the Hill functions will already satisfy  $f(1) \approx 1$ . For the transformation and simulation we developed a MATLAB toolbox called *Odefy* [Krumisiek et al.: *Odefy — From discrete to continuous models. In preparation (2009)*], which is publicly available at <http://cmb.helmholtz-muenchen.de/odefy> and allows the experimentalist to easily transform Boolean models into ODE models. Since *Odefy* can be integrated into *CellNetAnalyzer* [24], we were able to export the Boolean T-cell model from there. Numeric integration of the ODE system was carried out using MATLAB *ode15s*, a variable-order multistep solver based on the numerical differentiation formulas.

**Experimental data**

Kemp et al. [12] created a data set describing the dynamics of the activation of the key signaling elements ERK, JNK, IKK and NFAT upon activation of the TCR. The data was generated by stimulation of a T-cell line (1B6 T cell hybridoma) with three peptides with different affinities for the T-cell receptor, Q144, Y144 and L144. In the case of L144 experiments were conducted for three different peptide concentrations 0.04  $\mu\text{g/ml}$  (low), 0.4  $\mu\text{g/ml}$  (medium) and 4  $\mu\text{g/ml}$  (high). In the case of Q144 and Y144 only the high concentration of 4  $\mu\text{g/ml}$  was used. Concentrations of ERK, JNK, IKK and NFAT were measured at 0, 10, 30, 60, 120, 240 and 2400 minutes. Here we neglected the last time point, as on this slow time scale many interactions play a role, that are not included in the model, such as gene expression. For the parameter fitting the data were linearly rescaled to the unit interval.

**Parameter fitting**

The T-cell model consists of 40 species, 55 pairwise interactions and three external inputs. Hence, we have 40 life-time parameters and 58 pairs of Hill parameters, amounting to a total of 156 parameters. Due to this large

number of parameters compared to the number of experimental data points, the fitting problem is obviously ill-posed as for many different parameter sets the model reproduces the data equally well. For this reason, we performed a two-step fitting process. First, we determined a parameter set for which the model fits the experimental data reasonably well. Second, we added a regularization to account for the indeterminacies. Both steps are optimization problems. The two cost functions are given below. They take a parameter set consisting of all Hill parameters ( $n, k$ ) and all life-times  $\tau$  as input to yield a scalar loss value, that needs to be minimized. In both steps we used a *simulated annealing* algorithm [25] for minimization. As the threshold parameters  $k$  have to be precisely adjusted at small values, these were fitted on a log-scale, as is also done in [26]. Parameters downstream of the measured species were, of course, not changed but fixed at their manually determined value. We used the SBPD package of the *Systems Biology Toolbox* [27] to create a compiled MATLAB simulation function of our ODE model for faster performance.

**Least squares fitting**

In the first step we determined a parameter set for which the model reproduces the data reasonably well. To this end, we employed a least squares fitting, i.e. we minimized the sum of the squared offsets of the data points from the model prediction. In a first attempt, we used only the offsets at the six time points  $t = 0, 10, 30, 60, 120, 240$ . This led to the model showing fast oscillations, which almost perfectly fitted the data, and, clearly, were an unrealistic overfitting. To avoid this, we linearly interpolated the experimental data and minimized the cost function

$$\mathcal{L}_1 = \sum_{i \in \{\text{low, medium, high}\}} \sum_{s \in \{\text{NFAT, IKK, JNK, ERK}\}} \sum_{t=0}^{240} (\bar{s}_i(t) - \tilde{s}_i(t))^2,$$

where  $\bar{s}_i$  denotes the time courses predicted by the model and  $\tilde{s}_i$  the interpolated data points. The simulated annealing algorithm was started from the manually determined parameter set and finally converged at a cost function value of 18.98.

**Regularization**

In the regularization step we minimized the sum of the coefficients of variation  $\sigma_n/\mu_n$ ,  $\sigma_k/\mu_k$  and  $\sigma_\tau/\mu_\tau$  within the three parameter groups of  $n$ 's,  $k$ 's and  $\tau$ 's, respectively, under the constraint that the model's fit to the data did not deteriorate. The corresponding cost function is

$$\mathcal{L}_2 = \frac{\sigma_n}{\mu_n} + \frac{\sigma_k}{\mu_k} + \frac{\sigma_\tau}{\mu_\tau} + \begin{cases} \infty & \text{if } \mathcal{L}_1 \geq 19 \\ 0 & \text{if } \mathcal{L}_1 < 19 \end{cases}$$

where the last term is a penalty term ensuring a constant quality of the model's fit. The simulated annealing was started from the result of the first step. Idea of this regularization is to account for parameter indeterminacies by reducing their variation and to enhance the significance of 'outliers' like the affinity  $k_{L144}$  of the TCR for the ligand L144, cf. Figure 4E.

Additional data file 6 shows model simulations for 5 different results of step 1. While not perfectly agreeing, their overall dynamic behavior is the same. We can reasonably assume that neither the parameter indeterminacies nor the regularization significantly influence the dynamics of the experimentally observed species.

### Authors' contributions

DMW conceived and designed the methodology, performed the computational experiments and drafted the manuscript. JK helped to perform the computational experiments and developed the computational platform. JSR, DAL and SK interpreted the numerical and experimental data and contributed to the manuscript. FJT participated in the design of the methodology and of the computational platform and helped to draft the manuscript. All authors read and approved the final manuscript.

### Additional material

#### Additional file 1

*Example for the hypergraph representation of a Boolean model. Supplementary text (.pdf) giving an example for the hypergraph representation of Boolean models.*

Click here for file

[<http://www.biomedcentral.com/content/supplementary/1752-0509-3-98-S1.PDF>]

#### Additional file 2

*Hill functions. Figure (.pdf) showing Hill functions  $f(x) = x^n/(x^n + k^n)$  with Hill coefficients  $n = 2, 4, 8, 16, 32$  and threshold  $k = 0.5$ .*

Click here for file

[<http://www.biomedcentral.com/content/supplementary/1752-0509-3-98-S2.PDF>]

#### Additional file 3

*HillCubes. Figure (.png) showing HillCubes of all 16 two-variable Boolean gates. Hill parameters are  $n = 3$  and  $k = 0.5$  for both inputs.*

Click here for file

[<http://www.biomedcentral.com/content/supplementary/1752-0509-3-98-S3.PNG>]

#### Additional file 4

*Regulatory domains. Figure (.pdf) showing the regulatory domains in a two-variable example.*

Click here for file

[<http://www.biomedcentral.com/content/supplementary/1752-0509-3-98-S4.PDF>]

#### Additional file 5

*Steady-states in discrete and continuous models. Supplementary text (.pdf) discussing a toy example where the steady-states of a discrete model are not preserved in a continuous version of the model.*

Click here for file

[<http://www.biomedcentral.com/content/supplementary/1752-0509-3-98-S5.PDF>]

#### Additional file 6

*Comparison of different best fit parameter sets with respect to model dynamics. Figure (.pdf) showing simulations of the continuous T-cell model for 5 different best fit parameter sets (without regularization, cf. section on parameter fitting). While not perfectly agreeing, the overall dynamic behavior is the same in all simulations.*

Click here for file

[<http://www.biomedcentral.com/content/supplementary/1752-0509-3-98-S6.PDF>]

### Acknowledgements

The authors would like to thank Melissa Kemp for providing the experimental data, Melody Morris for critical reading of the manuscript and useful discussions as well as the anonymous reviewers for their helpful comments during the preparation of this manuscript.

This research was partially supported by the Initiative and Networking Fund of the Helmholtz Association within the Helmholtz Alliance on Systems Biology (project CoReNe), by the German Federal Ministry of Education and Research (MaCS, Magdeburg Centre for Systems Biology) and by the Ministry of Education of Saxony-Anhalt (Research Center "Dynamic Systems). JSR and DAL thank funding of Pfizer Inc., NIH grant GM68762, and DOD Institute for Collaborative Biotechnologies.

### References

1. Kauffman SA: **Metabolic stability and epigenesis in randomly constructed genetic nets.** *Journal of Theoretical Biology* 1969, **22(3)**:437–467.
2. Fauré A, Naldi A, Chaouiya C and Thieffry D: **Dynamical analysis of a generic Boolean model for the control of the mammalian cell cycle.** *Bioinformatics* 2006, **22(14)**:124–131.
3. Davidich MI and Bornholdt S: **Boolean network model predicts cell cycle sequence of fission yeast.** *PLoS ONE* 2008, **3(2)**.
4. Mendoza L, Thieffry D and Alvarez-Buylla ER: **Genetic control of flower morphogenesis in Arabidopsis thaliana: a logical analysis.** *Bioinformatics* 1999, **15(7/8)**:593–606.
5. Albert R and Othmer HG: **The topology of the regulatory interactions predicts the expression pattern of the segment polarity genes in Drosophila melanogaster.** *Journal of Theoretical Biology* 2003, **223**:1–18.
6. Klamt S, Saez-Rodriguez J, Lindquist JA, Simeoni L and Gilles ED: **A methodology for the structural and functional analysis of signalling and regulatory networks.** *BMC Bioinformatics* 2006, **7**:56.
7. Chavez M, Albert R and Sontag E: **Robustness and fragility of Boolean models for genetic regulatory networks.** *Journal of Theoretical Biology* 2005, **235**:431–449.
8. 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 Comput Biol* 2007, **3(8)**:e163.
9. Altan-Bonnet G and Germain R: **Modeling T cell antigen discrimination based on feedback control of digital ERK responses.** *PLoS Biol* 2005, **3(11)**:e356.
10. Mendoza L and Xenarios I: **A method for the generation of standardized qualitative dynamical systems of regulatory networks.** *Theoretical Biology and Medical Modelling* 2006, **3(13)**.

11. Glass L and Kauffman SA: **The logical analysis of continuous, non-linear biochemical control networks.** *Journal of Theoretical Biology* 1973, **39**:103–129.
12. Kemp ML, Wille L, Lewis CL, Nicholson LB and Lauffenburger DA: **Quantitative network signal combinations downstream of TCR activation can predict IL-2 production response.** *J Immunol* 2007, **178(8)**:4984–4992.
13. Gasca M and Sauer T: **On the history of multivariate polynomial interpolation.** *Journal of Computational and Applied Mathematics* 2000, **122(1-2)**:23–35.
14. Plahte E, Mestl T and Omholt S: **A methodological basis for description and analysis of systems with complex switch-like interactions.** *Journal of Mathematical Biology* 1998, **36(4)**:321–348.
15. Hill A: **The possible effects of the aggregation of the molecules of haemoglobin on its dissociation curves.** *J Physiol* 1910, **40**:4–7.
16. El Snoussi H and Thomas R: **Logical identification of all steady states: The concept of feedback loop characteristic states.** *Bulletin of Mathematical Biology* 1993, **55(5)**:973–991.
17. Thomas R, Thieffry D and Kaufman M: **Dynamical behaviour of biological regulatory networks-I. Biological role of feedback loops and practical use of the concept of the loop-characteristic state.** *Bulletin of Mathematical Biology* 1995, **57(2)**:247–276.
18. Werlen G, Hausmann B and Palmer E: **A motif in the alphabeta T-cell receptor controls positive selection by modulating ERK activity.** *Nature* 2000, **406(6794)**:422–426.
19. Werlen G: **Signaling Life and Death in the Thymus: Timing Is Everything.** *Science* 2003, **299(5614)**:1859–1863.
20. Daniels MA, Teixeira E, Gill J, Hausmann B, Roubaty D, Holmberg K, Werlen G, Holländer GA, Gascoigne NRJ and Palmer E: **Thymic selection threshold defined by compartmentalization of Ras/ MAPK signalling.** *Nature* 2006, **444(7120)**:724–729.
21. Munder M, Bettelli E, Monney L, Slavik J, Nicholson L and Kuchroo V: **Reduced Self-Reactivity of an Autoreactive T Cell After Activation with Cross-reactive Non-Self-Ligand.** *The Journal of Experimental Medicine* 2002, **196(9)**:1151–1162.
22. de Jong H, Gouzé J, Hernandez C, Page M, Sari T and Geiselman J: **Qualitative simulation of genetic regulatory networks using piecewise-linear models.** *Bulletin of Mathematical Biology* 2004, **66(2)**:301–340.
23. Zadeh LA: **Fuzzy sets.** *Information and Control* 1965, **8(3)**:338–353.
24. Klamt S, Saez-Rodriguez J and Gilles E: **Structural and functional analysis of cellular networks with CellNetAnalyzer.** *BMC Systems Biology* 2007, **1(2)**.
25. Kirkpatrick S, Gelatt C and Vecchi M: **Optimization by Simulated Annealing.** *Science* 1983, **220(4598)**:671–680.
26. Ma W, Lai L, Ouyang Q and Tang C: **Robustness and modular design of the Drosophila segment polarity network.** *Molecular Systems Biology* 2006, **2(70)**.
27. Schmidt H and Jirstrand M: **Systems Biology Toolbox for MATLAB: a computational platform for research in systems biology.** *Bioinformatics* 2006, **22(4)**:514–515.
28. Huang C and Ferrell J: **Ultrasensitivity in the mitogen-activated protein kinase cascade.** *Proceedings of the National Academy of Sciences* 1996, **93(19)**:10078–10083.

Publish with **BioMed Central** and every scientist can read your work free of charge

"BioMed Central will be the most significant development for disseminating the results of biomedical research in our lifetime."

Sir Paul Nurse, Cancer Research UK

Your research papers will be:

- available free of charge to the entire biomedical community
- peer reviewed and published immediately upon acceptance
- cited in PubMed and archived on PubMed Central
- yours — you keep the copyright

Submit your manuscript here:  
[http://www.biomedcentral.com/info/publishing\\_adv.asp](http://www.biomedcentral.com/info/publishing_adv.asp)

