

C. STÖTZEL¹, S. RÖBLITZ^{1,3,2}, H. SIEBERT³

Complementing ODE-based System Analysis Using Boolean Networks Derived from an Euler-like Transformation

¹Mathematics for Life and Materials Sciences, Zuse Institute Berlin, Berlin, Germany

²Corresponding author E-mail: susanna.roebnitz@zib.de

³Dep. of Mathematics and Computer Science, Freie Universität Berlin, Berlin, Germany

Herausgegeben vom
Konrad-Zuse-Zentrum für Informationstechnik Berlin
Takustraße 7
14195 Berlin
Germany

Telefon: 030-84185-0
Telefax: 030-84185-125

e-mail: bibliothek@zib.de
URL: <http://www.zib.de>

ZIB-Report (Print) ISSN 1438-0064
ZIB-Report (Internet) ISSN 2192-7782

Complementing ODE-based System Analysis Using Boolean Networks Derived from an Euler-like Transformation

C. Stötzel, S. Röblitz, H. Siebert

September 2015

Abstract

In this paper, we present a systematic transition scheme for a large class of ordinary differential equations (ODEs) into Boolean networks. Our transition scheme can be applied to any system of ODEs whose right hand sides can be written as sums and products of monotone functions. It performs an Euler-like step which uses the signs of the right hand sides to obtain the Boolean update functions for every variable of the corresponding discrete model. The discrete model can, on one hand, be considered as another representation of the biological system or, alternatively, it can be used to further the analysis of the original ODE model. Since the generic transformation method does not guarantee any property conservation, a subsequent validation step is required. Depending on the purpose of the model this step can be based on experimental data or ODE simulations and characteristics. Analysis of the resulting Boolean model, both on its own and in comparison with the ODE model, then allows to investigate system properties not accessible in a purely continuous setting. The method is exemplarily applied to a previously published model of the bovine estrous cycle, which leads to new insights regarding the regulation among the components, and also indicates strongly that the system is tailored to generate stable oscillations.

Introduction

When modeling biological phenomena, different levels of abstraction can be used to capture the mechanisms of the underlying system [13]. Ordinary differential equations (ODEs) are well suited to describe interactions based on concentration changes and allow for detailed predictions. However, due to the complexity of the models, it is often not possible to perform a global analysis of dynamical properties. Additionally, the model behavior strongly depends on the current parameter values such as rate constants, and the quantitative data necessary to identify them is often lacking [25]. In this case, qualitative modeling, e.g., using Boolean formalisms, constitutes a fruitful alternative. Herein, the system is described on a higher level of abstraction strongly based on the network structure and a logical description of interaction mechanisms. While the coarseness of such models does not allow for high resolution of model output, it does permit a much more global analysis of a system due to a finite state space and has shown to be capable of uncovering fundamental characteristics and functionalities [27].

Common to all modeling formalisms is the fact that a model always constitutes a simplification of reality, focusing on particular aspects of interest and neglecting details either by design or simply due to lack of knowledge. Utilizing different modeling approaches in parallel may therefore help to produce a more rounded picture of a system's capabilities and to highlight modeling artifacts (see e.g. [1, 18, 20]).

We aim at exploiting the strengths and weaknesses of ODE and logical modeling by using them in concert. Taking the opposite direction as [3], we start from an ODE system and propose a method to derive a Boolean model offering a new point of view on the system without having to repeat an explicit modeling process to obtain the more abstract model. The simplicity of the Boolean representation allows for a much more comprehensive analysis of its dynamics to

complement the more local results obtained from the ODE system. A global attractor analysis as often infeasible for ODE models, e.g., allows to understand capabilities of the system in terms of stable behavior and to uncover decision processes determining long-term dynamics. In application, such insights are of particular interest when analyzing differentiation or multi-stable response systems [11, 7]. The easy adaption of such models to knock-out/in scenarios facilitates extensive perturbation studies [19]. In addition, the coarser representation allows for easier identification of structural properties that are fundamental for specific dynamical effects, e.g., based on a feedback circuit analysis [17].

The main result of our paper is a new, systematic and simple method to obtain a Boolean from an ODE model for the purpose of enhancing system analysis. Existing approaches of relating continuous and logical models can roughly be divided according to two different goals. In one research line, the interest lies in capturing the way system properties are conserved when discretizing continuous models in a rigorous mathematical way. Here, good results can be achieved when focusing on small classes of ODE models where the translation into a discrete setting is rather straightforward [26]. In [10], a method is proposed to map continuous models of biochemical control networks, in which the control functions are continuous homologues of logical control functions, to discrete models. For a restricted subclass of dynamical systems, a correspondence regarding certain qualitative dynamical features has been proven. Later, the authors in [9] analyzed transitions of dynamics in randomly constructed ODE analogues of random Boolean networks.

In other works, the emphasis is more on the application side, aiming at deriving discrete models useful for system analysis. As in [5], such methods are often only described closely intertwined with a case study. This leads to an individual solution that depends on many regulation-based distinctions and steady state assumptions. In [5], the transformation from ODEs to Boolean networks is done in a controlled way, keeping the dynamical properties in mind. They find that the transition is possible for differential equations which have monotonic sigmoidal functions with distinct upper and lower asymptotes. Finally, the author show that the Boolean model reproduces the results of the initial ODE model.

In this paper, we do not focus on conservation of desired dynamical properties during the translation process. Instead, we propose a systematic and thus easily accessible, step-wise procedure of translation with subsequent validation and analysis suitable for a large class of ODE models interesting for application. Discretization of the continuous system is based on the Euler method, a well-known tool for the numerical approximation of continuous dynamical systems (see e.g [12] for a general reference and [23], Chapter 8, for a related application). We only use the structure and monotonicity behavior of the right hand side functions of the ODEs for directly deriving the discrete model. Since in general conservation of specific dynamical properties cannot be predicted a priori, we proceed with a validation step to check if at least the properties of interest are preserved. This allows for the construction of feasible models without the prerequisite of fulfilling specific mathematical assumptions in advance. Nevertheless, the corresponding Boolean model preserves the qualitative biological information comprised in the ODE model, namely the underlying network structure and the processing logic of regulatory effects, and thus includes the information that would be incorporated in a traditional bottom-up modeling procedure for Boolean networks.

The validation step ensures that the derived Boolean model satisfies the given requirements in terms of property conservation. It can be conducted utilizing Boolean-like biological data, aiming at a model capturing all important biological observations. A different possibility is to validate the Boolean with respect to the ODE model by generating data from the simulation of the latter for the purpose of validation. In particular, this might be of interest if data on the real system is lacking or if analysis of the Boolean model aims at providing pointers for a more targeted analysis of the ODE model. When the system functionality of interest is governed by qualitative regulation processes, as often the case, e.g., in differentiation, bistable switching or signal processing, validation is very likely to succeed. Note that, even if validation would fail, we always gain information on the system since the discretization procedure pinpoints processes whose functionality crucially depends on certain quantitative processes. Analysis of the Boolean model then concludes the procedure. As mentioned, the focus can either lie on gathering new

insights into the modeled system or aiming at a more comprehensive analysis of the ODE model.

In the first part of this paper, we present the general approach to deriving a Boolean from a given ODE model. Then, since the validation step depends on the availability of data as well as on the intended purpose of the model, we describe some validation possibilities often of use in application. Similarly, we shortly discuss possible analysis approaches for the discrete model to round off the description of the method. To illustrate the procedure, we apply the approach to a medium-sized model of the bovine estrous cycle, named *BovCycle*, that has been presented in [2, 21]. Utilizing the generated Boolean model, we investigate stability, prerequisites for certain observed component activities, and the impact of self-regulatory effects. In contrast to steady states, proving conservation of cyclic attractors in a purely mathematical setting is notoriously difficult. The proposed validation method demonstrates how to check the conservation of oscillatory behavior and is suitable for both transient trajectories and attractors. It nicely illustrates the potential of our more pragmatic approach in application.

Results

In this section, we present our procedure for deriving a discrete model from an ODE model. We formulate assumptions for the ODE model (Step 0), and perform an Euler-like discretization (Step 1) which leads to a Boolean model. We then propose strategies for validation (Step 2) and analysis (Step 3) of the discrete model according to the given specifics.

Proposed Method for the Discretization of an ODE Model

In the following, we describe the four steps of our discretization scheme.

Step 0: Preliminaries and Assumptions. As starting point, we consider an autonomous system of ODEs in the form

$$y'(t) = f(y, p), \quad y(0) = y_0$$

with time variable $t \in \mathbb{R}_{\geq 0}$, state vector $y(t) = (y_1(t), \dots, y_n(t))^T \in \mathbb{R}^n$, model parameters $p \in \mathbb{R}^q$, and right hand side $f : \mathbb{R}^{n \times q} \mapsto \mathbb{R}^n$. For simplicity of notation, the parameter vector p will be omitted in the following.

Throughout the paper, we assume that all continuous variables $y_i(t)$ are normalized to the interval $[0, 1]$. This can easily be achieved for every given ODE system by rescaling of model variables and parameters. Accordingly, initial values $y(0)$ are also non-negative and bounded by one.

In addition, we assume that the right hand side functions $f(y) = (f_1(y), \dots, f_n(y))^T$ consist only of sums and products of *monotone* functions $F_{i,j}(y_j)$,

$$y'_i(t) = f_i(F_{i,1}(y_1), \dots, F_{i,n}(y_n)), \quad i = 1, \dots, n, \quad (1)$$

whereby each monotone function $F_{i,j}(y_j)$ only takes values in $[0, 1]$, which can be achieved by appropriate shifting and scaling. An example for such a function $F_{i,j}(y_j)$ is the increasing Hill function

$$H^+(y_j, T, r) = \frac{y_j^r}{y_j^r + T^r}$$

with model parameters T and r . An example for the combination of two monotone functions is the product of two species

$$F_{i,j}(y_j) \cdot F_{i,k}(y_k) = y_j \cdot y_k$$

as it typically occurs in the modeling of a bimolecular reaction. Monotonicity is not a strong requirement in the context of modeling biological systems because classical mass action kinetics as well as Michaelis-Menten kinetics, Goldbeter-Koshland kinetics or Hill kinetics all fall into the class of monotone functions.

Step 1: Translating the ODE System into a Logical Model by an Euler-like Discretization. In the discrete counterpart of the ODE model, each real-valued variable $y_i(t)$, $i = 1, \dots, n$, is represented by a corresponding discrete variable x_i . In the following, we will focus on deriving a Boolean network from the ODE model, i.e., each variable x_i takes values in the set $\{0, 1\}$, yielding the state space $\{0, 1\}^n$.

For this purpose, the update function $g : \{0, 1\}^n \rightarrow \{0, 1\}^n$ of the Boolean model needs to be constructed. The goal is to capture the regulatory effects given by the ODE model, expressing them in an explicit form describing the state transitions directly rather than the implicit form of the ODE system. Clearly, the coarser nature of the logical model will emphasize certain effects while smoothing out others. However, the qualitative behavior of the ODE model should be preserved.

In our method, the expression ‘‘Euler-like step’’ refers to its similarity with the explicit Euler method for the numerical integration of ODEs. The idea is to evaluate the difference between a current state s of the ODE system and the state obtained by one Euler step, i.e., the state obtained by approximating the trajectory starting in s by the tangent in s and following it for a step of given length. In the discrete setting, we will discretize the impact of the regulators and only evaluate whether the tangent in s indicates a value increase, decrease or no value change at all.

To this end, we replace all continuous variables y_1, \dots, y_n in the right hand side functions $f_i(y)$ by their discrete counterparts x_1, \dots, x_n . Thereby, every monotonically increasing function $F_{i,j}(y_j)$ is mapped to x_j , and every monotonically decreasing function $F_{i,j}(y_j)$ is mapped to $(1 - x_j)$. We thus define an operator

$$T(F) := \begin{cases} Id & , \text{ if } F \text{ is monotonically increasing,} \\ 1 - Id & , \text{ if } F \text{ is monotonically decreasing,} \end{cases}$$

and map the right hand sides $f_i(y_1, \dots, y_n)$ in (1) to discrete counterparts $h_i(x_1, \dots, x_n)$,

$$f_i(F_{i,1}(y_1), \dots, F_{i,n}(y_n)) \mapsto f_i(\underbrace{TF_{i,1}(x_1)}_{x_1 \text{ or } 1-x_1}, \dots, \underbrace{TF_{i,n}(x_n)}_{x_n \text{ or } 1-x_n}) =: h_i(x_1, \dots, x_n),$$

for all $i = 1, \dots, n$.

Note that the values of the thus defined functions $h_i : \{0, 1\}^n \rightarrow \mathbb{R}$ depend not only on the values of the Boolean variables x_1, \dots, x_n , but also on the model parameters p . Depending on these values, $h_i(x) = h_i(x_1, \dots, x_n)$ can either be negative, positive, or equal to zero. The sign of $h_i(x)$ then determines the update rule for the Boolean variables. The update function g is defined as $g : \{0, 1\}^n \rightarrow \{0, 1\}^n$ with

$$g_i : \{0, 1\}^n \rightarrow \{0, 1\} : x \mapsto g_i(x) := \begin{cases} 1 & , \text{ if } \text{sgn}(h_i(x)) > 0 \\ x_i & , \text{ if } \text{sgn}(h_i(x)) = 0, \quad i = 1, \dots, n \\ 0 & , \text{ if } \text{sgn}(h_i(x)) < 0 \end{cases} \quad (2)$$

This can be interpreted as an Euler discretization step yielding the successor state x^{k+1} for a current state x^k in the form

$$x_i^{k+1} = g_i(x^k) = x_i^k \dot{+} \text{sgn}(h_i(x_1^k, \dots, x_n^k)),$$

where $\dot{+}$ denotes the operation $0 \dot{+} (-1) = 0$ and $1 \dot{+} 1 = 1$ and the usual addition in all other cases. Note that the discrete functions inherit the monotonicity from their continuous counterparts. This monotonicity behavior is independent from the length of any time step.

Remark: Step 1 can be applied in a generic manner to all ODE models satisfying the given conditions and results in a fully specified Boolean model. The following two steps, validation and analysis of the Boolean model, are highly dependent on the available data and the intended purpose of the model. Consequently, we aim at illustrating the two steps by discussing possible goals and well-established techniques.

Step 2: Validation of the Boolean Model. As mentioned before, the Boolean model can on one hand be considered as another representation of the biological system or, alternatively, it can be used to complement the analysis of the original ODE model. Depending on the purpose, two different data sources can be used to validate the model. In the first case, one should use available experimental data from the biological system of interest. In this case, the Boolean model, regardless of its construction, is in some sense uncoupled from the ODE model. The validation proceeds just as in the case of a model directly constructed from biological data.

In the second case, one should check whether the Boolean model conserves the behavior of the corresponding ODE model, or at least specific aspects of the behavior deemed to be of central interest. Note that we generally cannot ensure such a conservation based on the formal relation between the two models, as is possible to mathematically prove under more rigorous conditions [10, 9]. In this case simulated data from the ODE can be used for the validation procedures. Note that in general one would use simulations that are a good fit to the experimental observations underlying the ODE model, so indirectly this amounts to a validation according to the biological observations as well.

For the validation we concentrate on the dynamical behavior of the model, i.e., its state transition graph. Here, we consider the asynchronous update strategy to derive the state transitions (see Materials and Methods). This update strategy can be viewed as an over-approximation of the system behavior since it takes all possibilities for time delay orderings of regulatory processes into account. The result is a non-deterministic representation, which has been shown to include highly realistic trajectories [23].

We now shortly describe two validation strategies particularly useful in application. For both strategies, either data originating from lab experiments or from the simulation of the ODE model can be utilized.

One common approach exploits time series data. Within the discrete modeling formalism, we can utilize time series measurements as well as more qualitative observations, e.g., a sequence of activation events. Whenever a time series with quantitative measurements is available, we first need to discretize it. For a discussion of suitable discretization methods see [6, 8] and references therein. A discrete time series can be seen as a sequence of states that the system needs to visit in the given order. Additional constraints can be formulated to pose more strict demands on the trajectory, e.g., that component values are not allowed to oscillate between measurements or that the trajectory does not exceed a certain length. To validate the model we thus check whether the model is capable of generating such a trajectory. This can be done very efficiently using formal verification methods (see e.g. [14]).

Another type of data that is readily exploited for validation of discrete models is data from knockout or knock-in experiments. Here, the Boolean model is easily adapted to the perturbed system by fixing the update function of the over- or under-expressed components to the corresponding value [27]. The altered and the non-perturbed model dynamics can then be compared with the experimental observations, e.g., with respect to steady state behavior. Similarly, differentiation capabilities of a system, e.g., represented by bi-stability, can be verified by checking for corresponding attractors of the model.

If validation fails, the next step is to pinpoint the problematic aspects of the model (see e.g. [27, 14] for possible approaches). Comparison with the underlying ODE model can help to identify mechanisms dependent on gradual responses, e.g., concentration gradient dependent processes, that can not be easily captured in a Boolean setting.

Correcting the model, e.g., by adapting the update function of a specific component or adding components to achieve a finer resolution, can then yield a model capable of generating the observed behavior that can be used for further analysis of the biological system. Naturally, the necessary changes need to be analyzed in light of their biological interpretation. Usually, many different changes are possible, and adapting the model can be seen as hypothesis generation for not fully understood mechanisms that need to be more closely investigated experimentally.

Step 3: Analysis of the Boolean Model. Once the Boolean model is validated, it can be analyzed with respect to the properties of interest. Again, we give some examples of analysis methods that are particularly useful in application.

First, analysis of the network structure can be conducted utilizing the dependency graph of the Boolean model. Edges in this graph represent component interactions with an observable impact in the model dynamics (see Materials and Methods). Comparison with the dependencies as captured in the ODE model highlights essential regulatory effects while uncovering others not of importance for generating the qualitative dynamical characteristics. A higher level structural analysis focuses on the *feedback circuit functionality* which allows to relate subnetworks to important dynamical phenomena such as multistability and oscillations [17].

With respect to the dynamical behavior, analysis as well as validation can be focused on specific trajectories, but also on global properties [27]. Commonly, an attractor analysis is conducted to obtain an overview of the possible asymptotic behavior. Additionally, a perturbation analysis, both concerning temporary as well as maintained perturbations, often uncovers system characteristics related to robustness and mechanisms governing decision processes.

All mentioned methods have been implemented in various software packages, e.g., GINsim [16] and BoolNet [15]. Sophisticated algorithms for finite transition systems utilizing, e.g., symbolic encodings allow for very efficient analysis. As already discussed, this can provide a much more comprehensive view of the system dynamics than can be achieved in the ODE setting alone. On the one hand, this might yield possibilities to refine the ODE model analysis, e.g., by discovering additional attractors. These might then also be found in the ODE model when exploiting the rough information about their position in state space derived from the discrete representation. On the other hand, comparison and integration of the results obtained from both models gives a much more rounded picture of the design and functionality of specific model mechanisms as well as a clearer understanding on the system level.

Application to the Model BovCycle

We now apply the above described scheme to the ODE model BovCycle [21] of the bovine estrous cycle. The model describes the growth and decay of the follicles and the corpus luteum, as well as the change of the key reproductive hormones, enzymes, and processes over time. It generates hormone profiles of successive estrous cycles with several follicular waves per cycle. The model consists of 10 ODEs and 38 parameters and is described in more detail in the Materials and Methods section.

Step 0: Preliminaries and Assumptions. The model is an autonomous system of ODEs normalized in such a way that all variables take values in $[0, 1]$. This has been achieved as in [5] by running a simulation of the unscaled ODE system over a time interval I and then using the maximum values $y_i^{\max} = \max_{t \in I} y_i(t)$ to rescale all variables to y_i/y_i^{\max} .

The right hand sides include only terms representing mass action or Hill kinetics, thus they consist only of sums and products of monotone functions taking values in $[0, 1]$. Hence, we can proceed with Step 1.

Step 1: Translating the ODE System into a Logical Model by an Euler-like Discretization. The discrete functions $h_i(x)$ take the following form:

$$\begin{aligned}
h_{\text{GnRH}}(x_{\text{P4}}, x_{\text{E2}}, x_{\text{GnRH}}) &= m_{\text{E2,P4}}^{\text{GnRH}} \cdot (1 - x_{\text{P4}}) \cdot x_{\text{E2}} - c_{\text{GnRH}} \cdot x_{\text{GnRH}}, \\
h_{\text{FSH}}(x_{\text{Inh}}, x_{\text{FSH}}) &= m_{\text{Inh}}^{\text{FSH}} \cdot (1 - x_{\text{Inh}}) - c_{\text{FSH}} \cdot x_{\text{FSH}}, \\
h_{\text{LH}}(x_{\text{P4}}, x_{\text{GnRH}}, x_{\text{LH}}) &= m_{\text{GnRH,P4}}^{\text{LH}} \cdot (1 - x_{\text{P4}}) \cdot x_{\text{GnRH}} - c_{\text{LH}} \cdot x_{\text{LH}}, \\
h_{\text{Foll}}(x_{\text{FSH}}, x_{\text{P4}}, x_{\text{LH}}, x_{\text{Foll}}) &= m_{\text{FSH}}^{\text{Foll}} \cdot x_{\text{FSH}} (1 + m_{\text{Foll}}^{\text{Foll}} \cdot x_{\text{Foll}}) - (m_{\text{P4}}^{\text{Foll}} \cdot x_{\text{P4}} + m_{\text{LH}}^{\text{Foll}} \cdot x_{\text{LH}}) \cdot x_{\text{Foll}}, \\
h_{\text{CL}}(x_{\text{LH}}, x_{\text{Foll}}, x_{\text{IOF}}, x_{\text{CL}}) &= \text{SF}_{\text{CL}} \cdot m_{\text{LH}}^{\text{Foll}} \cdot x_{\text{LH}} \cdot x_{\text{Foll}} + m_{\text{CL}}^{\text{CL}} \cdot x_{\text{CL}} - m_{\text{IOF}}^{\text{CL}} \cdot x_{\text{IOF}} \cdot x_{\text{CL}}, \\
h_{\text{P4}}(x_{\text{CL}}, x_{\text{P4}}) &= k_{\text{CL}}^{\text{P4}} \cdot x_{\text{CL}} - c_{\text{P4}} \cdot x_{\text{P4}}, \\
h_{\text{E2}}(x_{\text{Foll}}, x_{\text{E2}}) &= k_{\text{Foll}}^{\text{E2}} \cdot x_{\text{Foll}} - c_{\text{E2}} \cdot x_{\text{E2}}, \\
h_{\text{Inh}}(x_{\text{Foll}}, x_{\text{Inh}}) &= k_{\text{Foll}}^{\text{Inh}} \cdot x_{\text{Foll}} - c_{\text{Inh}} \cdot x_{\text{Inh}}, \\
h_{\text{PGF}}(x_{\text{E2}}, x_{\text{P4}}, x_{\text{PGF}}) &= m_{\text{E2}}^{\text{PGF}} \cdot x_{\text{E2}} \cdot x_{\text{P4}} - c_{\text{PGF}} \cdot x_{\text{PGF}}, \\
h_{\text{IOF}}(x_{\text{PGF}}, x_{\text{CL}}, x_{\text{IOF}}) &= m_{\text{PGF,CL}}^{\text{IOF}} \cdot x_{\text{PGF}} \cdot x_{\text{CL}} - c_{\text{IOF}} \cdot x_{\text{IOF}}.
\end{aligned}$$

To obtain the Boolean update functions for all components, we now need to evaluate the signs of the function $h_i(x)$ for all Boolean states x . Apart from the component values x_j appearing on the right hand side of the function equations for each h_i , the value of $h_i(x)$ depends on the model parameter values, which are listed in the Materials and Methods section. The resulting truth tables for all Boolean functions are given in Tabs. 1 to 10.

Table 1: Updates for $x_{\text{GnRH}} \in \{0, 1\}$.

x_{P4}^k	x_{E2}^k	x_{GnRH}^k	h_{GnRH}	$\text{sgn}(h_{\text{GnRH}})$	x_{GnRH}^{k+1}
0	0	0	0	0	0
0	0	1	$-c_{\text{GnRH}}$	-1	0
0	1	0	$m_{\text{E2,P4}}^{\text{GnRH}}$	1	1
0	1	1	$m_{\text{E2,P4}}^{\text{GnRH}} - c_{\text{GnRH}}$	1	1
1	0	0	0	0	0
1	0	1	$-c_{\text{GnRH}}$	-1	0
1	1	0	0	0	0
1	1	1	$-c_{\text{GnRH}}$	-1	0

Table 2: Updates for $x_{\text{FSH}} \in \{0, 1\}$.

x_{Inh}^k	x_{FSH}^k	h_{FSH}	$\text{sgn}(h_{\text{FSH}})$	x_{FSH}^{k+1}
0	0	$m_{\text{Inh}}^{\text{FSH}}$	1	1
0	1	$m_{\text{Inh}}^{\text{FSH}} - c_{\text{FSH}}$	1	1
1	0	0	0	0
1	1	$-c_{\text{FSH}}$	-1	0

Step 2: Validation of the Boolean Model. An important feature of the underlying biological system is the oscillation of all components signifying the successive estrous cycles. Simulations of the ODE model reproduce this behavior. Now, to check whether the Boolean model captures the behavior of the numerical model, we decided to validate it using the simulations from the latter. This allows us to illustrate approaches to data discretization and to checking conservation of trajectories similarly applicable to validation with experimental data. In a first step, we derive

Table 3: Updates for $x_{\text{LH}} \in \{0, 1\}$.

x_{P4}^k	x_{GnRH}^k	x_{LH}^k	h_{LH}	$\text{sgn}(h_{\text{LH}})$	x_{LH}^{k+1}
0	0	0	0	0	0
0	0	1	$-c_{\text{LH}}$	-1	0
0	1	0	$m_{\text{GnRH,P4}}^{\text{LH}}$	1	1
0	1	1	$m_{\text{GnRH,P4}}^{\text{LH}} - c_{\text{LH}}$	1	1
1	0	0	0	0	0
1	0	1	$-c_{\text{LH}}$	-1	0
1	1	0	0	0	0
1	1	1	$-c_{\text{LH}}$	-1	0

Table 4: Updates for $x_{\text{Foll}} \in \{0, 1\}$.

x_{FSH}^k	x_{P4}^k	x_{LH}^k	x_{Foll}^k	h_{Foll}	$\text{sgn}(h_{\text{Foll}})$	x_{Foll}^{k+1}
0	0	0	0	0	0	0
0	0	0	1	0	0	1
0	0	1	0	0	0	0
0	0	1	1	$-m_{\text{LH}}^{\text{Foll}}$	-1	0
0	1	0	0	0	0	0
0	1	0	1	$-m_{\text{P4}}^{\text{Foll}}$	-1	0
0	1	1	0	0	0	0
0	1	1	1	$-m_{\text{P4}}^{\text{Foll}} - m_{\text{LH}}^{\text{Foll}}$	-1	0
1	0	0	0	$m_{\text{FSH}}^{\text{Foll}}$	1	1
1	0	0	1	$m_{\text{FSH}}^{\text{Foll}}(1 + m_{\text{Foll}}^{\text{Foll}})$	1	1
1	0	1	0	$m_{\text{FSH}}^{\text{Foll}}$	1	1
1	0	1	1	$m_{\text{FSH}}^{\text{Foll}}(1 + m_{\text{Foll}}^{\text{Foll}}) - m_{\text{LH}}^{\text{Foll}}$	-1	0
1	1	0	0	$m_{\text{FSH}}^{\text{Foll}}$	1	1
1	1	0	1	$m_{\text{FSH}}^{\text{Foll}}(1 + m_{\text{Foll}}^{\text{Foll}}) - m_{\text{P4}}^{\text{Foll}}$	1	1
1	1	1	0	$m_{\text{FSH}}^{\text{Foll}}$	1	1
1	1	1	1	$m_{\text{FSH}}^{\text{Foll}}(1 + m_{\text{Foll}}^{\text{Foll}}) - m_{\text{P4}}^{\text{Foll}} - m_{\text{LH}}^{\text{Foll}}$	-1	0

a discrete time series from the ODE simulation. Secondly, we search the asynchronous state transition graph of the Boolean model for a path matching this time series.

We start by running a forward simulation of the ODE model given in the Materials and Methods section, thus providing an approximation of the continuous solution as a time series of real-valued variables. As mentioned, several discretization schemes are available for converting continuous into Boolean data. Here, a threshold is chosen for each continuous variable y_i . Depending on whether the continuous variable is below or above this threshold at a certain point in time, the value 0 or 1 is assigned to the corresponding binary variable x_i for this time point. In the following, the thresholds are called *translation thresholds*.

Hill functions explicitly contain threshold values for some of the continuous variables. In most cases, however, the same continuous variable occurs in several Hill functions, but with different threshold values. Thus, some averaging must be applied. In addition, some variables do not exert their influence via Hill functions but directly, e.g., in terms of mass action kinetics. For these variables, their mean value over time can be used as threshold. Finally, the following rules are applied to derive the translation thresholds.

- (i) For each variable, its average value over time is calculated.

Table 5: Updates for $x_{\text{CL}} \in \{0, 1\}$.

x_{LH}^k	x_{Foll}^k	x_{IOF}^k	x_{CL}^k	h_{CL}	$\text{sgn}(h_{\text{CL}})$	x_{CL}^{k+1}
0	0	0	0	0	0	0
0	0	0	1	$m_{\text{CL}}^{\text{CL}}$	1	1
0	0	1	0	0	0	0
0	0	1	1	$-m_{\text{IOF}}^{\text{CL}}$	-1	0
0	1	0	0	0	0	0
0	1	0	1	$m_{\text{CL}}^{\text{CL}}$	1	1
0	1	1	0	0	0	0
0	1	1	1	$m_{\text{CL}}^{\text{CL}} - m_{\text{IOF}}^{\text{CL}}$	-1	0
1	0	0	0	0	0	0
1	0	0	1	$m_{\text{CL}}^{\text{CL}}$	1	1
1	0	1	0	0	0	0
1	0	1	1	$m_{\text{CL}}^{\text{CL}} - m_{\text{IOF}}^{\text{CL}}$	-1	0
1	1	0	0	$\text{SF}_{\text{CL}} \cdot m_{\text{LH}}^{\text{Foll}}$	1	1
1	1	0	1	$\text{SF}_{\text{CL}} \cdot m_{\text{LH}}^{\text{Foll}} + m_{\text{CL}}^{\text{CL}}$	1	1
1	1	1	0	$\text{SF}_{\text{CL}} \cdot m_{\text{LH}}^{\text{Foll}}$	1	1
1	1	1	1	$\text{SF}_{\text{CL}} \cdot m_{\text{LH}}^{\text{Foll}} + m_{\text{CL}}^{\text{CL}} - m_{\text{IOF}}^{\text{CL}}$	-1	0

Table 6: Updates for $x_{\text{P4}} \in \{0, 1\}$.

x_{CL}^k	x_{P4}^k	h_{P4}	$\text{sgn}(h_{\text{P4}})$	x_{P4}^{k+1}
0	0	0	0	0
0	1	$-c_{\text{P4}}$	-1	0
1	0	$k_{\text{CL}}^{\text{P4}}$	1	1
1	1	$k_{\text{CL}}^{\text{P4}} - c_{\text{P4}}$	1	1

Table 7: Updates for $x_{\text{E2}} \in \{0, 1\}$.

x_{Foll}^k	x_{E2}^k	h_{E2}	$\text{sgn}(h_{\text{E2}})$	x_{E2}^{k+1}
0	0	0	0	0
0	1	$-c_{\text{E2}}$	-1	0
1	0	$k_{\text{Foll}}^{\text{E2}}$	1	1
1	1	$k_{\text{Foll}}^{\text{E2}} - c_{\text{E2}}$	1	1

Table 8: Updates for $x_{\text{Inh}} \in \{0, 1\}$.

x_{Foll}^k	x_{Inh}^k	h_{Inh}	$\text{sgn}(h_{\text{E2}})$	x_{Inh}^{k+1}
0	0	0	0	0
0	1	$-c_{\text{Inh}}$	-1	0
1	0	$k_{\text{Foll}}^{\text{Inh}}$	1	1
1	1	$k_{\text{Foll}}^{\text{Inh}} - c_{\text{Inh}}$	1	1

- (ii) For each variable that exerts an influence via at least one Hill function, the mean of the thresholds in these Hill functions is calculated.

Table 9: Updates for $x_{\mathbf{PGF}} \in \{0, 1\}$

$x_{\mathbf{E2}}^k$	$x_{\mathbf{P4}}^k$	$x_{\mathbf{PGF}}^k$	$h_{\mathbf{PGF}}$	$\text{sgn}(h_{\mathbf{PGF}})$	$x_{\mathbf{PGF}}^{k+1}$
0	0	0	0	0	0
0	0	1	$-c_{\mathbf{PGF}}$	-1	0
0	1	0	0	0	0
0	1	1	$-c_{\mathbf{PGF}}$	-1	0
1	0	0	0	0	0
1	0	1	$-c_{\mathbf{PGF}}$	-1	0
1	1	0	$m_{\mathbf{E2}}^{\mathbf{PGF}}$	1	1
1	1	1	$m_{\mathbf{E2}}^{\mathbf{PGF}} - c_{\mathbf{PGF}}$	-1	1

Table 10: Updates for $x_{\mathbf{IOF}} \in \{0, 1\}$.

$x_{\mathbf{PGF}}^k$	$x_{\mathbf{CL}}^k$	$x_{\mathbf{IOF}}^k$	$h_{\mathbf{IOF}}$	$\text{sgn}(h_{\mathbf{IOF}})$	$x_{\mathbf{IOF}}^{k+1}$
0	0	0	0	0	0
0	0	1	$-c_{\mathbf{IOF}}$	-1	0
0	1	0	0	0	0
0	1	1	$-c_{\mathbf{IOF}}$	-1	0
1	0	0	0	0	0
1	0	1	$-c_{\mathbf{IOF}}$	-1	0
1	1	0	$m_{\mathbf{PGF,CL}}^{\mathbf{IOF}}$	1	1
1	1	1	$m_{\mathbf{PGF,CL}}^{\mathbf{IOF}} - c_{\mathbf{IOF}}$	-1	1

- (iii) The mean between the values from (i) and (ii) or, if (ii) does not apply, only the value from (i) is used as translation threshold.

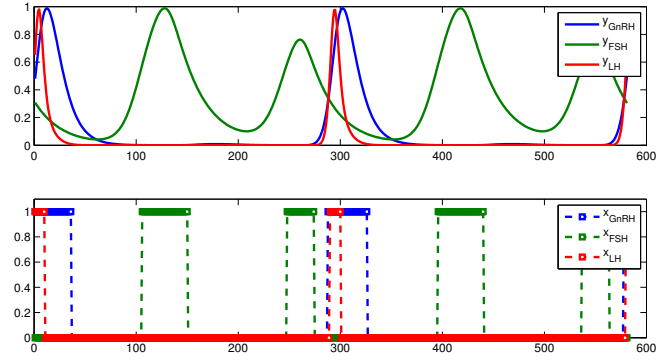
Averaging in (iii) is reasonable because a Hill-threshold does not need to be fully reached by a continuous variable in order to have a regulatory impact on the other variables. Based on these translation thresholds, the binary time series is generated, and repeating entries are removed. The derived sequence of states is depicted in Fig. 1, and explicitly given in Tab. 11. Double-checking the discretization result, we also find that it matches the observed qualitative behavior of the underlying biological system.

To check whether this state sequence is generated by the Boolean model, we search for a path in the state transition graph that visits the given states in the correct order. Hereby, we allow for intermediate states since not every system state might be captured by the time series. This can be done, e.g., by using a path search in GINsim [16] or via model checking as in [14]. We find indeed that such a path exists, i.e., the Boolean model is capable of generating the desired behavior.

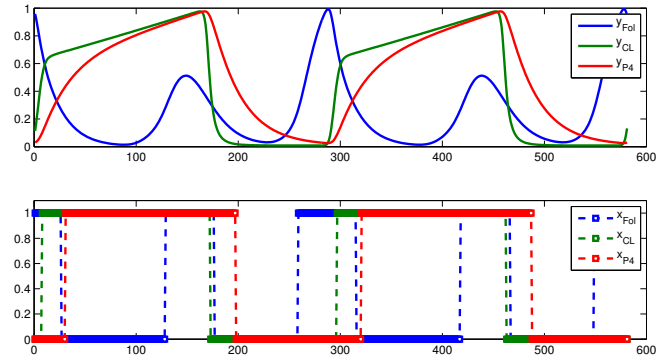
Step 3: Analysis of the Boolean Model. We start with a structural analysis of the Boolean model, having a closer look at the dependency graph (Fig. 2) which is directly derived from the listed truth tables of the Boolean functions. Looking at the equations for the functions h_i derived in Step 1, we see the potential for self-regulation for each component in the decay terms incorporated in the differential equations.

However, a closer look at the truth tables shows that these decay terms in almost all cases need not be explicitly modeled in the Boolean setting. As an example consider the regulation of FSH, where in the left two columns of Table 2 the two variables Inh and FSH as given in the ODE are listed. However, looking at the rightmost column and comparing the first two with the last two rows, we can easily see that the value of the FSH update function only depends on the value of Inh. The decay term included in the FSH ODE thus has no functional impact on this level of abstraction.

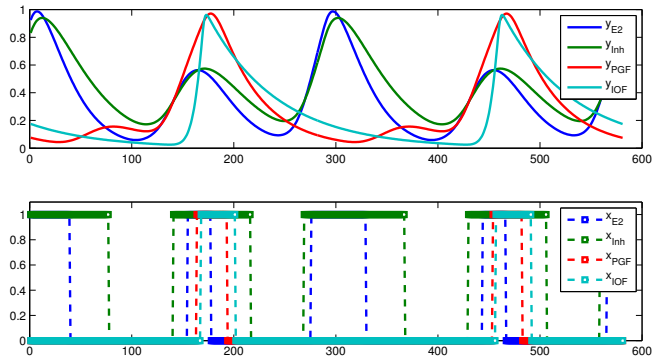
There are two exceptions to this observation, the components Foll and CL. Further analysis



(a) Simulation output for GnRH, FSH, and LH



(b) Simulation output for Fol, CL, and P4



(c) Simulation output for E2, Inh, PGF, and IOF

Figure 1: ODE simulation from the model *BovCycle* and the generated discrete time series. This time series is used to create a sequence of binary states for the validation of the discrete model.

Table 11: **Discrete sequence of states generated from the time series of the continuous model.**

x_{GnRH}	x_{FSH}	x_{LH}	x_{Foll}	x_{CL}	x_{P4}	x_{E2}	x_{Inh}	x_{PGF}	x_{IOF}	Event
1	0	1	1	0	0	1	1	0	0	$x_{\text{CL}} \uparrow$
1	0	1	1	1	0	1	1	0	0	$x_{\text{LH}} \downarrow$
1	0	0	1	1	0	1	1	0	0	$x_{\text{Foll}} \downarrow$
1	0	0	0	1	0	1	1	0	0	$x_{\text{P4}} \uparrow$
1	0	0	0	1	1	1	1	0	0	$x_{\text{GnRH}} \downarrow$
0	0	0	0	1	1	1	1	0	0	$x_{\text{E2}} \downarrow$
0	0	0	0	1	1	0	1	0	0	$x_{\text{Inh}} \downarrow$
0	0	0	0	1	1	0	0	0	0	$x_{\text{FSH}} \uparrow$
0	1	0	0	1	1	0	0	0	0	$x_{\text{Foll}} \uparrow$
0	1	0	1	1	1	0	0	0	0	$x_{\text{Inh}} \uparrow$
0	1	0	1	1	1	0	1	0	0	$x_{\text{FSH}} \downarrow$
0	0	0	1	1	1	0	1	0	0	$x_{\text{E2}} \uparrow$
0	0	0	1	1	1	1	1	0	0	$x_{\text{PGF}} \uparrow$
0	0	0	1	1	1	1	1	1	0	$x_{\text{IOF}} \uparrow$
0	0	0	1	1	1	1	1	1	1	$x_{\text{CL}} \downarrow$
0	0	0	1	0	1	1	1	1	1	$x_{\text{Foll}} \downarrow$
0	0	0	0	0	1	1	1	1	1	$x_{\text{E2}} \downarrow$
0	0	0	0	0	1	0	1	1	1	$x_{\text{PGF}} \downarrow$
0	0	0	0	0	1	0	1	0	1	$x_{\text{P4}} \downarrow$
0	0	0	0	0	0	0	1	0	1	$x_{\text{IOF}} \downarrow$
0	0	0	0	0	0	0	1	0	0	$x_{\text{Inh}} \downarrow$
0	0	0	0	0	0	0	0	0	0	$x_{\text{FSH}} \uparrow$
0	1	0	0	0	0	0	0	0	0	$x_{\text{Foll}} \uparrow$
0	1	0	1	0	0	0	0	0	0	$x_{\text{Inh}} \uparrow$
0	1	0	1	0	0	0	1	0	0	$x_{\text{FSH}} \downarrow$
0	0	0	1	0	0	0	1	0	0	$x_{\text{E2}} \uparrow$
0	0	0	1	0	0	1	1	0	0	$x_{\text{GnRH}} \uparrow$
1	0	0	1	0	0	1	1	0	0	$x_{\text{LH}} \uparrow$
1	0	1	1	0	0	1	1	0	0	

shows that both exhibit an observable self-activating effect, which again can already be seen in the respective differential equations. Interestingly, both also display a self-inhibition, e.g., in the state where all their other regulators are active, as can be seen in Tabs. 4 and 5. This indicates that certain mechanisms driving the system function potentially rely on these self-inhibitory effects that might be implemented either as an active inhibition or via a simple decay.

Next, we have a closer look at the shortest path in the state transition graph traversing the time series given in Tab. 11. Most of the given states can be reached by a direct transition from the preceding state, meaning there is an edge between these two states in the state transition graph. Exceptions are the second and the third transition in Tab. 11. Although the states also only differ in one component value, it takes several intermediate steps to get from one state to the next, i.e., there are only paths of length greater than 1 linking the states in the state transition graph. In the second transition, e.g., we want to see a down-regulation of x_{LH} . Comparing the current state with the truth table of the x_{LH} -update function, we see that to obtain this effect first down-regulation of x_{GnRH} or up-regulation of x_{P4} needs to occur, which in turn is driven by other regulators. Activity level changes in components other than x_{LH} need then to be reversed after adapting the activity of x_{LH} to match the given measurement. Thus, we uncovered a hidden oscillation of a subnetwork driving the system along the observed trajectory. Similar results can be obtained for the down-regulation of x_{Foll} in the third transition.

Lastly, we want to address a phenomenon observed in the continuous setting. A remarkable

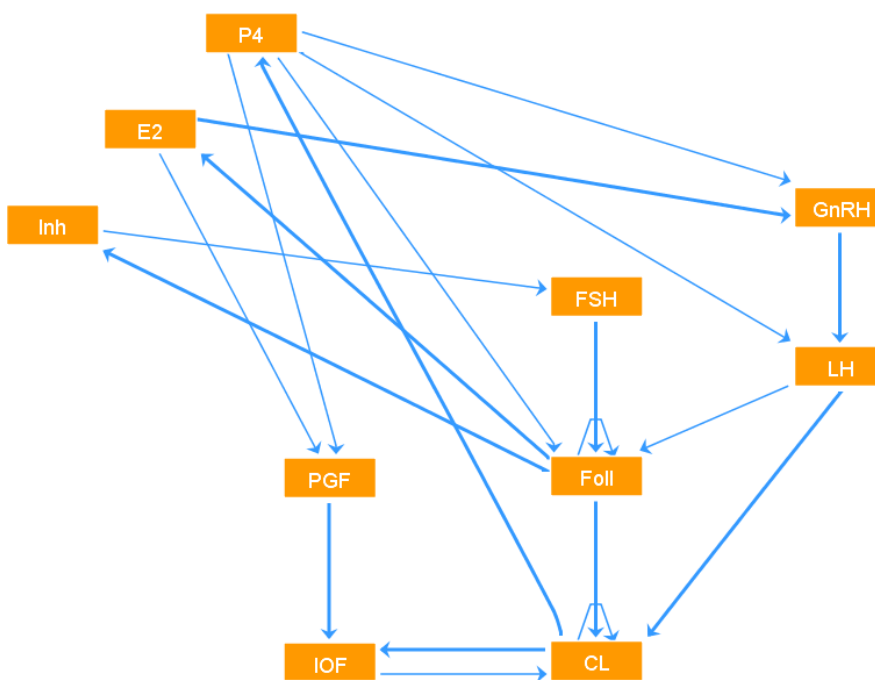


Figure 2: Dependency graph of the Boolean model for the bovine estrous cycle. This graph directly follows from the discrete updates calculated in Step 1, which are derived from the right hand sides of the ODE model. Implemented in GINsim [16].

property of the ODE model for the bovine estrous cycle is the stability of the limit cycle solution with respect to perturbations of state variables and parameter values. For the continuous case, local stability has been proven [21], but the proof cannot be extended to show global stability. For the discrete model, however, we can perform a more global analysis.

To this end, we conducted an attractor analysis considering the state transition graph of the model in its entirety using the software GINsim [16]. As a first result, we found that the network does not have any steady states. This already indicates strongly that the system is tailored to generate stable oscillatory behavior.

Further analysis shows that the system has a single attractor of high complexity, which contains the time series given in Tab. 11 encoding the normal estrous cycle. Since the entire state space constitutes the basin of attraction for this attractor, the cyclic path encoding the time series is reachable from every state of the system. This again indicates a high degree of stability, since the system can recover its standard behavior after an arbitrary state perturbation.

The analysis steps shown here already yield several interesting results that could be investigated much further. A circuit functionality analysis [4] could shed light on the question of why there is only a single attractor and provide a better understanding of what modules of the network may be responsible for certain characteristics of the stable oscillation. In other applications, one could also investigate the influence of e.g. knock-outs/ins on the model behavior to identify key components. A comparative analysis of such scenarios of the logical and ODE model might also further clarify the relation between the two models.

Materials and Methods

In this section, we introduce the concepts and notations of ordinary differential equation models and logical models. In addition, we briefly introduce the software GINsim that is used for validation of the logical model. Moreover, we present the equations and parameter values for the model BovCycle, which is used as illustrative example in the paper.

ODE Modeling

An ordinary differential equation (ODE) model is based on a state-vector $y(t) = (y_1(t), \dots, y_n(t)) \in \mathbb{R}^n$ of time-dependent, continuous variables representing the concentrations or activation levels of certain system components, e.g., chemical compounds, populations of cells or animals. Their change over time is described by the first derivatives, $y'(t) = f(t, y, p)$, modeled by a parameter-dependent function $f(t, y, p) = (f_1(t, y, p), \dots, f_n(t, y, p))$. Here $p \in \mathbb{R}^q$ denotes the vector of (unknown) model parameters values.

ODEs have become a standard tool in systems biology to model complex systems enriched in non-linear motifs like feedback and feed-forward loops [24]. Construction and analysis of ODE models for such systems, however, face a number of difficulties. Two fundamental questions have to be addressed: the derivation of model equations and the estimation of model parameters. For the construction of model equations, canonical modeling frameworks like mass-action kinetic, Hill kinetics or power-law modeling are often used.

Parameter estimation, however, is a computationally intensive task that requires large amounts of quantitative experimental data (dose-response-experiments, time series data, etc.) and sophisticated optimization algorithms. Alternative methods for exploring the dynamical behavior of ODE models are therefore desirable.

Logical Modeling

A logical network model is based on a set of discrete variables $\{x_1, \dots, x_n\}$, $n \in \mathbb{N}$, representing the network components $\{1, \dots, n\}$. Each variable x_i can take values in a set $\{0, 1, \dots, m_i\}$, $m_i \in \mathbb{N}$, of integers interpreted as the different activity levels of the component. In the simplest case, all variables only take values 0 and 1, resulting in a Boolean network. The state space of

the network is given by the set $S := \{0, \dots, m_1\} \times \dots \times \{0, \dots, m_n\}$. All interactions between the different components generating the system behavior are then modeled by a logical function $g = (g_1, \dots, g_n) : S \rightarrow S$, where each coordinate function g_i captures the way component x_i is regulated in a given system state. The dependencies of the coordinate functions on particular components are captured in a dependency or interaction graph where the vertices represent the network components, and there is an edge from i to j if f_j depend on x_i . Edges can also carry a sign indicating whether the regulatory effect is positive or negative (see e.g. [27]). It indicates the existence of at least one network state, where the change in the regulator's activity level generates a similar, in the case of a positive interaction, or opposite, in the case of a negative interaction, effect on the target component.

The dynamics of the network can be represented by a finite directed state transition graph with vertex set S and edges representing state transitions derived from g according to a certain update strategy. The simplest strategy, called synchronous update, is to define the successor of a state s as $g(s)$. A strategy taking into account the different time delays associated to different regulatory events is the so-called asynchronous update where only one component value is changed per update step: In the Boolean case, we get that \hat{s} is a successor of s if there exists $i \in \{1, \dots, n\}$ such that $\hat{s}_i = g_i(s) \neq s_i$ and $\hat{s}_j = s_j$ for all $j \neq i$ [22]. This results in a non-deterministic state transition graph usually comprising more realistic trajectories than generated by the synchronous update.

Discrete Analysis Tools

All validation and analysis steps presented for the Boolean model of the bovine estrous cycle can be conducted using GINsim [16], a software for simulation and analysis of qualitative models. It is based on the multilevel logical formalism introduced in [22], which includes the Boolean formalism used in this paper. In GINsim, among other functionalities, the state transition graph of a model can be calculated and then analyzed using a number of tools. Included are efficient algorithms for the computation of steady states, shortest path search in the state transition graph and computation of the strongly connected component graph allowing to identify cyclic attractors and basins of attraction.

The ODE Model BovCycle

As in humans, the bovine endocrine system consists of several glands and regulates the periodic changes of multiple substances necessary for reproduction. In every cycle, hormones are secreted from the hypothalamic-pituitary-gonadal axis into the bloodstream, where they distribute and influence several functions in the body. Their most important task in reproduction is to regulate processes in the ovaries, where follicles and corpus luteum develop. These produce steroids that are released into the blood and from therein regulate the processes in the hypothalamic-pituitary-gonadal axis. The complex interplay of these multiple components generates the cyclic hormonal changes without external stimuli. The large feedback loop of regulations results in the hormonal cycle.

In the ODE model BovCycle, interactions between biological substances are mostly of regulatory nature and are modeled by Hill functions. An increasing Hill function stands for activation or stimulation, a decreasing Hill function for inhibition. Since Hill functions have a sigmoidal shape, their approximation by discrete variables in a logical model is a reasonable approach. The current model [21], which serves as an example for the discretization method in this paper, contains 10 ODEs and 38 parameters. Model equations and the here used parameter values are listed in Tabs. 12 and 13. The model represents a reduced version of a more complex model published in [2].

Table 12: **The model BovCycle consisting of 10 ODEs and 38 parameters.**

$$\begin{aligned}
\frac{d}{dt} y_{\text{GnRH}}(t) &= m_{\text{E2,P4}}^{\text{GnRH}} \cdot H_{\text{P4,GnRH}}^-(y_{\text{P4}}(t)) \cdot H_{\text{E2,GnRH}}^+(y_{\text{E2}}(t)) - c_{\text{GnRH}} \cdot y_{\text{GnRH}}(t) \\
\frac{d}{dt} y_{\text{FSH}}(t) &= m_{\text{Inh}}^{\text{FSH}} \cdot H_{\text{Inh,FSH}}^-(y_{\text{Inh}}(t)) - c_{\text{FSH}} \cdot y_{\text{FSH}}(t) \\
\frac{d}{dt} y_{\text{LH}}(t) &= m_{\text{GnRH,P4}}^{\text{LH}} \cdot H_{\text{P4,LH}}^-(y_{\text{P4}}(t)) \cdot H_{\text{GnRH,LH}}^+(y_{\text{GnRH}}(t)) - c_{\text{LH}} \cdot y_{\text{LH}}(t) \\
\frac{d}{dt} y_{\text{Foll}}(t) &= m_{\text{FSH}}^{\text{Foll}} \cdot H_{\text{FSH,Foll}}^+(y_{\text{FSH}}(t)) \cdot \left(1 + m_{\text{Foll}}^{\text{Foll}} \cdot H_{\text{Foll,Foll}}^+(y_{\text{Foll}}(t))\right) \\
&\quad - \left(m_{\text{P4}}^{\text{Foll}} \cdot H_{\text{P4,Foll}}^+(y_{\text{P4}}(t)) + m_{\text{LH}}^{\text{Foll}} \cdot H_{\text{LH,Ovul}}^+(y_{\text{LH}}(t))\right) \cdot y_{\text{Foll}}(t) \\
\frac{d}{dt} y_{\text{CL}}(t) &= \text{SF}_{\text{CL}} \cdot m_{\text{LH}}^{\text{Foll}} \cdot H_{\text{LH,Ovul}}^+(y_{\text{LH}}(t)) \cdot y_{\text{Foll}}(t) + m_{\text{CL}}^{\text{CL}} \cdot H_{\text{CL,CL}}^+(y_{\text{CL}}(t)) \\
&\quad - m_{\text{IOF}}^{\text{CL}} \cdot H_{\text{IOF,CL}}^+(y_{\text{IOF}}(t)) \cdot y_{\text{CL}}(t) \\
\frac{d}{dt} y_{\text{P4}}(t) &= k_{\text{CL}}^{\text{P4}} \cdot y_{\text{CL}}(t) - c_{\text{P4}} \cdot y_{\text{P4}}(t) \\
\frac{d}{dt} y_{\text{E2}}(t) &= k_{\text{Foll}}^{\text{E2}} \cdot y_{\text{Foll}}(t) - c_{\text{E2}} \cdot y_{\text{E2}}(t) \\
\frac{d}{dt} y_{\text{Inh}}(t) &= k_{\text{Foll}}^{\text{Inh}} \cdot y_{\text{Foll}}(t) - c_{\text{Inh}} \cdot y_{\text{Inh}}(t) \\
\frac{d}{dt} y_{\text{PGF}}(t) &= m_{\text{E2}}^{\text{PGF}} \cdot H_{\text{E2,PGF}}^+(y_{\text{E2}}(t)) \cdot H_{\text{P4,PGF}}^+(y_{\text{P4}}(t)) - c_{\text{PGF}} \cdot y_{\text{PGF}}(t) \\
\frac{d}{dt} y_{\text{IOF}}(t) &= m_{\text{PGF,CL}}^{\text{IOF}} \cdot H_{\text{PGF,IOF}}^+(y_{\text{PGF}}(t)) \cdot H_{\text{CL,IOF}}^+(y_{\text{CL}}(t)) - c_{\text{IOF}} \cdot y_{\text{IOF}}(t)
\end{aligned}$$

The notation for the Hill functions $H_{S_1, S_2}^{+/-}(y_{S_1}(t))$ is an abbreviation for $H^{+/-}(y_{S_1}(t), T_{S_1}^{S_2}, r_{S_1}^{S_2})$, whereby $H^-(y_{S_1}(t), T_{S_1}^{S_2}, r_{S_1}^{S_2}) = 1 - H^+(y_{S_1}(t), T_{S_1}^{S_2}, r_{S_1}^{S_2})$.

Discussion

In this paper, we presented a generic method for translating an ODE into a Boolean model based on an Euler approximation of the continuous behavior. This straightforward approach is augmented by a subsequent mandatory validation step ensuring that the Boolean network is suitable for describing the properties of interest, either with respect to the ODE model or the underlying biological system. Analysis then allows to address questions not easily accessible in the context of an elaborate ODE model and provides an alternative view on system characteristics, as we illustrated in our case study.

The Euler step is executed and evaluated to derive the qualitative update functions from the ODEs. Mathematically, it captures the behavioral tendencies of the continuous systems in a given state via the derivative signs. Normalization and monotonicity constraints of the continuous regulatory functions ensure that, regardless of the step size, these tendencies capture a long-term effect rather than some initial fluctuations in the transient phase. To account for the fact that, in a continuous system of interacting components, the asymptotic component behavior does not always play out along a trajectory due to regulatory influences, we chose the asynchronous update for the resulting Boolean models. Here, a component update is not necessarily carried out on a specific trajectory if other component updates compete. Based on these observations, the derived Boolean model can be expected to capture the qualitative aspects of its ODE counterpart. However, property conservation is not guaranteed. The validation step of our procedure is necessary to double-check the behavioral match for the dynamical phenomenon of interest. Depending on the modeling objective, the validation strategies presented here could be complemented by other methods, e.g. an explicit analysis on conservation of transition sequences in both the ODE and the Boolean model.

In the Supporting Information, we present an alternative method to transform a system of ODEs fulfilling the preliminaries of Step 0 into a Boolean network. This method is inspired by a translation scheme presented in [5] where a specific coarse-grained limit of an ODE system leads to the formulation of a Boolean network. In [5], the results of a bifurcation analysis are used to derive a sequence of stationary states. In the stationary states, variables can be directly described by the Goldbeter-Koshland function, a function with two limiting states. Its approximation by Heaviside functions allows to derive an interaction matrix. Together with a state vector of thresholds, this determines the transition rules between the states and leads to a system whose trajectory is a sequence of Boolean states.

Table 13: BovCycle model parameters and their values which lead to the simulation of an estrous cycle with two follicular waves per cycle.

No.	Symbol	Value	Unit
1	$m_{E2,P4}^{GnRH}$	5.707	[GnRH]/[t]
2	T_{P4}^{GnRH}	0.271	[P4]
3	T_{E2}^{GnRH}	1.127	[E2]
4	c_{GnRH}	1.223	1/[t]
5	m_{Inh}^{FSH}	1.044	[FSH]/[t]
6	T_{Inh}^{FSH}	0.217	[Inh]
7	c_{FSH}	0.559	1/[t]
8	$m_{GnRH,P4}^{LH}$	46.647	[LH]/[t]
9	T_{P4}^{LH}	0.0542	[P4]
10	T_{GnRH}^{LH}	0.896	[GnRH]
11	c_{LH}	9.006	1/[t]
12	m_{FSH}^{Foll}	0.269	[Foll]/[t]
13	T_{FSH}^{Foll}	0.787	[FSH]
14	m_{Foll}	3.927	-
15	T_{Foll}	0.289	[Foll]
16	m_{P4}^{Foll}	0.79	1/[t]
17	T_{P4}^{Foll}	0.125	[P4]
18	m_{LH}^{Foll}	1.7	[1]/[t]
19	T_{LH}^{Foll}	0.881	[LH]
20	SF_{CL}	1.2	[CL]
21	m_{CL}^{CL}	0.0372	[CL]/[t]
22	T_{CL}^{CL}	0.314	[CL]
23	m_{IOF}^{CL}	7.534	1/[t]
24	T_{IOF}^{CL}	1.035	[IOF]
25	k_{CL}^{P4}	0.564	[P4]/[t]
26	c_{P4}	0.533	1/[t]
27	k_{Foll}^{E2}	1.009	[E2]/[t]
28	c_{E2}	0.72	1/[t]
29	k_{Foll}^{Inh}	0.644	[Inh]/[t]
30	c_{Inh}	0.368	1/[t]
31	m_{E2}^{PGF}	1.291	[PGF]/[t]
32	T_{E2}^{PGF}	0.221	[E2]
33	T_{P4}^{PGF}	0.969	[P4]
34	c_{PGF}	0.356	1/[t]
35	$m_{PGF,CL}^{IOF}$	12.269	[IOF]/[t]
36	T_{PGF}^{IOF}	1.282	[PGF]
37	T_{CL}^{IOF}	0.639	[CL]
38	c_{IOF}	0.215	1/[t]

Hill exponents are fixed as $r_{E2}^{GnRH} = r_{Inh}^{FSH} = r_{P4}^{Foll} = r_{IOF}^{CL} = r_{P4}^{PGF} = r_{PGF}^{IOF} = r_{CL}^{IOF} = 5$, all other Hill exponents are set to 2.

Our method described in the Supporting Information first approximates the unregulated variables or functions in the right hand-sides as integers. Instead of considering bifurcation points, our alternative method takes into account all possible combinations of integers. Due to the approximation, for all cases, the variables can be directly described by linear ODEs which are solved analytically. Then, to overcome the time dependency, the time limit of the system is considered in order to derive the update functions. We show that, for any system of ODEs fulfilling the preliminaries, the therein proposed method leads to the same Boolean network as the described Euler-like transformation.

From the significant abstraction step it is clear that the resulting Boolean network cannot capture all intricacies of the ODE model, and even much less of the underlying biological system. The lack of rigorous proofs for property conservation across formalisms might be considered as main limitation of our proposed method. However, such proofs would certainly require much more severe restrictions on the class of ODE models to be transformed. Here, our aim was to show the easy applicability of the proposed procedure to a broad class of models.

Due to the above considerations, the validation step is of major importance to verify that key aspects of the system are reproduced and should be handled with care. In particular, preprocessing of the data, e.g. discretization of quantitative measurements, can influence the results [6]. The choice of a suitable preprocessing technique and subsequent evaluation, e.g. a sensitivity analysis, can contribute to ensure rigorous results. Generally, a closer look at the reason for a possibly failed validation step can lead to uncover crucial mechanisms not yet captured in the model. Comparison of the validated ODE model with the derived logical model may also help to uncover quantitative properties of functional importance for the biological regulation mechanisms.

In future research, our proposed approach can be adapted and extended to achieve a better fit. A first easy alteration could concern the evaluation of the regulatory effects within the Euler step. Here, one can introduce a threshold for positive and negative tendencies to become effective in the qualitative setting rather than to factor in each non-zero effect.

Furthermore, a higher resolution of activity levels might be very useful, leading to multi-valued instead of Boolean models. For this, the Euler step could be adapted in a way that the update function, instead of directly calculating the next value for a component, only provides the augmentation of the component, which is set to -1 in case of negative derivative of the right hand side, and zero in case that the derivative is zero. Additionally, one could define the update function such that it depends not only on the sign but also, with the help of thresholds, on the value of the derivative. Provided that a reasonable validation is applied, the multi-valued formalism might then allow functionally necessary complexity while keeping the available advantages for analysis.

Supporting Information

S1 An Alternative Discretization Method

In our paper, we presented a method for translating an ODE model into a discrete model. The update functions in the discrete model were derived from an Euler-like step. In this section, we describe an alternative discretization method that leads to the same update functions but allows for a different interpretation, as addressed in the discussion of the paper.

Discretization by Limit Considerations. Similar to the method described in the paper, continuous variables y_1, \dots, y_n in the right hand side function $f(y)$ are replaced by their discrete counterparts x_1, \dots, x_n to map the right hand sides $f_i(y)$ in (1) to their discrete counterparts $\hat{h}_i(x)$. This time, however, an exception is made for variables y_i in the right hand side functions $f_i(y)$: monotonically increasing functions $F_{i,i}(y_i)$ are replaced by y_i , whereas monotonically decreasing functions $F_{i,i}(y_i)$ are replaced by $1 - y_i$. The mapping thus takes the form

$$f_i(F_{i,1}(y_1), \dots, F_{i,n}(y_n)) \mapsto f_i(\underbrace{TF_{i,1}(x_1)}_{x_1 \text{ or } 1-x_1}, \dots, \underbrace{TF_{i,i}(y_i)}_{y_i \text{ or } 1-y_i}, \dots, \underbrace{TF_{i,n}(x_n)}_{x_n \text{ or } 1-x_n}) =: \hat{h}_i(x_1, \dots, y_i, \dots, x_n)$$

for $i = 1, \dots, n$. The right hand sides $f_i : \mathbb{R}^n \rightarrow \mathbb{R}$ of the ODE system are thus mapped to functions $\hat{h}_i : \{0, 1\}^{n-1} \times \mathbb{R} \rightarrow \mathbb{R}$, $i = 1, \dots, n$. This leads to n linear ODEs in the form

$$y_i' = \hat{h}_i(x_1, \dots, x_{i-1}, y_i, x_{i+1}, \dots, x_n) = c_{i1} - c_{i2} \cdot y_i, \quad y_i(t_0) = y_i^0, \quad (3)$$

with constant terms c_{i1} and c_{i2} . These constants depend on the values of the discrete variables $x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_n$ and the model parameters p .

These linear ODEs are now solved analytically and, for all combinations of values c_{i1} and c_{i2} , the limit as $t \rightarrow \infty$ is considered to derive the update rule. The initial value y_i^0 is identified with the current state of the discrete system, x_i^k . In particular, for $c_{i2} \neq 0$, the analytic solution for such a linear ODE is given by

$$y_i(t) = \frac{c_{i1}}{c_{i2}} + \left(y_i(t_0) - \frac{c_{i1}}{c_{i2}} \right) \exp(-c_{i2}(t - t_0)).$$

For $c_{i2} = 0$, we have $y_i' = c_{i1}$, and thus the analytic solution

$$y_i(t) = c_{i1}(t - t_0) + y_i(t_0).$$

Now, to get rid of the time-dependence, we consider the limit of $y_i(t)$ as t tends to infinity. Depending on the values of c_{i1} , c_{i2} (smaller than, equal to, or larger than zero), and $y_i(t_0)$ (0 or 1), 18 different cases must be considered, compare Tab. 14. The update rule for the Boolean variable x_i^k in step $k + 1$ is then chosen according to

$$x_i^{k+1} = \begin{cases} 1 & , \text{ if } \lim_{t \rightarrow \infty} y_i(t) > x_i^k, \\ x_i^k & , \text{ if } \lim_{t \rightarrow \infty} y_i(t) = y_i(t_0) = x_i^k, \\ 0 & , \text{ if } \lim_{t \rightarrow \infty} y_i(t) < x_i^k. \end{cases} \quad (4)$$

Comparison with the Euler Method Note that in the Euler method, the right hand sides $f_i : \mathbb{R}^n \rightarrow \mathbb{R}$ of the ODE system are mapped to functions $h_i : \{0, 1\}^n \rightarrow \mathbb{R}$. For comparison with the alternative method, these functions can be written as

$$h_i(x_1, \dots, x_n) = c_{i1} - c_{i2} \cdot x_i,$$

where the constant terms c_{i1} and c_{i2} are the same as above.

In Tab. 14, the transformation from an ODE to a discrete model is performed with both methods. One can observe that, in any case, both methods lead to the same update. Thus, for the ODE models that fulfill the assumptions described in Step 0 of our paper, the two discretization methods lead to the same binary model.

Table 14: Comparison of the results of the two discretization methods for the update rule of a discrete variable x^k .

Cases	Limit method	Euler method
$c_1 \quad c_2 \quad y(t_0)$ or x^k	$y(t) = \begin{cases} \frac{c_1}{c_2} + \left(y(t_0) - \frac{c_1}{c_2}\right) e^{-c_2(t-t_0)} \\ c_1(t-t_0) + y(t_0), \text{ if } c_2 = 0 \end{cases}$	$x_h^{k+1} \quad h(x^k) \quad \text{sgn}(h(x^k)) \quad x_h^{k+1}$
$0 \quad 0$ 1	$y(t) = y(t_0) \begin{cases} = 0 \\ = 1 \end{cases}$	$0 \quad 0 \quad 0$ $1 \quad 0 \quad 0 \quad 1$
$0 > 0$ 1	$y(t) = y(t_0) e^{-c_2(t-t_0)} \xrightarrow{t \rightarrow \infty} 0$	$0 \quad 0 \quad 0$ $0 \quad -c_2 \quad < 0 \quad 0$
$0 < 0$ 1	$y(t) = y(t_0) e^{-c_2(t-t_0)} \begin{cases} = 0 \\ \xrightarrow{t \rightarrow \infty} \infty \end{cases}$	$0 \quad 0 \quad 0$ $1 \quad -c_2 \quad > 0 \quad 1$
$> 0 \quad 0$ 1	$y(t) = c_1(t-t_0) + y(t_0) \xrightarrow{t \rightarrow \infty} \infty$	$1 \quad c_1 \quad > 0 \quad 1$ $1 \quad c_1 \quad > 0 \quad 1$
$> 0 > 0$ 1	$\frac{c_1}{c_2} + \left(y(t_0) - \frac{c_1}{c_2}\right) e^{-c_2(t-t_0)} \xrightarrow{t \rightarrow \infty} \frac{c_1}{c_2} \begin{cases} > 0 \\ = 1, \quad c_1 = c_2 \\ > 1, \quad c_1 > c_2 \\ < 1, \quad c_1 < c_2 \end{cases}$	$1 \quad c_1 \quad > 0 \quad 1$ $\begin{cases} 1 \\ 1 \\ 0 \end{cases} \quad c_1 - c_2 \begin{cases} = 0, \quad c_1 = c_2 \\ > 0, \quad c_1 > c_2 \\ < 0, \quad c_1 < c_2 \end{cases} \begin{cases} 1 \\ 1 \\ 0 \end{cases}$
$> 0 < 0$ 1	$y(t) = \frac{c_1}{c_2} + \underbrace{\left(y(t_0) - \frac{c_1}{c_2}\right)}_{> 0} e^{-c_2(t-t_0)} \xrightarrow{t \rightarrow \infty} \infty$	$1 \quad c_1 \quad > 0 \quad 1$ $1 \quad c_1 - c_2 \quad > 0 \quad 1$
$< 0 \quad 0$ 1	$y(t) = c_1(t-t_0) + y(t_0) \xrightarrow{t \rightarrow \infty} -\infty$	$0 \quad c_1 \quad < 0 \quad 0$ $0 \quad c_1 \quad < 0 \quad 0$
$< 0 > 0$ 1	$y(t) = \frac{c_1}{c_2} + \left(y(t_0) - \frac{c_1}{c_2}\right) e^{-c_2(t-t_0)} \xrightarrow{t \rightarrow \infty} -\left \frac{c_1}{c_2}\right $	$0 \quad c_1 \quad < 0 \quad 0$ $0 \quad c_1 - c_2 \quad < 0 \quad 0$
$< 0 < 0$ 1	$\frac{c_1}{c_2} + \left(y(t_0) - \frac{c_1}{c_2}\right) e^{-c_2(t-t_0)} \begin{cases} \xrightarrow{t \rightarrow \infty} -\infty \\ \begin{cases} \xrightarrow{t \rightarrow \infty} \frac{c_1}{c_2} = 1, \quad c_1 = c_2 \\ \xrightarrow{t \rightarrow \infty} -\infty, \quad c_1 > c_2 \\ \xrightarrow{t \rightarrow \infty} \infty, \quad c_1 < c_2 \end{cases} \end{cases}$	$0 \quad c_1 \quad < 0 \quad 0$ $\begin{cases} 1 \\ 0 \\ 1 \end{cases} \quad c_1 - c_2 \begin{cases} = 0, \quad c_1 = c_2 \\ < 0, \quad c_1 > c_2 \\ > 0, \quad c_1 < c_2 \end{cases} \begin{cases} 1 \\ 0 \\ 1 \end{cases}$

References

- [1] ABOU-JAOUDE, W., OUATTARA, D. A., AND KAUFMAN, M. From structure to dynamics: Frequency tuning in the p53-Mdm2 network. I: Logical approach. *J Theor Biol* 258, 4 (2009), 561 – 577.
- [2] BOER, H. M. T., STÖTZEL, C., RÖBLITZ, S., DEUFLHARD, P., VEERKAMP, R. F., AND WOELDERS, H. A simple mathematical model of the bovine estrous cycle: follicle development and endocrine interactions. *J Theor Biol* 278 (2011), 20–31.
- [3] CHAVES, M., AND PRETO, M. Hierarchy of models: From qualitative to quantitative analysis of circadian rhythms in cyanobacteria. *Chaos: An Interdisciplinary Journal of Nonlinear Science* 23, 2 (2013), 025113.
- [4] COMET, J.-P., NOUAL, M., RICHARD, A., ARACENA, J., CALZONE, L., DEMONGEOT, J., KAUFMAN, M., NALDI, A., SNOUSSI, E., AND THIEFFRY, D. On circuit functionality in boolean networks. *Bulletin of Mathematical Biology* 75, 6 (2013), 906–919.
- [5] DAVIDICH, M., AND BORNHOLDT, S. The transition from differential equations to Boolean networks: a case study in simplifying a regulatory network model. *J Theor Biol* 255, 3 (2008), 269–277.
- [6] DIMITROVA, E. S., LICONA, P. V., MCGEE, J. J., AND LAUBENBACHER, R. C. Discretization of time series data. *J Comput Biol* 17, 6 (2010), 853–868.
- [7] FAURÉ, A., VREDEDE, B. M. I., SUCENA, E., AND CHAOUIYA, C. A discrete model of drosophila eggshell patterning reveals cell-autonomous and juxtacrine effects. *PLoS Comput Biol* 10, 3 (03 2014), e1003527.
- [8] GALLO, C. A., CECCHINI, R. L., CARBALLIDO, J. A., MICHELETTO, S., AND PONZONI, I. Discretization of gene expression data revised. *Briefings in Bioinformatics* (2015).
- [9] GLASS, L., AND HILL, C. Ordered and disordered dynamics in random networks. *EPL (Europhysics Letters)* 41, 6 (1998), 599.
- [10] GLASS, L., AND KAUFFMAN, S. A. The logical analysis of continuous, non-linear biochemical control networks. *J Theor Biol* 39, 1 (1973), 103–129.
- [11] GRIECO, L., CALZONE, L., BERNARD-PIERROT, I., RADVANYI, F., KAHN-PERLES, B., AND THIEFFRY, D. Integrative modelling of the influence of mapk network on cancer cell fate decision. *PLoS Comput Biol* 9, 10 (10 2013), e1003286.
- [12] HAIRER, E., NØRSETT, S. P., AND WANNER, G. *Solving Ordinary Differential Equations I*, 2nd ed. Springer Series in Computational Mathematics. Springer-Verlag Berlin Heidelberg, 1993.
- [13] KARLEBACH, G., AND SHAMIR, R. Modelling and analysis of gene regulatory networks. *Nat Rev Mol Cell Biol* 9 (2008), 770–780.
- [14] KLARNER, H., SIEBERT, H., AND BOCKMAYR, A. Time series dependent analysis of unparametrized Thomas networks. *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)* 9, 5 (2012), 1338–1351.
- [15] MÜSSEL, C., HOPFENSITZ, M., AND KESTLER, H. A. BoolNet – an R package for generation, reconstruction and analysis of Boolean networks. *Bioinformatics* 26, 10 (2010), 1378–1380.
- [16] NALDI, A., BERENGUIER, D., FAURÉ, A., LOPEZ, F., THIEFFRY, D., AND CHAOUIYA, C. Logical modelling of regulatory networks with GINsim 2.3. *Biosystems* 97, 2 (2009), 134.
- [17] NALDI, A., THIEFFRY, D., AND CHAOUIYA, C. Decision diagrams for the representation and analysis of logical models of genetic networks. In *Computational Methods in Systems Biology* (2007), Springer, pp. 233–247.
- [18] OUATTARA, D. A., ABOU-JAOUDE, W., AND KAUFMAN, M. From structure to dynamics: Frequency tuning in the p53-Mdm2 network. II: Differential and stochastic approaches. *J Theor Biol* 264, 4 (2010), 1177 – 1189.
- [19] REMY, E., REBOUSSOU, S., CHAOUIYA, C., ZINOVYEV, A., RADVANYI, F., AND CALZONE, L. A modeling approach to explain mutually exclusive and co-occurring genetic alterations in bladder tumorigenesis. *Cancer Res* 75, 19 (2015), 1–11.
- [20] RYLL, A., BUCHER, J., BONIN, A., BONGARD, S., GONÇALVES, E., SAEZ-RODRIGUEZ, J., NIKLAS, J., AND KLAMT, S. A model integration approach linking signalling and gene-regulatory logic with kinetic metabolic models. *Biosystems* 124 (2014), 26–38.
- [21] STÖTZEL, C., APRI, M., AND RÖBLITZ, S. A reduced ode model of the bovine estrous cycle. Tech. rep., ZIB-Report 14-33, Konrad-Zuse-Zentrum für Informationstechnik Berlin, 2014.
- [22] THOMAS, R. Regulatory networks seen as asynchronous automata: a logical description. *J Theor Biol* 153, 1 (1991), 1–23.
- [23] THOMAS, R., AND D’ARI, R. *Biological Feedback*. CRC Press, 1990.
- [24] TYSON, J. J., CHEN, K. C., AND NOVAK, B. Sniffers, buzzers, toggles and blinkers: dynamics of regulatory and signaling pathways in the cell. *Curr Opin Cell Biol* (2003), 221–231.
- [25] VAN RIEL, N. A. Dynamic modelling and analysis of biochemical networks: mechanism-based models and model-based experiments. *Brief Bioinform* 7, 4 (2006), 364–374.
- [26] VELIZ-CUBA, A., ARTHUR, J., HOCHSTETLER, L., KLOMPS, V., AND KORPI, E. On the relationship of steady states of continuous and discrete models arising from biology. *Bull Math Biol* 74, 12 (2012), 2779–2792.
- [27] WANG, R.-S., SAADATPOUR, A., AND ALBERT, R. Boolean modeling in systems biology: an overview of methodology and applications. *Phys Biol* 9, 5 (2012), 055001.