From Binary to Multivalued to Continuous Models: The lac Operon as a Case Study

Raimo Franke*, Fabian J. Theis and Steffen Klamt

1Department of Chemical Biology, Helmholtz Centre for Infection Research, 38124 Braunschweig, Germany

2Institute for Bioinformatics and Systems Biology, Helmholtz Centre Munich - German Research Center for Environmental Health, 85764 Neuherberg, Germany

3Max Planck Institute for Dynamics of Complex Technical Systems, 39106 Magdeburg, Germany

Summary

Using the lac operon as a paradigmatic example for a gene regulatory system in prokaryotes, we demonstrate how qualitative knowledge can be initially captured using simple discrete (Boolean) models and then stepwise refined to multivalued logical models and finally to continuous (ODE) models. At all stages, signal transduction and transcriptional regulation is integrated in the model description. We first show the potential benefit of a discrete binary approach and discuss then problems and limitations due to indeterminacy arising in cyclic networks. These limitations can be partially circumvented by using multilevel logic as generalization of the Boolean framework enabling one to formulate a more realistic model of the lac operon. Ultimately a dynamic description is needed to fully appreciate the potential dynamic behavior that can be induced by regulatory feedback loops. As a very promising method we show how the use of multivariate polynomial interpolation allows transformation of the logical network into a system of ordinary differential equations (ODEs), which then enables the analysis of key features of the dynamic behavior.

1 Introduction

Biological networks can be subdivided into metabolic, signal transduction and regulatory networks. Here the term "regulatory network" is used for transcriptional and translational regulation by convention, although regulatory features are associated with all cellular processes [1]. With rare exceptions [2,3] most of the previous computational analyses of cellular networks have focused on one of these network layers separately, without considering the interplay that exists between them. This conceptual division can be useful for the mathematical treatment, however in the cell all of the components work in an integrated fashion that promotes fitness [1]. A simple example for the interplay of the different network layers would be a receptor that is triggered by an extracellular stimulus and induces a signaling cascade, which results in the activation of a transcription factor. As a consequence, expression of a target gene is induced, resulting in the production of a protein which inhibits the signaling cascade at a certain stage and thereby prevents activation of the transcription factor in a negative feedback loop. This example shows the interconnectedness of signal transduction and transcriptional regulation. It is desirable to develop modeling frameworks that enable an integrated treatment of all layers that make up cellular life.

* To whom correspondence should be addressed. Email: raimo.franke@helmholtz-hzi.de

doi:10.2390/biecoll-jib-2010-151
Mechanism-based modeling approaches based on differential equations enable a dynamic analysis of cellular networks and have a high predictive power. However, they are limited to rather small systems because they rely on detailed knowledge of kinetic laws and parameters of the underlying biochemical reactions which is often not available. For modeling large-scale networks, qualitative modeling approaches and network analysis techniques are usually better suited as they seek to elucidate functional features from the often well-characterized network topologies alone (e.g., wiring diagram of regulatory networks or reaction stoichiometries in metabolic networks). For example, in signaling and regulatory networks one is typically interested in (i) detection of network-wide functional interdependencies between network elements, (ii) identification of feedback loops, (iii) identification of interventions that induce a specific response and (iv) qualitative predictions on the effect of perturbations. We introduced a logical modeling framework (Boolean networks represented as logical interaction hypergraphs, LIHs [4,5]) that is ideal for the reconstruction and qualitative analysis of cellular networks with signal or information flows. Examples that have been studied include diverse signaling networks such as T cell signaling [6], cMet signal transduction [7], NF-κB signal transduction [8], and EGF signaling [9]. Signaling networks are structured into input, intermediate and output layer, which facilitate crosstalk and integrated decision making and are often treated as acyclic network as a first approximation. Gene regulatory networks on the other hand are strongly determined by their feedback loops (cyclic networks). Signaling and regulatory networks are intertwined in the cell and accounting for the coupling of signaling and gene regulation is highly desirable towards a more complete model of the cell.

As a case study we chose the well-described lac operon, the paradigmatic example for a gene regulatory system in prokaryotes [10-12]. The involvement of species from metabolic, signaling and gene regulatory network layers and also the existence of feedback loops in the regulation of the lac operon make it a very attractive system to evaluate different modeling approaches.

In this study we initially start with a Boolean model of the lac operon and evaluate its potential to capture essential features of the mechanisms involved in the regulation of the lac operon. The problems and limitations that arise following a binary treatment will be shown and ways how to refine this representation to multivalued logic and then to qualitative ODE’s will be presented.

2 Methods

2.1 Boolean networks represented as logical interaction hypergraphs

For a detailed introduction into the formalism of logical interaction hypergraphs (LIHs) and its implementation in CellNetAnalyzer, we refer the reader to our previous publications [4,5,13]. In brief, this Boolean modeling framework was tailored for studying the qualitative input-output response of signaling networks. As in all Boolean networks, nodes in the network represent biomolecular species (e.g. kinases, adaptor molecules, transcription factors, or genes) each having an associated logical state (in the binary case only “on” (1) or “off” (0)) expressing whether the species is active (or present) or not. Signaling events are encoded as Boolean operations on the network nodes. In contrast to other works focusing on discrete dynamics in Boolean networks [14,15], the approach of LIHs was mainly used to study the input/output behavior of signaling networks by analyzing the qualitative (logical) steady state that results from a given external stimulation or perturbation.

LIHs make only use of the Boolean operators AND (·), OR (+), and NOT (!), which are sufficient to represent any logical relationship. The so-called sum-of-product representation,
where AND terms are connected via OR operators, makes it possible to represent a given Boolean network as a (logical interaction) hypergraph (LIH) [5,13]. To exemplify this for a node in the lac operon network: allolactose is only produced (gets “on”, i.e. value 1) if lactose is present in the cell AND the β-galactosidase LacY enzyme is expressed, i.e. lactose_int (for internal lactose) AND LacY must be “on” to produce allolactose (see figure 1). Hence, for the example described above we would write

\[
lactose\text{\_int} \text{ AND } \text{LacY} \rightarrow \text{allolactose}
\]

or, shorter,

\[
lactose\text{\_int} \cdot \text{LacY} \rightarrow \text{allolactose}
\]

In a graphical representation of the network such an AND connection is displayed as a hyperarc (see figure 1a) indicating that all start nodes of the hyperarc (lactose_int, LacY) must be in the "on" state in order to activate the end node of the hyperarc (note that hyperarcs may have several start or end nodes). NOT operators for variables entering a hyperarc are allowed and are graphically indicated, e.g., by a red color or/and bars. For example, interaction 5 (see table 2 and its (hyper-)graphical representation in figure 1) reads

\[
!\text{PTS-EIIA} \cdot \text{lactose\_ex} \cdot \text{LacY} \rightarrow \text{lactose\_int}
\]

indicating that the unphosphorylated form of EIIA\text{glc} must be off (i.e. it is has to be in the phosphorylated form EIIA\text{glc}\_P) AND the permease LacY AND the substrate lactose must be available in order to get internal lactose. Finally, OR connections can be accounted for in the hypergraphical representation by allowing a node to be independently activated by several incoming hyperarcs (i.e. by several independent AND connections).

By this hypergraphical representation, we can study a number of useful properties of the logical network or its underlying interaction graph [5-9]. One particular application of which we will make use herein is the prediction of the input-output behavior that follows from a given input stimulus (possibly combined with internal perturbations such as knockout of certain nodes) by computing the resulting logical steady state (LSS). A detailed description of the algorithm for computing the LSS was given in [13]; here we will apply it to the lac operon model by computing the binary response of the involved species for given substrate mixtures (glucose or/and lactose).

The logical models studied herein were implemented and analyzed with the software tool CellNetAnalyzer [4].

### 2.2 Multivalued logic

As already proposed and applied by others (see e.g. [15]), the discretization of a node’s activation level in more than two (binary) levels is possible. This mimics the fact, that in reality multiple relevant threshold values for a species may exist.

It is straightforward to extend binary (Boolean) logic to multivalued logic. Embedded in the LIH formalism, CellNetAnalyzer also supports multivalued logic: in addition to the binary on/off-case, other levels for the species can be defined. For example we can formulate a logical function like this:

\[
!A \cdot 2 \rightarrow 3C.
\]

This equation means that "C reaches level 3 if A is inactive (level 0) AND B is at least at level 2". The word “at least” indicates that CellNetAnalyzer assumes monotone relationships (in principle, non-monotone logical functions can also be defined but are not considered herein).
2.3 Multivariate polynomial interpolation: ODEfy

A more traditional model of transcriptional regulation and signaling based on chemical reaction kinetics involves the continuous description of concentration changes of the various species. The most common approach hereby is the use of a system of coupled ordinary differential equations (ODEs), which essentially ignore spatial dimensions as well as time delays and stochasticity for simplicity. In contrast to the discrete systems based on multivalued logic, the use of ODEs allows the ready inclusion of gradual concentration changes as well as fully time-resolved dynamics. This more fine-grained approach of course comes with the cost of many parameters such as reaction and degradation rates. While this allows a more detailed description of the observations and predictions, the rates need to be approximated by literature values or learned by fitting the model to data. Here, we will take the latter, unbiased approach.

We have previously described how to extend a Boolean logic model to an ODE model [16], which we denote as ODEfy in the following. Given a n-variate Boolean function B, which is defined on the vertices of an n-dimensional hypercube, we defined a continuous extension C of B on the full hypercube by multi-linear interpolation. In order to accommodate different levels of activity for each input in B, we then concatenate C with a component-wise sigmoidal nonlinearity using the Hill function \( f_{k,n}(x) = \frac{x^n}{(x^n + k^n)} \). Here n is the degree of nonlinearity (which in the following we fix to \( n=3 \) for simplicity) and k the switching threshold. For \( n \rightarrow \infty \) this approaches a discrete switch at level \( x=k \), which corresponds to the discrete function. This interpolation technique can be derived from a thermodynamical model for gene regulation [17]. However, other non-linear functions (e.g. logistic functions) could, in principle, be used.

Using the example of ‘lactose_int AND LacY \( \rightarrow \) allolactose’ from above, this is linearly interpolated by the function \( C(\text{lactose_int}, \text{LacY}) = \text{lactose_int} \times \text{LacY} \), because \( C(x,y)=1 \) only if both \( x=1 \) and \( y=1 \). The resulting dynamics is given by

\[
\frac{d}{dt}(\text{allolactose}(t)) = \frac{1}{\tau_{\text{allolactose}}} (f_{k_1,n_1}(\text{lactose_int}) \cdot f_{k_2,n_2}(\text{LacY}) - 1)
\]

where \( \tau_{\text{allolactose}} \) denotes the life-time of the species allolactose, and \( k1, k2, n1, n2 \) the parameters of the Hill nonlinearity. We have shown that the Boolean attractors are conserved under this transformation given a sufficiently high degree of nonlinearity [16]. Applications to network inference from spatial patterns in neurodevelopment illustrate that this technique is capable of giving insight into biological dynamics based on initial qualitative information [18].

We provide the toolbox ODEfy as plugin to CellNetAnalyzer [19], which can be easily used to generate the ODE model in various formats. Here, we opt for export to SBToolbox2 for Matlab, where we can compile the resulting ODE in binary format for fast and efficient evaluation in the parameter fitting part described below.

Other methods for transforming Boolean into continuous ODE models have been proposed and a comparison of ODEfy with alternative methods can be found in [16]. In short, we can group these methods into piecewise-linear interpolations, which are still discrete-output generalizations of step-functions, fuzzy-logic continuations, which depend on the choice of degree-of-memberships, and ad-hoc interpolations, which do not possess specific theoretical properties, but reflect various biomathematical aspects. We view the ODEfy approach of
multi-linear interpolation first pioneered by [20] in possible combination with Hill functions as the most simple nonlinear interpolation technique (in terms of complexity measured by degree of the used polynomial) that still allows a mechanistic derivation as proposed in [22].

3 Results

3.1 Conceptual model of the lac operon

In the following section a detailed description of a conceptual model of the lac operon is given, which comprises all the components that are translated into the Boolean model.

The lac operon in E. coli consists of three different structural genes that are transcribed as a single mRNA [11] in response to a certain glucose/lactose ratio. This polycistronic lacZYA mRNA is translated into three proteins, which are required to import and digest the disaccharide lactose. The lac operon is the prototype of a single promoter being under the control by two different transcription regulators, the lac repressor LacI and the activator protein CAP (catabolite activator protein).

The basic idea of Jacob and Monod was that the structural genes of the lac operon are regulated by a repressor, which represses transcription, until it interacts with a chemical "inducer" [11]. The lac operon can be described as an inducible regulatory system, although the term "inducer" can be misleading, because the environmental stimulus (lactose) halts the repression ("repression of the repressor"), which has in essence the effect of inducing transcription.

Soon after Jacob and Monod published their model [12] the lactose repressor protein LacI that controls the structural genes responsible for lactose metabolism was experimentally identified [22,23]. The regulatory gene lacI encodes the Lac repressor LacI, which is capable of inhibiting transcription of the structural genes of the lac operon by binding with high affinity to the lac operon at a specific operator DNA sequence (lacO1) near the lac promoter [11,23,24]. In addition to the primary operator site O1, two auxiliary pseudo-operators were identified (O2 and O3) [11]. The binding of the LacI repressor next to the lac promotor has the effect that RNA polymerase binding is compromised [25,26]. As a consequence the lac mRNA level is strongly reduced but not to zero. The lac mRNA is thus not completely eliminated by LacI binding, but reduced to a basal level (this aspect will later become important for building the multivalued model). LacI is a tetrameric protein, which can in principle bind two operator sites simultaneously [11], but this aspect is not considered in our model. The regulated structural genes are lacZ, lacY and lacA, which encode enzymes, that are all involved in lactose metabolism [11]. The lacZ gene encodes the β-galactosidase LacZ, a hydrolase enzyme that catalyzes the hydrolysis of the disaccharide lactose into the monosaccharides glucose and galactose, which is the first step in lactose metabolism [27]. The lacY gene encodes β-galactoside permease (LacY), a membrane-bound transport protein that enables the entry of lactose into the cell [28]. The third gene lacA, which is also under the control of the lac promotor, encodes a thiogalactoside transacetylase, which transfers an acetyl group from coenzymeA (CoA) to the hydroxyl group of galactosides [11]. The transacetylase is not essential for lactose metabolism [29] and therefore not included in the model.

The lac repressor LacI inhibits transcription of the lac structural genes only when it is activated. LacI is inactivated, when it is bound by allolactose, a by-product of lactose metabolism (a small fraction of the cleavage products of lactose –glucose and galactose – can recondense to form allolactose). In this context the basal expression levels of lac permease and β-galactosidase play an essential role. Enzymatic activity of the permease enables transport of lactose from the medium into the cell, where β-galactosidase converts a small
fraction of lactose into 1,6-allolactose [11,30]. By binding to LacI with high affinity, 1,6-
allolactose lowers LacI’s affinity to the operator and thereby induces transcription of the lac
structural genes. In the absence of lactose the Lac repressor switches off the operon – a
mechanism for the cell not to waste energy for the production of enzymes for the lactose
metabolism, when they are not needed.

In the presence of glucose in the medium, E. coli uses it as the sole carbon source to produce
energy via respiration, even when both lactose and glucose are available [31,32]. The
presence of lactose in the medium alone is thus not sufficient for full induction of the lac
operon. Although the repressor LacI does not occupy the operator site, the operon is
transcribed infrequently and remains largely inactive, as long as glucose is available [11]. The
uptake of glucose into the cell by the phosphoenol pyruvate-dependent phosphotransferase
systems (PTS) decreases the level of phosphorylation of one of its components, the enzyme
EIIA\textsuperscript{Glc} [33]. The dephosphorylated EIIA\textsuperscript{Glc} prevents the uptake of lactose by binding to the
lac permease LacY [34,35]. As a consequence when both glucose and lactose are present in
the medium, E. coli cells preferentially utilize glucose and the use of lactose is prevented until
the glucose is depleted [33]. The result of reducing the transport activity of LacY by
unphosphorylated EIIA\textsuperscript{Glc} is called inducer exclusion, because lactose cannot enter the cell,
and as a consequence the inducer of the lac operon expression allolactose is not produced.

An additional control mechanism by the catabolite activator protein (CAP), regulated by cyclic
AMP (cAMP), contributes as well to the selective utilization of metabolites. In bacteria,
cAMP is low when glucose is used as carbon source. This occurs through inhibition of the
cAMP-producing enzyme, adenyl cyclase, as a side-effect of glucose transport into the cell
[36]. Glucose is transported into the E. coli cell by the PTS [37]. A phosphate group is
transferred from phosphoenol pyruvate through a series of intermediary proteins to the
EIIA\textsuperscript{Glc} complex, which finally transfers a phosphate group to glucose, that enters the cell as
glucose-6-phosphate [37]. The EIIA\textsuperscript{Glc} subunit of the EII complex is also involved in the
activation of the adenylate cyclase (AC). With glucose in the medium, the EIIA\textsuperscript{Glc}-phosphate
will be used to supply phosphate to the glucose and the amount of EIIA\textsuperscript{Glc}-phosphate will thus
be reduced. As only the phosphorylated form of EIIA\textsuperscript{Glc} stimulates the adenylate cyclase
activity, the cAMP level will fall [37].

When glucose, the preferred carbon source of E. coli, is no longer available in the medium,
the intracellular concentration of cAMP rises. The change of the cAMP level signals to the
bacterium that glucose is no longer available and that it has to switch to lactose metabolism.
This is achieved by binding of cAMP to CAP, which from a cAMP-CAP complex. In turn
binding of this complex to a DNA sequence in the promotor region just upstream from the lac
promoter enhances affinity of the RNA polymerase for the promotor and thereby initiates full
transcription of the lac structural genes [11,31,32]. Without the binding of the activator CAP,
the lac promotor is only marginally able to bind and position the RNA polymerase resulting in
a low level of transcription (50-fold reduced) [31,32].

When glucose is present and cAMP concentration is low in the cell, the lac operon is switched
off, because the activator protein CAP can only bind to DNA if it is bound to cAMP. Glucose
uptake also interferes with LacY activity, leading to inducer exclusion as discussed above
[37]. Inducer exclusion and the drop of the cAMP level in response to glucose uptake together
mediate catabolite repression - the ability of glucose to inhibit lac expression [37,38].

The logical model should be able to explain how repression and activation control
mechanisms account for the selective utilization of the available metabolites.
3.2 A Boolean model of the *lac* operon

The Boolean network model of the *lac* operon was constructed based on the biological knowledge described in the previous section. Figure 1a shows the resulting model in hypergraphical notation. The figure was drawn by using the Software CellDesigner following SBGN (Systems Biology Graphical Notation) conventions as far as possible. Figure 1b shows again the rules to depict Boolean gates in a hypergraphical representation (as LIH). Table 1 and 2 lists the network species and interactions of the logical model.

![Figure 1a: Boolean model of the lac operon](image)

![Figure 1b: Legend of graphical representations of logic operators](image)

Figure 1: (a) Boolean model of the lac operon, hyperarcs are numbered according to the annotations in table 2. (b) Legend of graphical representations of logic operators.
<table>
<thead>
<tr>
<th>Symbol</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>AC</td>
<td>adenylate cyclase</td>
</tr>
<tr>
<td>allolactose</td>
<td>allolactose</td>
</tr>
<tr>
<td>cAMP</td>
<td>cyclic adenosine monophosphate</td>
</tr>
<tr>
<td>CAP</td>
<td>catabolite activator protein</td>
</tr>
<tr>
<td>glucose_ex</td>
<td>external glucose in the growth medium</td>
</tr>
<tr>
<td>LacI-bound</td>
<td>LacI repressor bound to the lac promotor</td>
</tr>
<tr>
<td>lactose_ex</td>
<td>external lactose in the growth medium</td>
</tr>
<tr>
<td>lactose_int</td>
<td>lactose inside of the cell</td>
</tr>
<tr>
<td>LacY</td>
<td>LacY permease</td>
</tr>
<tr>
<td>lacZYA</td>
<td>the structural genes of the lac operon: lacZ, lacY and lacA</td>
</tr>
<tr>
<td>LacZ</td>
<td>β-galactosidase LacZ</td>
</tr>
<tr>
<td>lacZYA_mRNA</td>
<td>polycistronic lacZYA mRNA</td>
</tr>
<tr>
<td>PTS-EIIA</td>
<td>unphosphorylated EIIA&lt;sub&gt;↑&lt;/sub&gt; subunit of the phosphotransferase system</td>
</tr>
</tbody>
</table>

Table 1: Species of the Boolean lac operon model.

<table>
<thead>
<tr>
<th>Hyperarc</th>
<th>Annotation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>glucose&lt;sub&gt;↑&lt;/sub&gt; → PTS-EIIA</td>
</tr>
<tr>
<td></td>
<td>When glucose is imported, phosphate transfer from the PTS subunit EIIA&lt;sub&gt;↑&lt;/sub&gt; to glucose is induced. Active PTS-EIIA in the model corresponds to the unphosphorylated form of EIIA.</td>
</tr>
<tr>
<td>2</td>
<td>!PTS-EIIA → AC</td>
</tr>
<tr>
<td></td>
<td>Phosphorylated EIIA&lt;sub&gt;↑&lt;/sub&gt; activates AC, thus PTS-EIIA must be off.</td>
</tr>
<tr>
<td>3</td>
<td>AC → cAMP</td>
</tr>
<tr>
<td></td>
<td>AC produces cAMP through conversion of ATP.</td>
</tr>
<tr>
<td>4</td>
<td>cAMP → CAP</td>
</tr>
<tr>
<td></td>
<td>cAMP binds to the receptor molecule CAP to form the activator CAP complex.</td>
</tr>
<tr>
<td>5</td>
<td>!PTS-EIIA · lactose&lt;sub&gt;↑&lt;/sub&gt; · LacY → lactose&lt;sub&gt;↑&lt;/sub&gt;</td>
</tr>
<tr>
<td></td>
<td>Entry of lactose from the medium into the cell is enabled by the LacY permease; unphosphorylated EIIA blocks the import of lactose.</td>
</tr>
<tr>
<td>6</td>
<td>lactose&lt;sub&gt;↑&lt;/sub&gt; · LacZ → allolactose</td>
</tr>
<tr>
<td></td>
<td>If β-galactosidase LacZ is expressed and lactose is present in the cell, allolactose is produced as a byproduct of lactose metabolism.</td>
</tr>
<tr>
<td>7</td>
<td>!allolactose → LacI-bound</td>
</tr>
<tr>
<td></td>
<td>Binding of allolactose to LacI inactivates the repressor.</td>
</tr>
<tr>
<td>8</td>
<td>!LacI-bound · CAP · lacZYA → lacZYA&lt;sub&gt;↑&lt;/sub&gt; mRNA</td>
</tr>
<tr>
<td></td>
<td>Only when LacI is inactive and not bound to lacO and the lacZYA structural genes are present and CAP binds, lacZYA&lt;sub&gt;↑&lt;/sub&gt; mRNA is produced.</td>
</tr>
<tr>
<td>9</td>
<td>lacZYA mRNA → LacY</td>
</tr>
<tr>
<td></td>
<td>Translation of lacZYA mRNA produces the LacY permease.</td>
</tr>
<tr>
<td>10</td>
<td>lacZYA mRNA → LacZ</td>
</tr>
<tr>
<td></td>
<td>Translation of lacZYA mRNA produces the LacZ β-galactosidase.</td>
</tr>
<tr>
<td>11</td>
<td>lacZYA mRNA → LacA (dashed arcs are not incorporated in the model)</td>
</tr>
<tr>
<td></td>
<td>Translation of lacZYA mRNA produces the LacA thiogalactoside transacetylase (not included in the model; only for illustration).</td>
</tr>
<tr>
<td>12,13,14</td>
<td>dashed arcs are not incorporated in the model.</td>
</tr>
<tr>
<td></td>
<td>lactose&lt;sub&gt;↑&lt;/sub&gt;, glucose-phosphate and LacZ enter the lactose and glucose metabolism that are not considered in the model.</td>
</tr>
</tbody>
</table>

Table 2: Hyperarcs of the Boolean lac operon model. Exclamation mark within the equations denote a logical NOT, dots indicate AND operations.

As validation of the network model, well-known experimental scenarios should be reproduced by the model. As described under 3.1, the presence of glucose and lactose in the growth
medium control the state of the lac operon. We approximate the behaviours by on and off states of the network nodes. Four ($2^2$) combinations of glucose/lactose in the medium are possible when considering the binary case.

These four scenarios were simulated with the Boolean model by setting the input cues glucose and lactose appropriately. The resulting logical steady states were computed with CellNetAnalyzer. The steady state values for the most important nodes are shown in table 3 together with the experimentally observed state of the lac operon (measured by lacY expression).

<table>
<thead>
<tr>
<th>binary model</th>
<th>+Glc / +Lac</th>
<th>-Glc / -Lac</th>
<th>+Glc / -Lac</th>
<th>-Glc / +Lac</th>
</tr>
</thead>
<tbody>
<tr>
<td>cAMP</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>allolactose</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>#</td>
</tr>
<tr>
<td>LacI-bound</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>#</td>
</tr>
<tr>
<td>lacZYA mRNA</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>#</td>
</tr>
<tr>
<td>LacZ</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>#</td>
</tr>
<tr>
<td>LacY</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>#</td>
</tr>
<tr>
<td>lactose_int</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>#</td>
</tr>
</tbody>
</table>

Table 3: Logical steady states for the key nodes in the binary model and the observed lac operon state for four different scenarios with glucose/lactose as substrates.

For the first three scenarios the model simulations are in agreement with the expected outcome from literature. The status of the lac operon can be deduced from the steady state value for lacZYA mRNA, i.e. whether mRNA is produced or not. In all these three cases, the model predicts correctly that the lac operon is switched off.

For the last scenario, where only lactose is used as a stimulus (glucose is off), a unique logical response for this set of input stimuli cannot be resolved (see screenshot of CellNetAnalyzer in figure 2). A logical steady state (the final response of the system) cannot be calculated for all network nodes, because in the simulation the signal cannot be further propagated from lactose_ex to lactose_int without the LacY permease being in on-state. The limitations of a binary view of biology are made obvious: when lactose is sensed in the environment, the system should switch on the lac operon, which is achieved by binding of allolactose to the repressor. The only problem is that the genes for the permease and the β-galactosidase, which are needed (i) to enable lactose to enter the cell and (ii) to produce allolactose are both structural genes within the lac operon whose activity, as mentioned above, depends in turn on the presence of allolactose (whose initial state is not given). To resolve this cyclic causality (induced by a positive feedback loop which is also the source of bistable behavior discussed in a later section) we need a basal expression level of the lac operon to produce some mRNA which will make the system work for the Glc~/Lac+ scenario.
Multivalued logical model

Using the approach of multivalued logic described under 2.2 we can modify the model to circumvent problems that arise when the state of a network species is discretized to only two levels (on/off). The real biochemical behaviour can be approximated much better using multivalued logic. We therefore made the following alterations to our model: we defined three (instead of two) levels for the state of the lac operon. These three levels (0,1,2) correspond to different expression rates and are implemented as differential production of lacZYA mRNA:

Level 0: The lac genes are not transcribed at all, e.g. due to a knock-out of lacZYA.

Level 1: Basal expression level of the lac operon, when LacI is bound to the lacO operator and the CAP complex is not present. From the on/off-binding kinetics of the lac repressor we know that the repressor is tightly bound, but still occasionally not bound, which enables a low basal expression level of the lac operon. When lactose then gets into the cell, the equilibrium is shifted towards a higher expression rate and higher level of permease will enable more lactose to enter the cell.

Level 2: Full expression of the lac structural genes, LacI does not bind to lacO and the cAMP-CAP complex binds next to lacP and by enhancing affinity of the polymerase for lacP induces full transcription.

To translate this into logical equations we replace interaction 8 by two novel hyperarcs that represent the two levels for the production of lacZYA mRNA:
Level 1: \( \text{lacZYA} \rightarrow \text{lacZYA}_\text{mRNA} \)

Level 2: \( \text{lacZYA} \cdot \text{CAP} \cdot \text{!LacI} \rightarrow 2 \text{lacZYA}_\text{mRNA} \)

Moreover we define that lactose can enter the cell in two cases: i) the \( \text{lac} \) operon is at basal expression level (level 1) and thus the permease LacY is also produced at level 1. In this case the PTS has to be inactive (meaning that EIIA\text{Glc} has to be phosphorylated), to enable lactose to enter the cell. ii) The \( \text{lac} \) operon is fully switched on (level 2), thus the permease LacY is also expressed at level 2. In this case it is sufficient for LacY to be at level 2 to allow lactose to enter the cell. This is translated by replacing the hyperarc 5 in the Boolean model (table 2) with the following two new hyperarcs:

i) lactose\_ex \cdot \text{LacY} \cdot \text{!PTS-EIIA} \rightarrow \text{lactose\_int} 

ii) lactose\_ex \cdot 2 \text{LacY} \rightarrow \text{lactose\_int} 

With these adjustments we simulated again the case where lactose is present and glucose is absent (figure 3).

Figure 3: The multivalued logical model for the \( \text{lac} \) operon (screenshot from CellNetAnalyzer). The text boxes show the resulting logical steady state for the scenario -glucose/+lactose switching the \( \text{lac} \) operon fully on (level 2). For the colour-code of the boxes see figure 3. The numbers in the grey circles denote the level of multivalued nodes in the respective hyperarc.

Figure 3 shows that now a unique steady state for the system can be calculated. In agreement with the experimental values, the \( \text{lac} \) operon is fully switched on (level 2 indicates full transcription). With the introduction of a basal expression level (level 1), which is already reached when the \( \text{lacZYA} \) structural genes are present, lactose\_ex can enter the cell and the \( \text{lac} \)
The operon is fully switched on (level 2). Multivalued logic thus enables a more realistic
description of the system compared to the Boolean model.

The remaining three scenarios were also simulated with the multivalued logical model. The
steady state values for the key nodes are shown in table 4 indicating agreement with
biological knowledge (similar as in the binary model): the model predicts that the structural
lac genes are transcribed with low levels if either glucose or no lactose is present. The only
difference to the Boolean model is, that lacZYA mRNA, LacZ and LacY are at level 1
corresponding to the implemented basal expression level.

<table>
<thead>
<tr>
<th>model multival</th>
<th>+Glc / +Lac</th>
<th>-Glc / -Lac</th>
<th>+Glc / -Lac</th>
<th>-Glc / +Lac</th>
</tr>
</thead>
<tbody>
<tr>
<td>cAMP</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>allolactose</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>LacI</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>lacZYA mRNA</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>LacZ</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>LacY</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>lactose_int</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>observed</td>
<td>off</td>
<td>off</td>
<td>off</td>
<td>on</td>
</tr>
</tbody>
</table>

Table 4: Steady state values for the key nodes in the multivalued logic model for the 4 scenarios.

In summary with the introduction of multi-level logic we get a more differentiated picture of
the activity state of the lac operon and now all four scenarios can be correctly reproduced by
the model.

### 3.4 A qualitative ODE model for the lac operon derived with ODEfy

As basis for the ODE-model, we use the multivalued discrete model from section 3.3. In
principle there are two different approaches for interpolating multivalued discrete models: we
can either rewrite the update rules in terms of Boolean species by introducing dummy species
and interpolate those, or do a multivalued interpolation. We chose the second approach
because it should be sufficient for monotone models considered herein. To build the
continuous interpolation of a multivalued function, we therefore interpolate it on the maximal
and minimal values, and then select appropriate parameters in the nonlinearities to be close to
the interpolated discrete values.

The resulting ODE-model consists of a 13-dimensional differential equation (one for each
node) with in total 38 parameters. To derive meaningful dynamics, we need to select
parameters by fitting the model to a set of observed data. Here, we chose a time-series of
LacY-activity after growing on lactose [40], see figure 4a. We fix the nonlinearity parameter
n to 3 to reduce the dimensionality of the search space, and end up with 24 free parameters.

In a first run, we estimate the parameters using normalized distance to the LacY data as single
criterion. Fitting was performed using a global heuristic minimization technique based on
differential evolution [41]. In figure 4a, we see that the interpolated LacY concentration fits
the data very well. The dynamics of the other species concentrations is shown in figure 4b.
Figure 4: ODEfy simulations of the lac operon when switching from growth on glucose to lactose. a) LacY data (taken from [40]) and resulting ODEfy-interpolation after parameter estimation. b) Resulting dynamics of all state variables.

However, as was to be expected we find many quite different parameter sets satisfying small distances to the data. Importantly, some parameter sets may lead to bistability whereas others do not. The potential of the lactose utilization network to induce bistability could be confirmed in experiments with the non-metabolizable lactose derivate thio-methylgalactoside (TMG) [38,39]. To further reduce the parameter indeterminacies, in a second fitting run, we now search for models that fit the LacY data well but at the same time also produce different steady states when first grown on either lactose or glucose and then switched to equally low lactose levels (without presence of glucose). This bistable behavior is achieved by maximizing the distance between the two steady states. The resulting updated dynamics interpolate the data from figure 4a equally well. But at the same time they produce bistable behavior, see figure 5.

Figure 5: Bistable behavior of the fitted system: Growth on lactose (a) or glucose (b), then switch to low lactose (0.2) at time t=200min. The resulting steady states differ (indicating bistability) and depend on the previous growth conditions. Differences in the steady states are particularly significant for LacI-bound, LacY, LacZ and lacZYA_mRNA. The changes of the internal concentrations of allolactose and lactose are less pronounced (for the chosen parameter set) but still sufficient to establish two steady states.

Again, even when demanding bistability, we cannot hope to estimate a single ‘best model’ and therefore aim at determining a whole ensemble of models satisfying the fitting criterion.
This is done using the Monte Carlo extension of the differential evolution algorithm put forward in [41]. The distribution of the resulting cost function values is shown in figure 6. The resulting parameters for distances to LacY data lower than 0.5 are shown in figure 7. We observe that many switching thresholds are not localized. However, some are such as parameter 10 (Km value of LacI in the regulation of lacZYA expression) which needs to be sufficiently high – presumably a requirement for inducing bistability. The lifetime distributions shown in figure 7b are more localized but tend to be at the boundary of the optimization interval, implying that longer lifetimes in particular for species 4 (CAP), 5 (lacZYA_mRNA), 7 (LacY) and 8 (AC) are more optimal in the fitting procedure.

![Distribution of the resulting cost function from the Monte Carlo extension of the differential evolution algorithm [41]](image)

**Figure 6: Distribution of the resulting cost function from the Monte Carlo extension of the differential evolution algorithm [41]**

### 4 Discussion

For the large scale analysis of signalling networks Boolean networks represented as logical interaction hypergraphs (LIHs) have proven to be a useful framework. Here we take the well described lac operon as an example for a gene regulatory network and integrate signal transduction and regulation of transcription in a Boolean model. Boolean models have their merits in the straight forward way to reconstruct them from biological knowledge and they can be used to identify important features of signaling and regulatory networks. However, the example of the lac operon also showed the limitations of a Boolean approach in cyclic networks because a unique network behavior can then often not be computed from a Boolean description. Accordingly, in one of the test scenarios a logical steady state could not be deduced for all nodes.

With the introduction of multivalued logic, the problems and limitations that arise in the Boolean description can be partially circumvented by introducing multiple levels for the activity state of the lac operon. Multivalued logic is thus a promising extension for logical networks, enabling a more realistic and sophisticated description of biological networks. Multivalued logic can be used as an intermediate modeling step between a Boolean and a dynamical ODE-based model and is not limited to small systems.
Figure 7: Distributions of selected fitted parameters. The boxes have lines at the lower quartile, median, and upper quartile values, and whiskers span 1.5 times the interquartile ranges. Outliers, indicated by red plus signs, are data with values beyond the ends of the whiskers.

a) switching thresholds (1=lactose_int_k_lactose_ex; 2=lactose_int_k_LacY; 3=lactose_int_k_PTS; 4=allolactose_k_lactose_int; 5=allolactose_k_LacZ; 6=cAMP_k_AC; 7=CAP_k_cAMP; 8=lacZYA_mRNA_k_CAP; 9=lacZYA_mRNA_k_lacZYA; 10=lacZYA_mRNA_k_LacI_bound; 11=lacZ_k_lacZYA_mRNA; 12=LacY_k_lacZYA_mRNA; 13=AC_k_PTS; 14=PTS_k_glucose_ex; 15=LacI_bound_k_allolactose).

b) lifetimes (1=lactose_int_tau; 2=allolactose_tau; 3=cAMP_tau; 4=CAP_tau; 5=lacZYA_mRNA_tau; 6=LacZ_tau; 7=LacY_tau; 8=AC_tau; 9=PTS_tau; 10=LacI_bound_tau). See text for discussion.

By finally interpolating the discrete model to a continuous Hill-type ODE model using ODEfy, we can map the developed qualitative models onto a quantitative model, ready to be compared with transient and quantitative concentration data. This opens a promising modeling pipeline: Starting from the commonly available qualitative and topological information in the biological literature, we derive first discrete, then refined discrete and eventually continuous models in order to finally predict concentration time series. This however comes at the cost of increased complexity and in particular the issue of model fitting in the presence of many model indeterminacies. We have proposed to deal with the latter issue using Bayesian model fitting, thereby describing a whole ensemble of fitted models. We can then assign a plausibility value to any predictions by evaluating it on each of the ensemble’s models. This approach was successful in predicting a novel interaction lying at the intersection of many discrete models in [41].

The case of the lac operon was used herein as an example network where much (though by far not all) biological details are known. We emphasize that even the ODE model does still not correctly reproduce all aspects of the behavior of the lac operon. For example, it is known that the cAMP level increases only transiently when switching from glucose to lactose utilization and it decreases when the shift to lactose metabolism has been achieved [37]. However, that this behavior is not reflected in the ODE model is not a limitation of the modeling approach: we probably have to include additional regulatory links into the model. For example, the expression of the adenylate cyclase is transcriptionally regulated and the activity of the latter (and thus the cAMP level) will also be affected by the lactose metabolism which was not taken into account in our models. Hence, the model results shown in figures 4-7 actually reflect the case where TMG (non-metabolizable lactose derivate) instead of lactose is used as external signal [38,39]. However, with our modeling approach it is straightforward to introduce the negative effect of lactose on the cAMP level in the logical model (AC · ∼lactose_int → cAMP) and subsequently in the ODE model (by applying ODEfy) and then to
investigate the dynamic effects of this additional regulatory loop. Again we fitted the model to the data points as in figure 4a. Figure 8a shows a typical time course of the state variables when switching from glucose to lactose utilization (for a parameter set fitting the model well to the LacY data from figure 4a). We also studied the capability of this altered wiring diagram to induce bistable behavior (Figure 8b). As for the original model, there exist parameter sets that do not induce bistability at all whereas others have the potential. There is also no clear answer whether the looped model increases the capability of the system for bistability: although slightly more samples from the looped-model give better fit to the data while keeping strong bistability values (=distances of the two steady states), this is not yet sufficient to argue that based on the data we should choose the looped model over the non-looped one.

The samples have been obtained by running multiple (connected) local optimizations, so they do not represent independent samples of the cost function’s landscape. In order to more robustly infer model parameters and to perform model selection, we are working on implementing Bayesian reasoning and inference within the proposed model. However, this is out of the scope of this contribution.

**Figure 8:** Simulations with the “looped model”. The looped model was derived from a multivalued logical model that accounts for the negative effect of the lactose metabolism on the cAMP level (the arc AC $\rightarrow$ cAMP in the non-looped (“standard”) model in figure 3 is replaced by the hyperarc AC $\cdot$ lactose_int $\rightarrow$ cAMP). a) Simulations of the looped ODE Model. In contrast to figure 4b, the cAMP level increases during the transition from glucose to lactose utilization but decreases again when this shift has been accomplished. b): Potential of the looped and non-looped model to show bistability. Bistable behavior is quantified as the distance of the two steady states reached when growing on low lactose levels (0.2 in the ODEfy model) after preculturing either on lactose or glucose (the distance is zero if only one steady state exists). In both the standard and the looped model, we can identify parameters that fit the data (low distance to LacY) and also exhibit strong bistable behavior. The current sampling methods are not sufficient to differentiate these two models.

**5 Conclusion**

In summary, we used the lac operon as a paradigmatic example to demonstrate how qualitative knowledge can be initially captured using simple discrete (Boolean) models and then stepwise refined to multivalued logical models and finally to continuous (ODE) models. Each modeling formalism and the (forward) transformations between them are supported by CellNetAnalyzer enabling one to switch between different types of models without the need to reestablish the whole model building process.

doi:10.2390/biecoll-jib-2010-151
Acknowledgements

We are grateful to Katja Bettenbrock for helpful comments during the preparation of the manuscript. This work was supported by the German Federal Ministry of Education and Research (FORSYS-Center MaCS (Magdeburg Centre for Systems Biology) and MedSys (project SysMBo)), the Helmholtz Alliance on Systems Biology (project CoReNe) and the Ministry of Education and Research of Saxony-Anhalt (Research Center “Dynamic Systems”). We thank Jan Krumsiek and Dominik Wittmann for developing Odefy and discussing extensions thereof.

References


doi:10.2390/biecoll-jib-2010-151


