BoolSim, a Graphical User-friendly Interface for Asynchronous Simulations of Boolean Networks and Application to the Guard Cell CO2 Signaling Network.

Aravind Karanam[1,#] , David He[1,#], more authors….Sebastian Schulze[2] ,Julian I. Schroeder[2], Wouter-Jan Rappel[1*]

[1]Physics Department, University of California, San Diego, La Jolla, CA 92093, USA.
[2]Division of Biological Sciences, Section of Cell and Developmental Biology, University of California, San Diego, La Jolla, CA 92093-0116, USA.

ABSTRACT

Signaling networks are at the heart of almost all biological processes. Most of these networks contain a large number of components and often the connections between these components are either not known or the rate equations that govern the dynamics of the components are not quantified. This uncertainty in the network topology and parameters makes it challenging to formulate detailed mathematical models. Boolean networks, in which all components are either on (1) or off (0), have emerged as viable alternatives to more detailed mathematical models but can be hard to implement. Here we present BoolSim, a freely available graphical user interface (GUI) algorithm that allows users to easily construct and analyze Boolean networks. Although BoolSim can be applied to any Boolean network, we demonstrate its use using a previously published network for abscisic acid driven stomatal closure in *Arabidopsis thaliana.* We also show how BoolSim can be used to generate experimentally testable predictions by extending the network to include input from CO_2. The generated predictions were subsequently examined using quantitative experiments, which in turn resulted in modification of the network model.

Introduction

Intra-cellular signaling networks are essential in almost all biological processes. These networks are often complex, involving a large number of components (or nodes) that are inter-connected. To gain insights into these networks, it is possible to construct mathematical models. One of the strengths of these mathematical models is the ability to develop predictive outcomes of experimental perturbations (Phillips, Theory in biology: Figure 1 or Figure 7?; Shou et al, Theory, models and biology, Elife 2015;). These perturbations can be much more easily implemented in simulations than in experiments. After all, removing or changing a component or connection between components is a trivial task in simulations but usually is a task that requires lengthy wet lab experimental procedures. Predictions developed through models can enable narrowing the parameters for subsequent wet lab examination. Furthermore, mathematical models can be used to test potential biological mechanisms or can be utilized to pinpoint the most important components of a signaling network (Brockland, How computational models can help unlock biological systems 2015, Seminars in cell & developmental biology).

One way of constructing mathematical models for signaling networks is to create a rate-equation model. In such a model, the concentrations for the signaling components can take on all real values, and their change is governed by differential equations involving rate constants and the concentration of other components (Thiele and Palsson, 2011; Muraro, Daniele, et al. "The influence of cytokinin–auxin cross-regulation on cell-fate determination in Arabidopsis thaliana root development." *Journal of theoretical biology* 283.1 (2011): 152-167; Melke, Pontus, et al. "A rate equation approach to elucidate the kinetics and robustness of the TGF-β pathway." *Biophysical journal* 91.12 (2006): 4368-4380). Such models, however, quickly cease to be analytically tractable as the number of components increases. Furthermore, the strength of many of the connections between different nodes of the network is either poorly known or not known at all. This results in considerable uncertainty in parameter values, especially the rate constants, potentially limiting the value of these mathematical models, in particular for signal transduction networks for which rate constants are more difficult to define in a cellular context than for example metabolic flux networks (Feist et al., 2007).

An alternative to these “analog” networks is to formulate the problem in terms of Boolean networks (Kauffman, 1969). In these binary networks, each node can be either “on” (1, or high) or “off” (0, or low) (Bornholdt, 2008; Wang et al., 2012; Schwab et al., 2020). The state of the nodes is then determined by an update rule, which involves information from the upstream

nodes. In Boolean networks, the regulation is no longer encoded in terms of rate constants, which may or may not be quantified by experiments, but in terms of NOT, AND, and OR logic gates. For example, in the case of an AND gate, a downstream node will be turned on (i.e., 0 transitions to 1) if and only if the upstream node is on.

Despite the significant simplification associated with the binarization, Boolean networks have been shown to be able to predict behavior in a wide variety of networks, including genetic networks (Kauffman, 1969; Herrmann et al., 2012), protein networks (Dahlhaus et al., 2016), synthetic gene networks (Zhang et al., 2014), and cellular regulatory networks (Li et al., 2004; Lau et al., 2007)(Albert et al., 2017). The price one has to pay for the simplification is the loss of dynamic information. The order in which the update rules are applied can critically affect the outcome of the network. Therefore, the networks are mainly used to probe steady-state conditions for the network.

In plants, Boolean networks have been applied to genetic networks to investigate possible crosstalk (Genoud and Métraux, 1999). It has also extensively been used to study abscisic acid (ABA) driven stomatal closure in *Arabidopsis thaliana* (Li et al., 2006; Albert et al., 2017; Waidyarathne and Samarasinghe, 2018; Maheshwari et al., 2019; Maheshwari et al., 2020). This ABA signal transduction network contains a large number of components (>80), with many unknown rate constants, and is thus challenging to encode using an analog model. In a series of papers, it was shown that the formulation of a Boolean network for ABA-induced stomatal closure was able to confirm interesting experimental data (Albert et al., 2017; Maheshwari et al., 2019; Maheshwari et al., 2020). Furthermore, it was shown that the Boolean network could function as a vehicle to generate predictions. Specifically, predictions were generated through perturbations that either removed nodes or set nodes permanently to the “on” state. Some of these predictions were subsequently tested and validated in novel quantitative experiments (Albert et al., 2017).

The aforementioned studies have clearly demonstrated the potential value of casting signaling pathways into Boolean networks. However, encoding these networks, especially ones with a large number of components, might present a significant impediment to wider spread use of Boolean networks to probe, analyze, and understand signaling networks. Motivated by the challenge of creating Boolean networks, we present in this paper an open source, user-friendly algorithm that can simulate Boolean networks that can be easily formulated by the users. This

algorithm, which we term BoolSim, uses a graphical user interface (GUI) and allows the users to define nodes and their internode connections, add nodes, subtract nodes, introduce mutations, and analyze the results. BoolSim can be freely downloaded from the GitHub repository (https://github.com/dyhe-2000/BoolSim-GUI), and requires a current version of Python and C++. A detailed manual, including installation instructions, is provided in the GitHub repository as well. Although BoolSim can be applied to any Boolean network, we demonstrate its use using the abscisic acid signaling network described in Albert et al. (Albert et al., 2007). Furthermore, we describe an extension of this network that includes input from CO_2. Finally, we test predictions from this extended network using quantitative experiments.

RESULTS

Algorithm

Our algorithm, BoolSim, simulates a Boolean network with user-defined variables and interactions. Each node of the network corresponds to a Boolean variable that can be in one of two states: 0 (off) or 1 (on). In biophysical terms, an ON state of a node corresponds to a concentration of its active form that is high enough to affect change through its interactions in the system. Conversely, an OFF state corresponds to a low concentration, not able to affect change. Nodes can flip their states as a result of their interactions with other nodes and these interactions are encoded using an update equation. Some nodes, denoting external conditions or inputs, remain fixed but affect the states of other nodes.

An update equation for a given node relates its future state to the current states of all upstream nodes. While there are several ways of formulating an update equation, our algorithm uses the so-called Sum of Products (SoP) form that is intuitive and easy to formulate (Kime, Charles R., and M. Morris Mano. *Logic and computer design fundamentals*. Prentice Hall, 2003.). This form consists of a series of terms linked by the OR logic operator. Each of these terms is a product, which contain Boolean variables and/or their negations connected by the AND operator. A typical example reads:

AnionEM = SLAC1 | SLAH3 & QUAC1

where the symbols | and & stand for OR and AND, respectively. In this example, the future state of node AnionEM is determined by the current states of three other nodes: SLAC1, SLAH3, and QUAC1. Specifically, AnionEM is set to 1 or remains 1 if either a) SLAC1 is 1, or b) both SLAH3 and QUAC1 are 1. If neither condition is met, AnionEM remains 0 or is set to 0. The Supplementary Text has a brief primer on Boolean algebra and several illustrations for formulating Boolean equations using the SoP form.

In a typical simulation in BoolSim, there are a few nodes which affect the nodes downstream of them but do not have any nodes upstream; as such their states remain unchanged during the simulation. We call such nodes “input” nodes. There is typically one “output” node denoting the product or end state of the pathway which can be monitored to determine the function of the network. For example, in the case of the ABA signaling network, ABA, Nitrite, GTP, etc., are inputs and the output node is “Closure”, representing stomatal closure (Alberts et al., 2017). The initial states of other network nodes can be specified, based on prior knowledge, or can be assigned a random (0 or 1) value.

Once the initial states of all the nodes are specified, the program evaluates the update equations in a random order and exactly once for each node, with the output node evaluated last. The update for a single node is based on the current state of the network so that some of its upstream nodes may already have been updated in the same time step. This so-called asynchronous updating is motivated by the fact that many of the reaction rates are unknown, resulting in non-deterministic outcomes. A single time step in the simulation corresponds to each node being updated once. The number of time steps, corresponding to one iteration over all the nodes, can be chosen by the user and should be large enough for the system to reach a steady state. Furthermore, the user also specifies the number of simulations, each with randomly chosen initial conditions. All parameters, equations, and initial conditions can be easily entered into BoolSim using an intuitive graphical interface. Finally, BoolSim is able to display graphs of the dynamics of one or more nodes and all variables are stored for later analysis.

Simulations

Simple example

In a first example, we investigate a very simple Boolean network, shown in Fig. 1A. This network contains an input node (IN), an output node (OUT), and three intermediate nodes X, Y, and Z. X

and Y reinforce each other while X inhibits Z. Furthermore, IN inhibits X, Y inhibits OUT and Z activates OUT. For both X and OUT, the two inputs combine according to AND logic as denoted. The Boolean update equations can thus be written as follows

IN = IN

X = Y and (not IN)

Y = X

Z = not X

OUT = (not Y) and Z

The dynamics of this network can be worked out by hand or, since it only contains 3 nodes, and can be analyzed graphically. One feature of an asynchronous update scheme is that the updated state is always the 'nearest neighbor' of the previous state. Therefore, the evolution of the states can be represented as a continuous trajectory through the state space with dimensions equal to the number of nodes. This is shown in Fig. 1B&C for our simple model in the absence (IN = 0; B) and in the presence of input (IN = 1; C). The 3-dimensional state space is spanned by X, Y, and Z and all possible states of the intermediate nodes are points in this 3-dimensional space that are connected by arrows according to the rules of the Boolean network. By following these arrows, the attractors for the model can be determined. For example, in the absence of input (Fig. 1B), the state X=1, Y=0, and Z=0, compactly written as (1,0,0) (Fig. 1B) can either transition to (0,0,0) or to (1,1,0). Since no arrows originate from (1,1,0), this is an attractor of the system: the state will remain unchanged indefinitely. Furthermore, the only permissible transition from (0,0,0) is to (0,0,1), which can easily be seen to be an attractor as well. Since only (0,0,1) leads to the ON state of the OUT node, we find that OUT=1 in 50% of the possible initial conditions. In the presence of input, however, it is easy to verify that the only possible attractor is (0,0,1) and thus OUT = 1 for all initial conditions. Calculations and further analysis on this simple network are provided in the supplementary text.

The implementation in BoolSim is shown in Fig. 1D, obtained after specifying the input files in the folder sample_data_files/simple_network_data_files of the repository. Here, the green/red arrows indicate positive/negative regulation and an arrow within the node indicates self-regulation (green/red for positive/negative). In BoolSim, all nodes are visualized in yellow by default. Connections between a particular node and other network nodes, however, can be easily visualized by double clicking on the specific node, after which it changes color to orange. Upstream nodes will then change color to magenta while downstream nodes will turn cyan. In

the example of Fig. 1D, this has been carried out for node X. Note that the color scheme for the nodes can be changed by the user (see Supplementary Text).

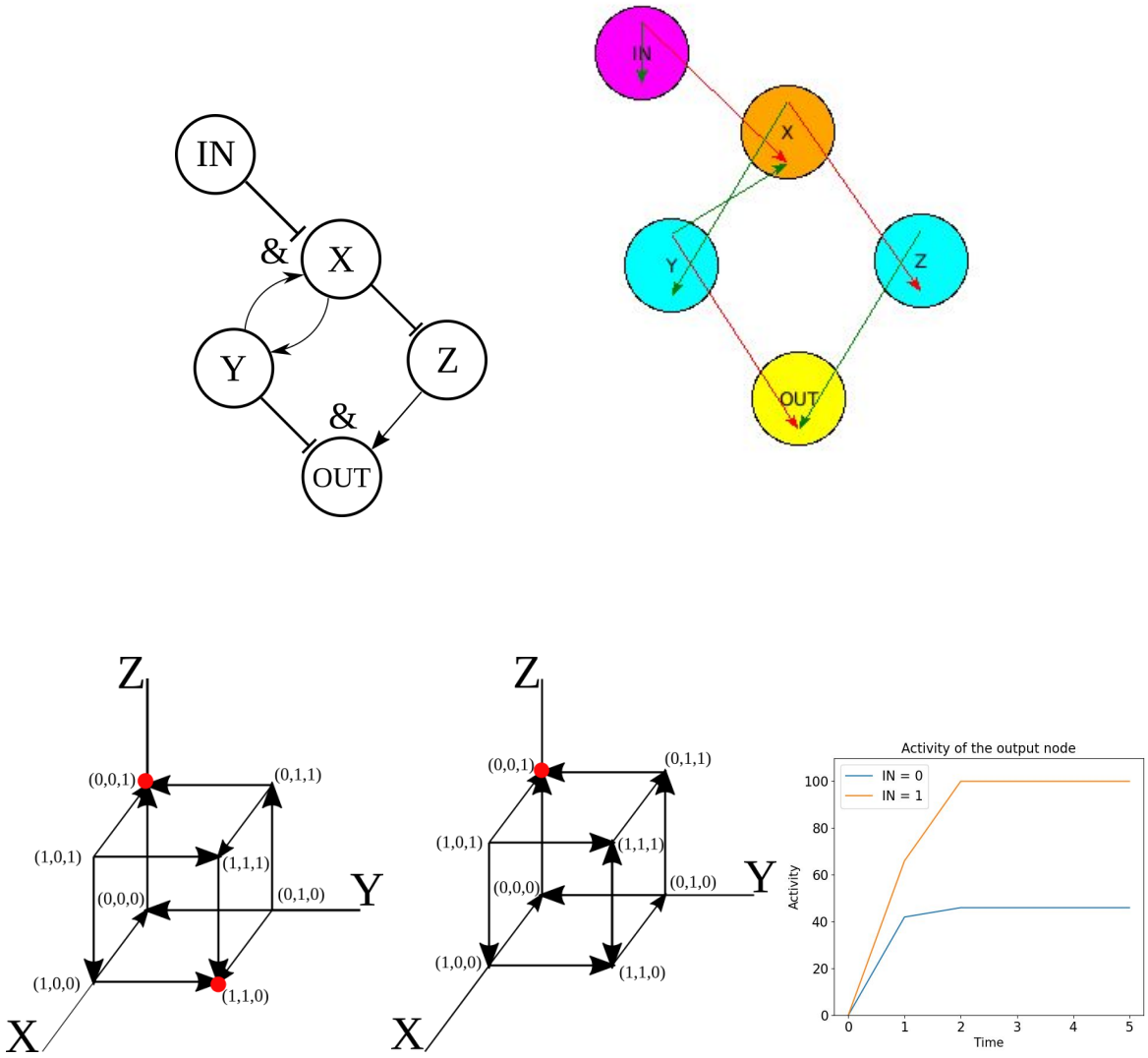

Figure 1: **A**: A schematic of the simple Boolean network. Pointed arrowheads indicate positive regulation and flat arrowheads indicate negative regulation. For nodes X and OUT, the two inputs combine using AND logic. **B:** State space and dynamics, represented by arrows, in the absence of input (IN = 0). The two possible attractors, (0,0,1) and (1,1,0), are indicated by red dots. **C:** State space and dynamics in the presence of input (IN = 1). The only one attractor, (0,0,1), is indicated by a red dot and leads to OUT = 1. **D**: Representation of the Boolean model in our GUI. Green and red arrows indicate positive and negative regulation, respectively. An

arrow within the node indicates self-regulation, which can be positive or negative. In this representation, one can 'search' for upstream and downstream nodes of a given node. Here, X (colored in orange) has two downstream nodes, colored in cyan and one upstream node, colored in magenta. Other node(s) are colored in yellow. **E**: Percentage activity of the output node vs time in an asynchronous update scheme starting with randomly assigned initial states for the intermediate nodes when the input is present (IN = 1, orange line) and absent (IN = 0, blue line).

This network was simulated using BoolSim, choosing 5 time steps and 50 different sets of initial conditions. The results of these simulations are shown in figure 1E where we plot the state of the output node, expressed as the percentage of runs in which OUT=1, in the absence (blue line) and presence (orange line) of input as a function of time.

ABA and CO_2 network

Next, we applied BoolSim to the ABA-induced stomatal closure network, adapted from Albert *et al.* (Albert et al., 2017). The input file for this network, containing all components by name and their interactions, can be found in the subfolder sample_data_files/ ABA_CO2_data_files/ of the repository for the Boolean equations and for the names of the node. A screenshot of this network (Alberts et al., 2017) encoded in the GUI is presented in Fig. 2, with the input ABA node shown in red and the "Closure" output node shown in green. This network contains 81 nodes, including input and output nodes, and was constructed by Albert *et al.* following an extensive survey of over a hundred peer-reviewed articles published through 2015. As in the simple example, the interactions between nodes in the GUI are color-coded, with green arrows representing positive interactions and red arrows representing negative interactions. Using the GUI interface, the user can move nodes around by simply dragging them to a new location. Furthermore, to facilitate examining inter-node connections, double clicking on a node reveals all downstream and upstream interactions of that node.

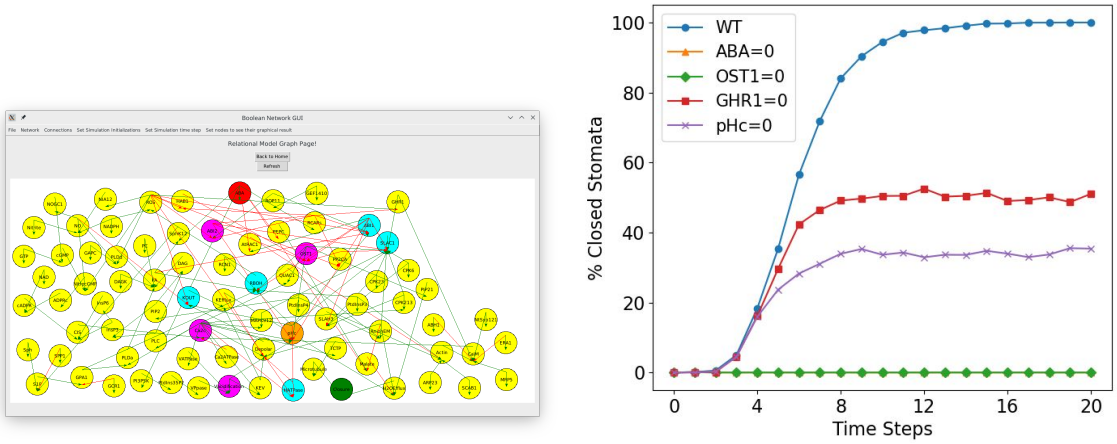

Figure 2: A: Visualization of the Boolean network for ABA-induced stomatal closure, rendered by the BoolSim GUI. The input ABA and output Closure are colored in red and green, respectively. The node denoting cytoplasmic pH (pHc), colored in orange, has its upstream nodes colored in magenta and downstream nodes in cyan. B: Percentage of closed stomata as a function of time for the wild type (WT) and the indicated representative mutants for the ABA-induced stomatal closure network.

Results of the BoolSim simulations for 20 timesteps and averaged over 2,500 initial conditions are presented in Fig. 2B, where each curve reports the percentage of stomatal closure among the copies of the simulations as a function of time. In the absence of ABA, simulated by setting the input node ABA to 0, the output node Closure is 0 for all timesteps, corresponding to 0% stomatal closure. Wild type (WT) cells, shown in blue, reach 100% stomatal closure after approximately 15 timesteps. Following the network defined by Albert et al., we have computed the stomatal closure for a number of “mutants”, labeled in Fig. 2B. For example, knocking out OST1 corresponds to forcing the node OST1 to 0 at all times and results in 0% stomatal closure. The results are similar to the ones obtained in the original publication, validating our algorithm.

In the present of this network, we did not aim to modify the ABA signaling model of Alberts et al (2017), but to examine whether other stimuli can be integrated into this network and then run simulations. To illustrate how one could use our GUI interface to examine and explore networks, we have taken the original ABA network and have extended it with a putative branch that models the input of carbon dioxide CO_2. Elevated CO2 closes stomatal pores and some genetic

elements of CO2 signaling overlap with those of ABA signaling, whereas others indirectly affect the CO2 output (Hsu et al., 2018; Zhang et al., 2020). Based on previous experimental data, we modeled the CO_2 branch to be upstream from GHR1 in the ABA network (Horak et al., 2016; Jakobson et al., 2016). The added branch contains CO_2 as input, which activates the beta carbonic anhydrases bCA4 and bCA1 (Hu et al., 2010; Hu et al., 2015), these then activate the node MPK12/MPK4. This node inhibits HT1, which, in turn, is assumed to activate CBC1/CBC2. Finally, CBC1/CBC2 enters the ABA network through an assumed inhibitory link to GHR1 (Fig. 3A).

Thus, this branch can be translated into the following Boolean equations:

CO_2 = CO_2

add: bCA4 + bCA1

MPK12/MPK4 = bCA4 + bCA1 CO_2

HT1 = not MPK12/MPK4

CBC1/CBC2 = HT1

GHR1 = (not ABI2) and ROS and (not CBC1/CBC2)In the CO2 network, right after CO2, can you add:

Note that the equation for GHR1 takes into account the existing connections from the ABA network (from ABI2 and ROS) as well. This simplified CO2 signaling model includes the presently identified and confirmed early CO2 signaling mechanisms that have been found to function in the CO2 signaling pathway upstream of the merging with the ABA-induced stomatal closing pathway (Hsu et al., 2018; Zhang et al., 2018; Zhang et al, 2020).

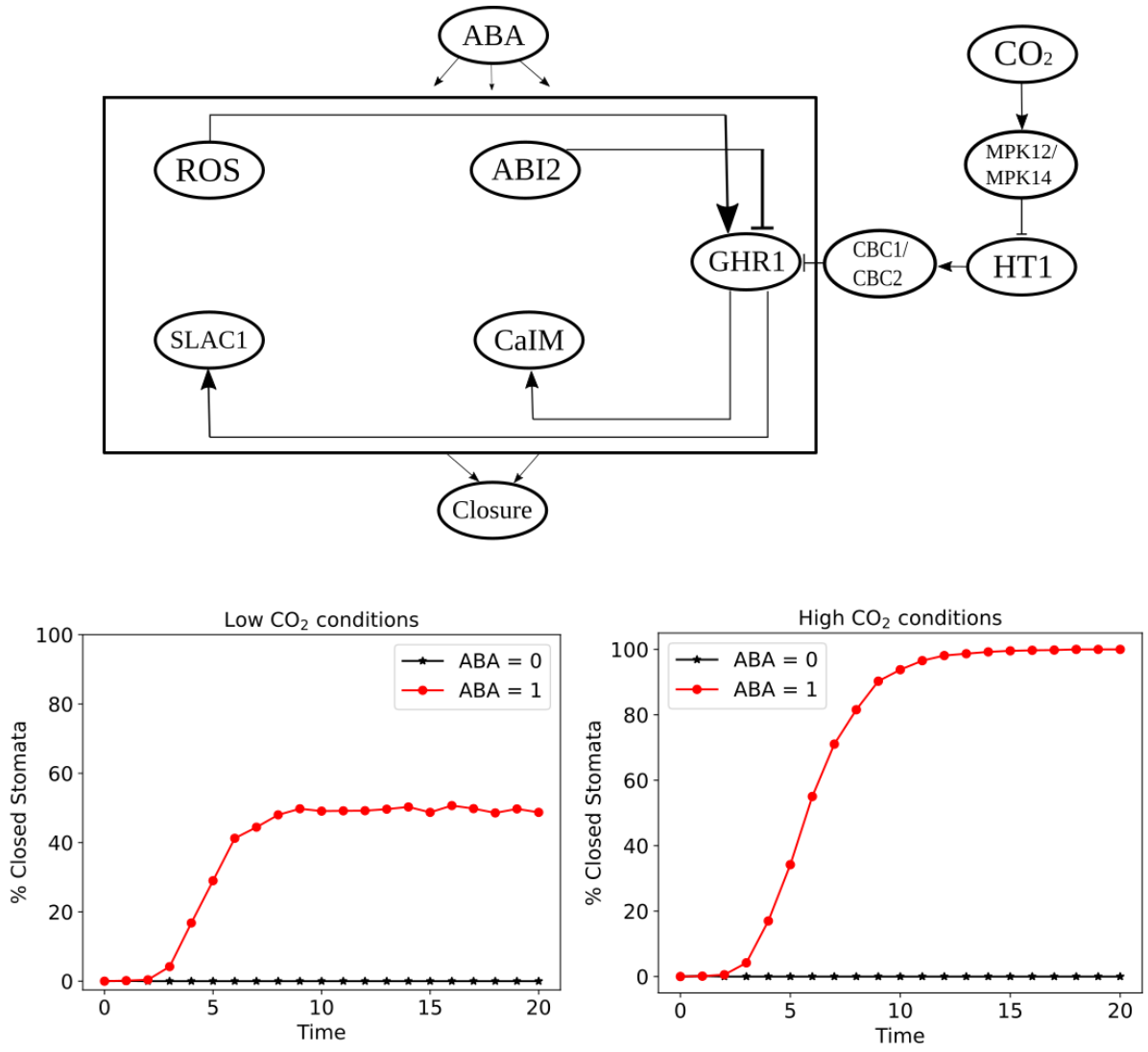

Figure 3: **Top**: ABA-driven stomatal closure model extended with a CO_2 branch, which negatively regulates GHR1. The box denotes all the intermediate nodes of the original ABA network shown in Fig. 2 with only GHR1 and its immediate upstream and downstream nodes shown.

Bottom: Percentage of stomatal closure upon the application of ABA in low (left) and high (right) CO_2 conditions.

Based on this model, we then how the extended network responds in simulations to ABA under high and low CO_2 conditions. This is simulated by setting the input node for CO_2 to either 0 (low concentration) or 1 (high concentration). The latter, of course, corresponds to the previously simulated WT case since high CO_2 does not alter the ABA network. Our simulations predicted,

however, that in the presence of low CO_2, the percentage of stomatal closure is reduced from 100% to 50%. Thus, our prediction is that under the low CO_2 conditions, the introduction of ABA results in a marked reduction in stomatal closure percentage and, thus, an increase in the stomatal conductance.

To test our predictions experimentally, we analyzed ABA-mediated stomatal closure under either 400 ppm or ~1.5 ppm [CO_2] by conducting gas-exchange experiments with ABA application to the transpiration stream of excised intact leaves (Ceciliato et al., 2019). Our results show elevated steady state stomatal conductance in leaves exposed to 1.5 ppm [CO_2] compared to leaves exposed 400 ppm [CO_2], which indicates increased stomatal opening (Fig. 4A). Application of 2 µM ABA induced robust stomatal closure in leaves exposed to 400 ppm [CO_2]. Leaves equilibrated under 1.5 ppm [CO_2] also reacted with stomatal closure to 2 µM ABA, although to a lesser extent (Fig. 4B). The absolute change in stomatal conductance was similar 60 min after exposure to ABA in leaves incubated under 400 ppm or 1.5 ppm [CO_2] (Fig. 4B). However, the rate of stomatal closing was slowed in leaves exposed to 1.5 ppm CO2 (t2-t1 in Fig. 4B). Furthermore, although the absolute change in stomatal conductance of leaves exposed to 1.5 ppm [CO_2] in response to ABA was similar to WT, stomatal conductance in these leaves only reached values which were comparable to steady state conductance of leaves incubated at 1.5 ppm [CO_2]. This could be attributed to the higher stomatal conductance before ABA exposure. This suggests that these stomata are clearly partially open after application of ABA (Fig. 4A).

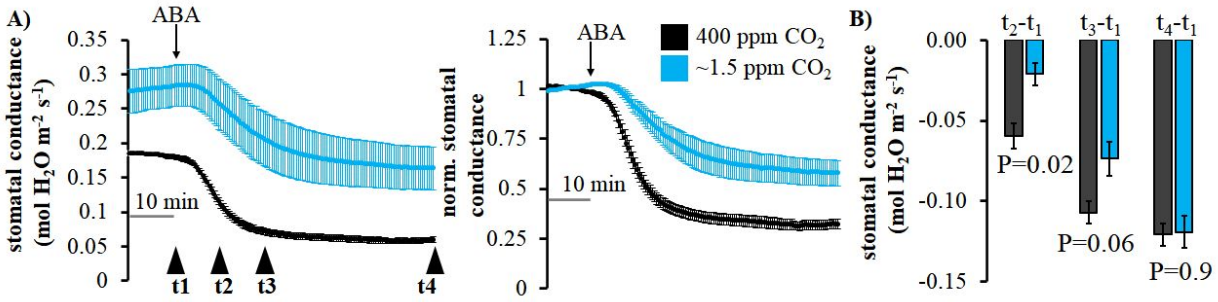

Figure 4: ABA-mediated stomatal response of WT leaves during CO_2 starvation. A) Intact excised leaves of wild-type plants (n=3 independent leaves per treatment) were either equilibrated at 400 ppm CO_2 or ~1.5 ppm CO_2 for 50 min prior to stomatal conductance measurements. Stomatal conductance (left panel) was normalized to the average of the first 10 minutes of stomatal conductance values recorded (right panel). Black downward pointing arrows indicate the application of 2 µM ABA to the transpiration stream. B) Differences in absolute

stomatal conductance (mean ± SEM) were calculated at the indicated time points (t1=application of ABA, t2=10 min after application of ABA, t3=final measurement). One-Way ANOVA was used for statistical tests.

Discussion

Large signaling networks are common in biology in general, and plant physiology in particular. Published ABA signaling networks, for example, contain more than 80 components. The vast majority of the interaction strengths between these components is not known, making it difficult to formulate mathematical models of these networks. Motivated by the simplicity and utility of Boolean networks and the challenges associated with formulating detailed rate equation based models for these large networks, we have presented here a software package with a graphical user interface (GUI) that can simulate, visualize, and plot the results of a user-defined Boolean network. Our package, named BoolSim, is free to use and distribute, and is built from free and open-source software. The interface is intuitive and users do not require extensive coding knowledge to use it. The Supplementary Text contains detailed instructions on downloading, installing, and running BoolSim.

BoolSim uses asynchronous and random order of update, which is best suited to simulate a network of chemical reactions in which the outcome of one reaction now affects the outcome of another in the near future (hence asynchronous), and when the relative rates of different reactions in the network are unknown (hence random order of update). Besides nodes and connections, users may also specify the number of time steps or iterations to run the simulations and the number of initial conditions to get a statistically robust sample. Once a system is defined, it may be visualized as a network in BoolSim. Visualization includes identifying the upstream and downstream nodes of a given node and the type of connections (activating or inhibiting) between them. The steady states of nodes of the system after simulation can be quickly plotted within BoolSim. The trajectory of the entire simulation is stored in a NumPy array; a Jupyter notebook is provided with the package that can be used to further analyze the system starting from the NumPy array, including producing publication-ready plots of the simulation.

We first tested BoolSim using a model for the stomatal closure in guard cells as mediated by abscisic acid (ABA). This ABA-mediated pathway was constructed by Albert *et al.* following an extensive survey of published results through 2015. As tests of the programmed network here, our simulations were consistent with those of Albert *et al.*. We also extended the network to

include the effects of CO_2 on stomatal movements To that end, based on present understanding, we added a signaling branch leading from CO_2 to GHR1 in the network. The simple CO2 signaling model investigated here does recapitulate findings showing that the two merging pathways, ABA signaling and CO2 signaling, amplify each other (Raschke, 1975; Hsu et al., 2018).

Previous research had indicated that CO2 may mediate signal transduction via the OST1 protein kinase, as *ost1* mutant leaves greatly slow and impair the stomatal response to CO2 elevation (Xue S et al., 2011; Merilo et al., 2013). However, more recent studies unexpectedly showed that CO2 elevation does not activate the OST1 protein kinase, in contrast to abscisic acid (Hsu et al., 2018; Zhang et al., 2020). This research further showed that basal OST1 protein kinase activity and basal ABA signaling are required for WT-like CO2-induced stomatal closure (Hsu et al., 2018; Zhang et al., 2020) (Figs. 3 and 4). The biochemical link by which CO2 signaling merges with ABA signaling is thus proposed to lie downstream of the OST1 protein kinase, but remains unknown. In the present study, we modeled this link to occur at the level of the transmembrane (pseudo)receptor-like kinase GHR1 (Hua et al., 2012; Sierla et al., 2018), although further research is needed to determine the precise mechanism by which CO2 signaling merges with abscisic acid signal transduction.

The simulations of the modified network predicted that the response to ABA should depend critically on the CO_2 concentration. This prediction was then subsequently analyzed and ABA-mediated stomatal closure of intact leaves was measured while leaves were either exposed to ambient 400 ppm [CO_2] or low (1.5 ppm) [CO_2] (Fig. 4).Our data show, that leaves exposed to 1.5 ppm [CO_2] did partially close in response to ABA, albeit stomatal conductance only reached values similar to steady state stomatal conductance of leaves exposed to 400 ppm [CO_2] prior to ABA application. This indicated that stomata of leaves under 1.5 ppm [CO_2] were still relatively open even after application of ABA (Fig. 4).

Our proposed addition to the existing ABA network is quite simple in that it is a linear chain that affects only a signal component and was also investigated here to illustrate the potential use of BoolSim. As indicated above, it is conceivable that CO_2 affects yet unknown mechanisms and this pathway may contain feedback loops.

Further experiments may be able to shed more light on the precise topology of the mechanisms that mediate CO_2 signaling and its connection to the ABA signaling network. Nevertheless, the simple addition of a CO2 signaling branch, led to testing of the effect of near CO2 free conditions (1.5 ppm CO2) on ABA signaling. Results from wet lab experiments in turn led to modification of the guard cell signaling model. This updated model was able to predict a stomatal ABA response that can proceed even at only 1.5 ppm CO2, but that none-the-less depends on an amplifying effect of CO_2 signaling on ABA-induced stomatal closing. Furthermore, the graphical user interface and stomatal signaling model developed here can be used and altered by users to test diverse predictions and to add expanded components to the model. Thus the method and software tools presented here can be of interest to the wider plant biology community interested in physiological pathways and may be used to generate further predictions. Importantly, the BoolSim platform developed here can be applied to any Boolean network and can be used to generate publicly manipulatable Boolean models for any desired process.

Acknowledgements

This research was supported by a grant from the National Science Foundation (MCB-1900567) to WJR and JIS. S.S. was supported by a Deutsche Forschungsgemeinschaft (DFG) fellowship (SCHU 3186/1-1:1).

References

Albert R, Acharya BR, Jeon BW, Zañudo JG, Zhu M, Osman K, Assmann SM (2017) A new discrete dynamic model of ABA-induced stomatal closure predicts key feedback loops. PLoS biology **15:** e2003451

Alon, U. (2007). *An introduction to systems biology: design principles of biological circuits*. CRC press.

Bornholdt S (2008) Boolean network models of cellular regulation: prospects and limitations. Journal of the Royal Society Interface **5:** S85-S94

Dahlhaus M, Burkovski A, Hertwig F, Mussel C, Volland R, Fischer M, Debatin K-M, Kestler HA, Beltinger C (2016) Boolean modeling identifies Greatwall/MASTL as an important regulator in the AURKA network of neuroblastoma. Cancer letters **371:** 79-89

Ceciliato PHO, Zhang J, Liu Q, Shen X, Hu H, Liu C, Schäffner AR, Schroeder JI. (2019). Intact leaf gas exchange provides a robust method for measuring the kinetics of stomatal conductance responses to abscisic acid and other small molecules in *Arabidopsis* and grasses. Plant Methods **15:** 38.

Genoud T, Métraux J-P (1999) Crosstalk in plant cell signaling: structure and function of the genetic network. Trends in plant science **4:** 503-507

Herrmann F, Groß A, Zhou D, Kestler HA, Kühl M (2012) A boolean model of the cardiac gene regulatory network determining first and second heart field identity. PloS one **7:** e46798

Kauffman SA (1969) Metabolic stability and epigenesis in randomly constructed genetic nets. Journal of theoretical biology **22:** 437-467

Lau K-Y, Ganguli S, Tang C (2007) Function constrains network architecture and dynamics: A case study on the yeast cell cycle Boolean network. Physical Review E **75:** 051907

Li F, Long T, Lu Y, Ouyang Q, Tang C (2004) The yeast cell-cycle network is robustly designed. Proceedings of the National Academy of Sciences **101:** 4781-4786

Li S, Assmann SM, Albert R (2006) Predicting essential components of signal transduction networks: a dynamic model of guard cell abscisic acid signaling. PLoS Biol **4:** e312

Maheshwari P, Assmann SM, Albert R (2020) A guard cell abscisic acid (ABA) network model that captures the stomatal resting state. Frontiers in physiology **11:** 927

Maheshwari P, Du H, Sheen J, Assmann SM, Albert R (2019) Model-driven discovery of calcium-related protein-phosphatase inhibition in plant guard cell signaling. PLoS computational biology **15:** e1007429

Schwab JD, Kühlwein SD, Ikonomi N, Kühl M, Kestler HA (2020) Concepts in Boolean network modeling: What do they all mean? Computational and Structural Biotechnology Journal

Sun, Z., Jin, X., Albert, R., & Assmann, S. M. (2014). Multi-level modeling of light-induced stomatal opening offers new insights into its regulation by drought. *PLoS Comput Biol*, *10*(11), e1003930.

Waidyarathne P, Samarasinghe S (2018) Boolean calcium signalling model predicts calcium role in acceleration and stability of abscisic acid-mediated stomatal closure. Scientific reports **8:** 1-16

Wang R-S, Saadatpour A, Albert R (2012) Boolean modeling in systems biology: an overview of methodology and applications. Physical biology **9:** 055001

Zhang H, Lin M, Shi H, Ji W, Huang L, Zhang X, Shen S, Gao R, Wu S, Tian C (2014) Programming a Pavlovian-like conditioning circuit in Escherichia coli. Nature communications **5:** 1-10

LEFT OVER PIECES:

Mathematical and computational models of biological pathways are often used to contextualize and summarize the wealth of available experimental data and can provide insights into the topology of pathways (Alon, 2007). Boolean networks offer a simple and straightforward way to simulate large signaling networks and can be used to analyze the relationships between the different network components. They are particularly useful to study the steady state(s) of the

system that result from the connectivity and the initial state of the system. In theory, specifying the connectivity of the network - which includes specifying if and how any two nodes are connected in the network - completely determines the system's set of states and its dynamics. In practice, enumerating or simulating all the states to understand the dynamics of the system is infeasible if not impossible. However, averaging over a sufficiently large random sample (a few thousand initial states) leads one to the same conclusions as would simulating the entire system. Further, there are no rate constants or other parameters in Boolean networks, hence one cannot predict or infer such quantities by simulating them. Individual nodes in a boolean network can assume only one of two values (HIGH or LOW), and making a correspondence between these binary states and analog values from experimental data is not straightforward. We note, however, that a few hybrid models that allow discrete but multiple (more than two) states for each node have also been proposed to model biological pathways (Sun *et al.,* 2014).

Model update sequence based on simulation results

In Albert *et al*'s 2017 model, the update equation for QUAC1 is:

QUAC1 = OST1 & Ca2c

Initially, we proposed a modification to the above equation as (change in boldface).

QUAC1 = OST1 & Ca2c **and GHR1**

Based on my notes, I do not remember this modification being a result of some simulations we ran. We could check with Julian the reason for this change. In the absence of the CO_2 branch and other mutations in the network, both the equations above display the same output, namely 100% closure when ABA=1 and 0% closure when ABA=0.

When GHR1 is set to 0 and we use the first equation above, we get a 50% closure in the presence of ABA. When GHR1 is set to 0 and we use the second equation above, we get a 0% closure in the presence of ABA. This is because in the former case, setting GHR1 to 0 doesn't affect the state of QUAC1.

The CO_2 branch feeds into GHR1. Since the CO_2 branch is a linear chain, we can read off the state of GHR1 from that of CO_2, independent of the state of ABA: if CO_2 = 1, then GHR1 = 1; if CO_2 = 0, then GHR1 = 0. Finally, the percentage of stomatal closure when CO_2=0 and ABA=1 depends on which equation for QUAC1 is used: if we use the first, we get 50% closure; if we use the second, we get 0% closure. Since the second equation doesn't agree with our intuition and/or the experiments, we *went back* to the equation proposed by Albet *et al.*