Modular Control of Biological Networks

David Murrugarra[1] Alan Veliz-Cuba[2] Elena Dimitrova[3] Claus Kadelka[4]
Matthew Wheeler[5] Reinhard Laubenbacher[5]

April 2023

[1]Department of Mathematics, University of Kentucky, Lexington, KY 40506, USA
[2]Department of Mathematics, University of Dayton, Dayton, OH 45469, USA
[3]Mathematics Department, California Polytechnic State University, San Luis Obispo, CA 93407, USA
[4]Department of Mathematics, Iowa State University, Ames, IA 50011, USA
[5]Department of Medicine, University of Florida, Gainesville, FL 32610, USA

Abstract

The concept of control is central to understanding and applications of biological network models. Some of their key structural features relate to control functions, through gene regulation, signaling, or metabolic mechanisms, and computational models need to encode these. Applications of models often focus on model-based control, such as in biomedicine or metabolic engineering. This paper presents an approach to model-based control that exploits two common features of biological networks, namely their modular structure and canalizing features of their regulatory mechanisms. The paper focuses on intracellular regulatory networks, represented by Boolean network models. A main result of this paper is that control strategies can be identified by focusing on one module at a time. This paper also presents a criterion based on canalizing features of the regulatory rules to identify modules that do not contribute to network control and can be excluded. For even moderately sized networks, finding global control inputs is computationally very challenging. The modular approach presented here leads to a highly efficient approach to solving this problem. This approach is applied to a published Boolean network model of blood cancer large granular lymphocyte (T-LGL) leukemia to identify a minimal control set that achieves a desired control objective.

Keywords— Boolean networks, modularity, control, canalization, gene regulatory networks.

1 Introduction

With the availability of more experimental data and information about the structure of biological networks, computational modeling can capture increasingly complex features of biological networks [1, 2]. However, the increased size and complexity of dynamic network models also poses challenges in understanding and applying their structure as a tool for model-based control, important for a range of applications [3, 4]. This is our focus here. To narrow the scope of the problems we address we limit ourselves to intracellular networks represented by Boolean network (BN) models. BNs are widely used in molecular systems biology to capture the coarse-grained dynamics of a variety of regulatory networks [5]. They have been shown to provide a good approximation of the dynamics of continuous processes [6].

For the commonly-used modeling framework of ordinary differential equations, there is a well-developed theory of optimal control, which is largely absent for other modeling frameworks, such as Boolean networks or agent-based models, both frequently used in systems biology and biomedicine. Furthermore, control inputs, in many cases, are of a binary nature, such as gene knockouts or the blocking of mechanisms. For BNs, there is no readily available mathematical theory that could be used for control, leaving sampling and simulation. As networks get larger, with hundreds [7] or even thousands of nodes [8], this leaves few computational tools to identify control inputs for achieving preselected objectives, such as moving a network from one phenotype (e.g., cancer) to another (e.g., normal). One approach is to reduce the system in a way that the reduced system maintains relevant dynamical properties such as its attractors [9, 10]. This allows the control methods to be applied to the reduced system, and the same controls can then be used for the original system.

Control targets in Boolean networks have been identified by a variety of approaches: using stable motifs [11, 12], feedback vertex sets [13, 14], model checking [15, 16], and other methods [17, 18]. A few approaches have used strongly connected components for model analysis by decomposing the wiring diagram or the state space of the network [19, 20]. However, to our knowledge, none of the approaches developed thus far exploit the modular structure exhibited by many biological systems, in order to identify control strategies by focusing on one module at a time, which is the approach used in this paper.

Modularity refers to the division of the system into separate units, or modules, that each have a specific function [21, 22]. Modularity is a fundamental property of biological systems that is essential for the evolution of new functions and the development of robustness [23, 24]. In [25], we developed a mathematical theory of modularity for Boolean network models and showed that one can identify network-level control inputs at the modular level. That is, we obtain global control inputs by identifying them at the local, modular level and assembling them to global control. This enables network control for much larger networks than would otherwise be computationally feasible. It is worth noting that, although the ideas and concepts behind our approach to modularity are intuitive and natural, to our knowledge, there is no well-developed mathematical theory of modular decomposition of Boolean networks and the use of modularity for the purpose of control. In this paper, we further develop this approach into a mathematical theory of biological network control.

We further propose to use another property of biological networks, represented through Boolean network models. Almost all Boolean rules that describe the dynamics of over 120 published, expert-curated biological Boolean network models have the property that they exhibit some degree of canalization [26]. A Boolean function is canalizing if it has one or more variables that, when they take on a particular value, they determine the value of the function, irrespective of the values of all the other variables. As an example, any variable in a conjunctive rule (e.g., $x \cap y \cap z$) determines the value of the entire rule, when it takes on the value 0. We derive a criterion for Boolean network models whose Boolean functions are all canalizing, that can be used to exclude certain modules from needing to be considered for the identification of controls.

Our approach to control via modularity is summarized in Figure 1. We decompose the network into its constituent modules, then apply control methods to each module to identify a control target for the entire network. We show that by combining the controls of the modules, we can control the entire network. In the last part of the paper, we present theoretical results that exploit the canalizing properties of the regulatory functions to exclude certain modules from the control search. Finally, we demonstrate our approach by applying it to a published model of the blood cancer large granular lymphocyte (T-LGL) leukemia [27].

2 Background

We first describe Boolean networks and how to decompose a network into modules. In a BN, each gene is represented by a node that can be in one of two states: ON or OFF. Time is discretized as well, and the state of a gene at the next time step is determined by a Boolean function that takes as input the current states of a subset of the nodes in the BN. The dependence of a gene on the state of another gene can be graphically represented by a directed edge, and the *wiring diagram* contains all such dependencies.

2.1 Boolean Networks

Boolean networks can be seen as discrete dynamical systems. Specifically, consider n variables $x_1, \dots, x_n$ each of which can take values in $\mathbb{F}_2 := \{0, 1\}$, where $\mathbb{F}_2$ is the field with two elements, 0 and 1, where arithmetic is performed modulo 2. Then, a synchronously updated Boolean network is a function $F = (f_1, \dots, f_n) : \mathbb{F}_2^n \to \mathbb{F}_2^n$, where each coordinate function f_i describes how the future value of variable x_i depends on the present values of all variables. All variables are updated at the same time (synchronously).

Definition 2.1. *The* wiring diagram *of a Boolean network $F = (f_1, \dots, f_n) : \mathbb{F}_2^n \to \mathbb{F}_2^n$ is the directed graph with vertices $x_1, \dots, x_n$ and an edge from x_i to x_j if f_j depends on x_i. That is, if there exists $\mathbf{x} \in \mathbb{F}_2^n$ such that $f_j(\mathbf{x}) \neq f_j(\mathbf{x} + \mathbf{e_i})$, where $\mathbf{e_i}$ is the ith unit vector.*

Definition 2.2. *The wiring diagram of a Boolean network is* strongly connected *if every pair of nodes is connected by a directed path. That is, for each pair of nodes x_i, x_j in the wiring diagram with $x_i \neq x_j$ there exists a directed path from x_i to x_j (and vice versa). In particular, a one-node wiring diagram is strongly connected by definition.*

Remark 2.3. *The wiring diagram of any Boolean network is either strongly connected or it is composed of a collection of strongly connected components where connections between different components move in only one direction.*

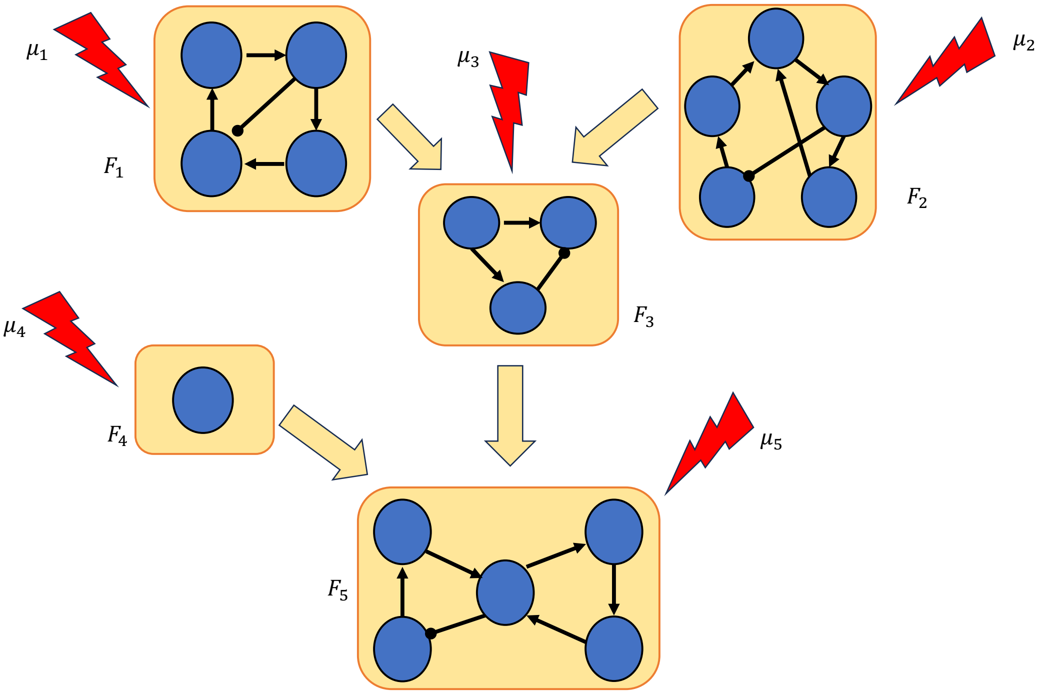


Figure 1: Control via modularity. First, the network is decomposed into its constituent modules: $F_1, \ldots, F_m$. Then, controls $\mu_1, \ldots, \mu_m$ are identified for each module. Combining the controls of the modules $\mu = (\mu_1, \ldots, \mu_m)$ yields a set of controls for the whole network.

Example 2.4. *Figure 2a shows the wiring diagram of the Boolean network $F : \mathbb{F}_2^3 \to \mathbb{F}_2^3$ given by*

$$F(x_1, x_2, x_3) = (x_2 \wedge \neg x_3, x_3, \neg x_1 \wedge x_2).$$

2.2 Dynamics of Boolean networks

Another directed graph associated with a BN is the *state transition graph*, also referred to as the *state space*. It describes all possible transition of the BN from one time step to another. The *attractors* of a BN are minimal sets of states from which there is no escape as the system evolves. An attractor with a single state is also called a *steady state* (or fixed point). In mathematical models of intracellular regulatory networks, the attractors of the model are often associated with the possible phenotypes of the cell. This idea can be traced back to Waddington [28] and Kauffman [29]. For example, in a model of cancer cells, the steady states of the model correspond to proliferative, apoptotic, or growth-arrest phenotypes [30]. Mathematically, a phenotype is associated with a group of attractors where a subset of the system's variables have the same states. These shared states are then used as biomarkers that indicate diverse hallmarks of the system.

There are two ways to describe the dynamics of a Boolean network $F : \mathbb{F}_2^n \to \mathbb{F}_2^n$, (i) as trajectories for all 2^n possible initial conditions, or (ii) as a directed graph with nodes in $\mathbb{F}_2^n$. Although the first description is less compact, it will allow us to formalize the dynamics of coupled networks.

Definition 2.5. *A trajectory of a Boolean network $F : \mathbb{F}_2^n \to \mathbb{F}_2^n$ is a sequence $(x(t))_{t=0}^{\infty}$ of elements of $\mathbb{F}_2^n$ such that $x(t+1) = F(x(t))$ for all $t \geq 0$.*

Example 2.6. *For the network in the example above, $F(x_1, x_2, x_3) = (x_2 \wedge \neg x_3, x_3, \neg x_1 \wedge x_2)$, there are $2^3 = 8$ possible initial states giving rise to the following trajectories (commas and parenthesis for states are omitted for*

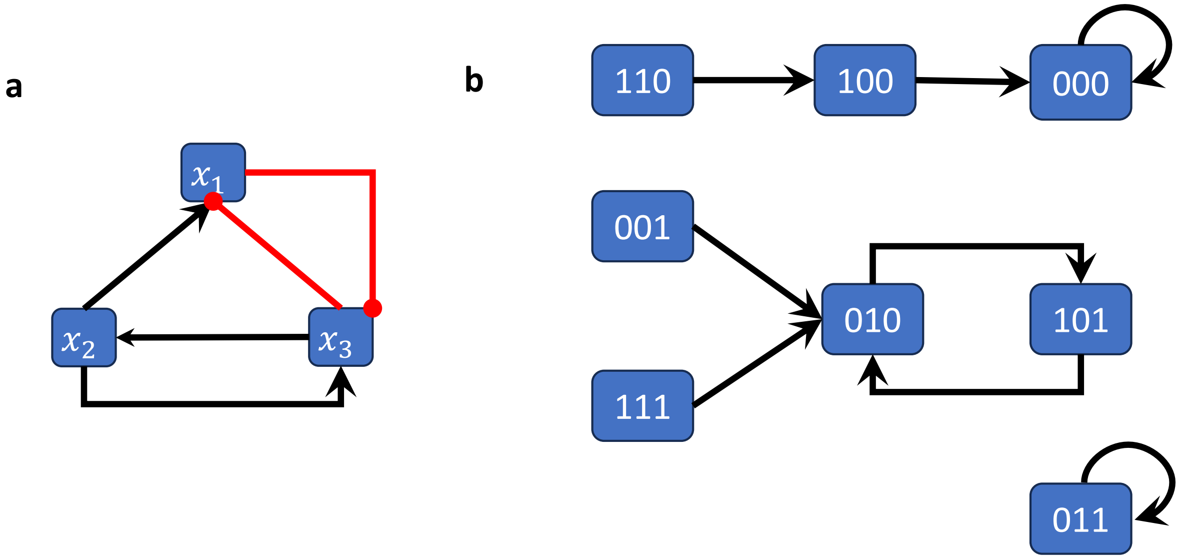


Figure 2: Wiring diagram and state space of the Boolean network in Example 2.4-2.11. (a) The wiring diagram encodes the dependency between variables. (b) The state space is a directed graph with edges between all states and their images. This graph therefore encodes all possible trajectories.

brevity).

$$\begin{aligned} T_1 &= (000, 000, 000, 000, \ldots) \\ T_2 &= (001, 010, 101, 010, \ldots) \\ T_3 &= (010, 101, 010, 101, \ldots) \\ T_4 &= (011, 011, 011, 011, \ldots) \\ T_5 &= (100, 000, 000, 000, \ldots) \\ T_6 &= (101, 010, 101, 010, \ldots) \\ T_7 &= (110, 100, 000, 000, \ldots) \\ T_8 &= (111, 010, 101, 010, \ldots) \end{aligned}$$

We can see that T_3 and T_6 are periodic trajectories with period 2. Similarly, T_1 and T_4 are periodic with period 1. All other trajectories eventually reach one of these 4 states.

When seen as trajectories, T_3 and T_6 are different, but they can both be encoded by the fact that $F(0,1,0) = (1,0,1)$ and $F(1,0,1) = (0,1,0)$. Similarly, T_1 and T_4 can be encoded by the equalities $F(0,1,1) = (0,1,1)$ and $F(0,0,0) = (0,0,0)$. This alternative, more compact way of encoding the dynamics of a Boolean network is the standard approach, which we formalize next.

Definition 2.7. *The* state space *of a (synchronously updated) Boolean network $F : \mathbb{F}_2^n \to \mathbb{F}_2^n$ is a directed graph with vertices in $\mathbb{F}_2^n$ and an edge from x to y if $F(x) = y$.*

Example 2.8. *Figure 2b shows the state space of the (synchronously updated) Boolean network from Example 2.4.*

From the state space, one can easily obtain all periodic points, which form the attractors of the network.

Definition 2.9. *The* space of attractors *for a Boolean network is the set $\mathcal{D}(F)$ of all* minimal *subsets $\mathcal{C} \subseteq \mathbb{F}_2^n$ satisfying $F(\mathcal{C}) = \mathcal{C}$.*

1. *The subset* $\mathcal{D}^1(F) \subset \mathcal{D}(F)$ *of sets of exact size 1 consists of all* steady states *(also known as* fixed points*) of* F.
2. *The subset* $\mathcal{D}^r(F) \subset \mathcal{D}(F)$ *of sets of exact size* r *consists of all* cycles *of length* r *of* F.

Equivalently, an attractor *of length* r *is an ordered set with* r *elements,* $\mathcal{C} = \{c_1, \ldots, c_r\}$*, such that* $F(c_1) = c_2, F(c_2) = c_3, \ldots, F(c_{r-1}) = c_r, F(c_r) = c_1$.

Remark 2.10. *In the case of steady states, the attractor* $\mathcal{C} = \{c\}$ *may be denoted simply by* c.

Example 2.11. *The Boolean network from Example 2.4 has 2 steady states (i.e., attractors of length 1) and one cycle of length 2, which can be easily derived from its state space representation (Figure 2b).*

2.3 Modules

In [25], a concept of modularity was introduced for Boolean networks. The decomposition into modules occurs on structural (wiring diagram) level but induces an analogous decomposition of the network dynamics, in the sense that one can recover the dynamics of the entire network from the dynamics of the modules. For this decomposition, a *module* of a BN is defined as a subnetwork that is itself a BN with external parameters in the subset of variables that specifies a strongly connected component (SCC) in the wiring diagram (see Example 2.12). More precisely, for a Boolean network F and subset of its variables S, we define the *restriction* of F to S to be the BN $F|_S = (f_{k_1}, \cdots, f_{k_m})$, where $x_{k_i} \in S$ for $i = 1, \cdots, m$. We note that f_{k_i} might contain inputs that are not part of S (e.g., when x_{k_i} is regulated by some variables that are not in S). Therefore, the BN $F|_S$ may contain external parameters (which are themselves fixed and do not possess an update rule). Given a F with wiring diagram W, let $W_1, \ldots, W_m$ be the SCCs of W with pairwise disjoint sets of variables S_i. The *modules* of F are then the restrictions to these sets of variables, $F|_{S_i}$. Further, the modular structure of F can be described by a directed acyclic graph $Q = \{(i,j) \mid W_i \longrightarrow W_j\}$ by setting $W_i \longrightarrow W_j$ whenever there exists a node from W_i to W_j (see Example 2.12).

Example 2.12. *Consider the Boolean network*

$$F(x) = (x_2 \wedge x_1, \neg x_1, x_1 \vee \neg x_4, (x_1 \wedge \neg x_2) \vee (x_3 \wedge x_4))$$

with wiring diagram in Figure 3a. The restriction of this network to $S_1 = \{x_1, x_2\}$ *is the 2-variable network* $F|_{S_1}(x_1, x_2) = (x_2 \wedge x_1, \neg x_1)$*, which forms the first module (indicated by the amber box in Figure 3a), while the restriction of* F *to* $S_2 = \{x_3, x_4\}$ *is the 2-variable network with external parameters* x_1 *and* x_2 $F|_{S_2}(x_3, x_4) = (x_1 \vee \neg x_4, (x_1 \wedge \neg x_2) \vee (x_3 \wedge x_4)))$*, which forms the second module (indicated by the green module in Figure 3a). Note that the module* $F|_{S_1}$*, i.e., the restriction of* F *to* S_1*, is simply the projection of* F *onto the variables* S_1 *because* W_1 *does not receive feedback from the other component.*

The wiring diagram of this network has two strongly connected components W_1 *and* W_2 *with variables* $S_1 = \{x_1, x_2\}$ *and* $S_2 = \{x_3, x_4\}$ *(Figure 3a), connected according to the directed acyclic graph* $Q = \{(1,2) \mid W_1 \longrightarrow W_2\}$ *(Figure 3b).*

3 Control via Modularity

In this section, we apply the modular decomposition theory described in the previous section and in [25] to make the control problem of Boolean networks more tractable. We show how the decomposition into modules can be used to obtain controls for each module, which can then be combined to obtain a control for the entire network. In this context, two types of control actions are generally considered: edge controls and node controls. For each type of control, one can consider deletions or constant expressions as defined below. The motivation for considering these control actions is that they represent the common interventions that can be implemented in practice. For instance, edge deletions can be achieved by the use of therapeutic drugs that target specific gene interactions, whereas node deletions represent the blocking of effects of products of genes associated to these nodes; see [31, 32].

Once the modules have been identified, different methods for phenotype control (that is, control of the attractor space) can be used to identify controls in these networks. Some of these methods employ stable motifs [11], feedback vertex sets [13], as well as algebraic approaches [33, 34, 35]. For our examples below, we will use the methods defined in [11, 33, 13] to find controls for the simple networks.

A Boolean network $F = (f_1, \ldots, f_n) : \mathbb{F}_2^n \to \mathbb{F}_2^n$ with *control* is a Boolean network $F^\mu : \mathbb{F}_2^n \times U \to \mathbb{F}_2^n$, where U is a set that denotes all possible controls, defined below. The case of no control coincides with the original Boolean network, that is, $F^\mu(x, 0) = F(x)$. Given a control $\mu \in U$, the dynamics are given by $x(t+1) = F^\mu(x(t), \mu)$. See [33] for additional details and examples of how to encode control edges and nodes in a Boolean network.

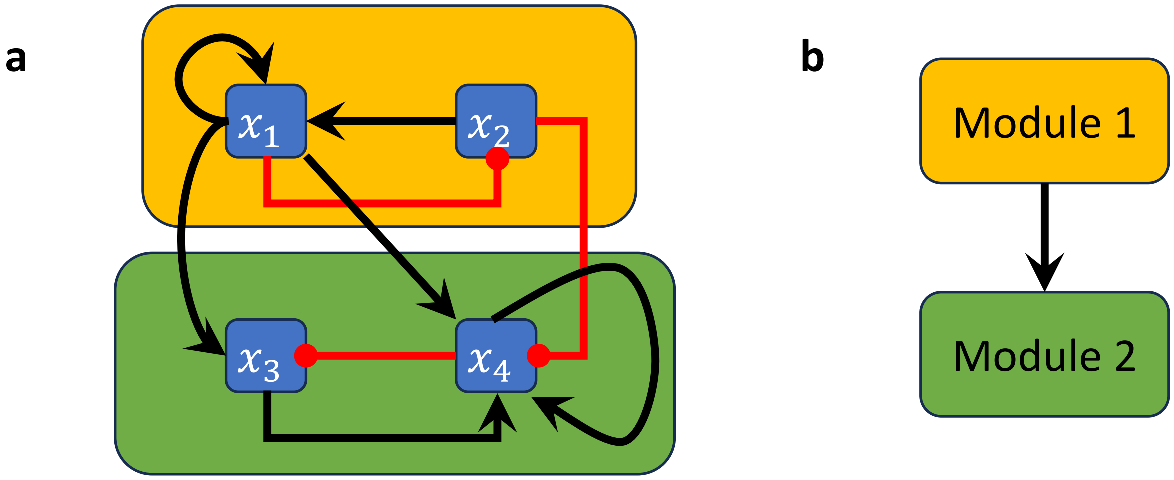


Figure 3: Boolean network decomposition into modules. (a) Wiring diagram of a non-strongly connected Boolean network where the non-trivial modules are highlighted by amber and green boxes. (b) Directed acyclic graph describing the corresponding connections between the nontrivial modules.

Definition 3.1 (Edge Control). *Consider the edge $x_i \to x_j$ in the wiring diagram W. The function*

$$F_j^\mu(x, \mu_{i,j}) := f_j(x_1, \ldots, (\mu_{i,j} + 1)x_i + \mu_{i,j} a_i, \ldots, x_n), \tag{1}$$

where a_i is a constant in $\mathbb{F}_2$, encodes the control of the edge $x_i \to x_j$, since for each possible value of $\mu_{i,j} \in \mathbb{F}_2$ we have the following control settings:

- *If $\mu_{i,j} = 0$, $F_j^\mu(x, 0) = f_j(x_1, \ldots, x_i, \ldots, x_n)$. That is, the control is not active.*
- *If $\mu_{i,j} = 1$, $F_j^\mu(x, 1) = f_j(x_1, \ldots, x_i = a_i, \ldots, x_n)$. In this case, the control is active, and the action represents the removal of the edge $x_i \to x_j$ when $a_i = 0$, and the constant expression of the edge if $a_i = 1$. We use $x_i \xrightarrow{a_i} x_j$ to denote that the control is active.*

This definition can be easily extended for the control of many edges, so that we obtain $F^\mu : \mathbb{F}_2^n \times \mathbb{F}_2^e \to \mathbb{F}_2^n$, where e is the number of edges in the wiring diagram. Each coordinate, $\mu_{i,j}$, of μ in $F^\mu(x, u)$ encodes the control of an edge $x_i \to x_j$.

Definition 3.2 (Node Control). *Consider the node x_i in the wiring diagram W. The function*

$$F_j^\mu(x, u_i^-, u_i^+) := (u_i^- + \mu_i^+ + 1) f_j(x) + \mu_i^+ \tag{2}$$

encodes the control (knock-out or constant expression) of the node x_i, since for each possible value of $(\mu_i^-, \mu_i^+) \in \mathbb{F}_2^2$ we have the following control settings:

- *For $u_i^- = 0, u_i^+ = 0$, $F_j^\mu(x, 0, 0) = f_j(x)$. That is, the control is not active.*
- *For $u_i^- = 1, u_i^+ = 0$, $F_j^\mu(x, 1, 0) = 0$. This action represents the knock-out of the node x_j.*
- *For $u_i^- = 0, u_i^+ = 1$, $F_j^\mu(x, 0, 1) = 1$. This action represents the constant expression of the node x_j.*
- *For $u_i^- = 1, u_i^+ = 1$, $F_j^\mu(x, 1, 1) = f_j(x_{t_1}, \ldots, x_{t_m}) + 1$. This action changes the Boolean function to its negative value. This case is usually not considered in the control search since it is biologically impractical to implement.*

We note that the algebraic framework is versatile enough that we can encode any type of control, such as a combination of node and edge control at the same time.

Definition 3.3. *A control μ* stabilizes *a given Boolean network $F : \mathbb{F}_2^n \to \mathbb{F}_2^n$ at an attractor $\mathcal{C} \subseteq \mathbb{F}_2^n$ when the resulting network after applying μ to F (denoted here as F^μ) has $\mathcal{C}$ as its only attractor.*

For a Boolean network F, we let $\mathcal{D}(F)$ denote the set of its attractors. Whenever a Boolean network F has more than one module we say that it is *decomposable* into its constituent modules $F_1, F_2, \cdots, F_m$ $(m \geq 2)$, and write $F = F_1 \rtimes_{P_1} F_2 \rtimes_{P_2} \cdots \rtimes_{P_{m-1}} F_m$ where the semi-product operation $\rtimes_{P_i}$ indicates the coupling of the subnetworks, as described in [25] (see Example 3.4 for an example of coupling). Furthermore, from the decomposition theory described in [25], the attractors of F are of the form $\mathcal{C} = \mathcal{C}_1 \oplus \mathcal{C}_2 \oplus \cdots \oplus \mathcal{C}_n$ where $\mathcal{C}_i \in \mathcal{D}(F_i)$ is an attractor of the subnetwork, for $i = 1, \ldots, n$. The following theorem takes advantage of the modular structure of the network to find controls one module at a time.

Example 3.4. *Consider the Boolean network in Example 2.12,*

$$F(x) = (x_2 \wedge x_1, \neg x_1, x_1 \vee \neg x_4, (x_1 \wedge \neg x_2) \vee (x_3 \wedge x_4))$$

with wiring diagram in Figure 3a. Let $F_1 = F|_{S_1}(x_1, x_2) = (x_2 \wedge x_1, \neg x_1)$ be the restriction of F to $S_1 = \{x_1, x_2\}$ which forms the first module (indicated by the amber box in Figure 3a) and let $F_2 = F|_{S_2}(x_3, x_4) = (e_1 \vee \neg x_4, (e_1 \wedge \neg e_2) \vee (x_3 \wedge x_4)))$ be the restriction of F to $S_2 = \{x_3, x_4\}$, which forms the second module (indicated by the green box in Figure 3a). Here, e_1 and e_2 are external parameters of $F|_{S_2}$ that take the place of x_1 and x_2, which is indicated by the coupling $P = \{x_1 \to e_1, x_2 \to e_2\}$. F is thus decomposable into F_1 and F_2 and we have $F = F_1 \rtimes_P F_2$.

Theorem 3.5. *Given a decomposable network $F = F_1 \rtimes_P F_2$, if μ_1 is a control that stabilizes F_1 in $\mathcal{C}_1$ (whether $\mathcal{C}_1$ is an existing attractor of F_1 or a new one, after applying control μ_1) and μ_2 is a control that stabilizes F_2 in $\mathcal{C}_2$ (whether $\mathcal{C}_2$ is an existing attractor of F_2 or a new one, after applying control μ_2), then $\mu = (\mu_1, \mu_2)$ is a control that stabilizes F in $\mathcal{C} = \mathcal{C}_1 \oplus \mathcal{C}_2$ provided that at least one of $\mathcal{C}_1$ or $\mathcal{C}_2$ is a steady state.*

Proof. Let $F_1^{\mu_1}$ be the resulting network after applying the control μ_1. Thus, the dynamics of $F_1^{\mu_1}$ is $\mathcal{C}_1$, that is $\mathcal{D}(F_1^{\mu_1}) = \mathcal{C}_1$. Similarly, the dynamics of $F_2^{\mu_2}$ is $\mathcal{C}_2$. Let Let $F_2^{\mathcal{C}_1,\mu_2}$ be the resulting network after evaluating the external parameters of F_2 in the states of $\mathcal{C}_1$ and applying the control μ_2. Then, $\mathcal{D}(F_2^{\mathcal{C}_1,\mu_2}) = \mathcal{C}_2$. Thus,

$$F^\mu = (F_1 \rtimes_P F_2)^\mu = F_1^\mu \rtimes_P F_2^\mu = F_1^{\mu_1} \rtimes_P F_2^{\mu_2}.$$

Therefore,

$$\mathcal{D}(F^\mu) = \mathcal{D}(F_1^{\mu_1} \rtimes_P F_2^{\mu_2}) = \bigsqcup_{\mathcal{C}' \in \mathcal{D}(F_1^{\mu_1})} \mathcal{C}' \oplus \mathcal{D}(F_2^{\mathcal{C}',\mu_2}) = \mathcal{C}_1 \oplus \mathcal{D}(F_2^{\mathcal{C}_1,\mu_2}) = \mathcal{C}_1 \oplus \mathcal{C}_2.$$

For the last equality we used the fact that the product of a steady state and a cycle (or vice versa) will result in only one attractor for the combined network. The former is not always true in general because multiplying two attractors (of length greater than 1) might result in several attractors for the composed network due to the attractors starting at different states.

It follows that there is only one attractor of F^μ and that attractor is $\mathcal{C}_1 \oplus \mathcal{C}_2$. Thus, F is stabilized by $\mu = (\mu_1, \mu_2)$ and we have $\mathcal{D}(F^\mu) = \mathcal{C}$. □

Theorem 3.5 shows how the modular structure can be used to identify controls that stabilize the network in any desired state. In particular, we can use the modular structure of a network to find controls that stabilize a network at an existing attractor, which is often the case in biological control applications. We state this fact in the following corollary.

Corollary 3.6. *Given a decomposable network $F = F_1 \rtimes_P F_2$, let $\mathcal{C} = \mathcal{C}_1 \oplus \mathcal{C}_2$ be an attractor of F, where $\mathcal{C}_1 \in \mathcal{D}(F_1)$ and $\mathcal{C}_2 \in \mathcal{D}(F_2^{\mathcal{C}_1})$ and at least $\mathcal{C}_1$ or $\mathcal{C}_2$ is a steady state. If μ_1 is a control that stabilizes F_1 in $\mathcal{C}_1$ and μ_2 is a control that stabilizes $F_2^{\mathcal{C}_1}$ in $\mathcal{C}_2$, then $\mu = (\mu_1, \mu_2)$ is a control that stabilizes F in $\mathcal{C}$.*

Remark 3.7. *In Theorem 3.5, we required one of the stabilized attractors to be a steady state in order to be able to combine the controls from the modules. We can remove this requirement from Theorem 3.5 by using the following definition of stabilization for non-autonomous networks, which will guarantee that $\mathcal{C}_1$ and $\mathcal{C}_2$ can be combined in a unique way, resulting in a unique attractor of the whole network.*

Definition 3.8. *A* non-autonomous Boolean network *is defined by*

$$y(t+1) = H(g(t), y(t)),$$

where $H : \mathbb{F}_2^{m+n} \to \mathbb{F}_2^n$ and $(g(t))_{t=0}^{\infty}$ is a sequence with elements in $\mathbb{F}_2^m$. We call this type of network non-autonomous because its dynamics will depend on $g(t)$. We use H^g to denote this non-autonomous network.

A state $c \in \mathbb{F}_2^n$ is a steady state *of H^g if $H(g(t), c) = c$ for all t. Similarly, an ordered set with r elements, $\mathcal{C} = \{c_1, \ldots, c_r\}$, is an* attractor of length r *of H^g if $c_2 = H(g(1), c_1)$, $c_3 = H(g(2), c_2), \ldots, c_r = H(g(r-1), c_{r-1})$, $c_1 = H(g(r), c_r)$, $c_2 = H(g(r+1), c_1), \ldots$. Note that in general $g(t)$ is not necessarily of period r and may even not be periodic.*

If $H(g(t), y) = G(y)$ for some network G (that is, it does not depend on $g(t)$) for all t, then $y(t+1) = H(g(t), y(t)) = G(y(t))$ and this definition of attractors coincides with the classical definition of attractors for (autonomous) Boolean networks (Definition 2.9).

Definition 3.9. *Consider a controlled non-autonomous network given by $y(t+1) = F_2(g(t), y(t), \mu)$, where $g(t)$ is a trajectory representation of an attractor $\mathcal{C}_1$ of an upstream network. We say that a control μ_2* stabilizes *this network, $F_2^{\mathcal{C}_1}$ (defined as in Definition 3.8), at an attractor $\mathcal{C}_2$ when the resulting network after applying μ_2 (denoted here as $F_2^{\mathcal{C}_1, \mu_2}$) has $\mathcal{C}_2$ as its unique attractor. For non-autonomous networks the definition of unique attractor requires that $(g(t), y(t))_{t=0}^{\infty}$ has a unique periodic trajectory up to shifting of t (which is automatically satisfied if $\mathcal{C}_1$ or $\mathcal{C}_2$ is a steady state).*

Example 3.10. *Consider the network $F(x_1, x_2, x_3, x_4) = (x_2, x_1, x_2 \wedge x_4, x_3)$, which can be decomposed into $F = F_1 \rtimes F_2$, with $F_1(x_1, x_2) = (x_2, x_1)$ and $F_2(x_3, x_4) = (x_4, x_3)$. Suppose we want to stabilize F in 0110 (which is not an attractor of F). Note that the non-autonomous network $F_2(x_1, x_2, x_3, x_4) = (x_2 \wedge x_4, x_3)$ and $\mathcal{D}(F_1) = \{00, 11, \{01, 10\}\}$.*

- *Consider the control $\mu_1 : (x_1 \xrightarrow{1} x_2, x_2 \xrightarrow{0} x_1)$. That is, the control is the combined action of setting the input from x_1 to x_2 to 1 and the input from x_2 to x_1 to 0. The control μ_1 stabilizes F_1 at* 01, *which is not an original attractor of F_1. Let $\mathcal{C}_1 = \{01\} \in \mathcal{D}(F_1^{\mu_1})$. Note that the space of attractors for $F_2^{\mathcal{C}_1}$ is $\mathcal{D}(F_2^{\mathcal{C}_1}) = \{00, 11, \{01, 10\}\}$.*
- *Now consider the control $\mu_2 : (x_4 \xrightarrow{1} x_3, x_3 \xrightarrow{0} x_4)$. That is, the control is the combined action of setting the input from x_4 to x_3 to 1 and the input from x_3 to x_4 to 0. This control stabilizes $F_2^{\mathcal{C}_1}$ at $\mathcal{C}_2 = \{10\} \in \mathcal{D}(F_2^{\mathcal{C}_1})$, which is not an original attractor of $F_2^{\mathcal{C}_1}$.*
- *Finally, the control $\mu = (\mu_1, \mu_2)$ stabilizes F at $\mathcal{C} = \mathcal{C}_1 \oplus \mathcal{C}_2 = \{0110\}$. Note that $\mathcal{C}$ is a new attractor for F.*

Theorem 3.5 uses the modular structure of a Boolean network to identify controls that stabilize the network in any desired attractor. In biological applications, the attractors typically correspond to distinct biological phenotypes (defined more rigorously in the next section) and a common question is how to force a network to always transition to only one of these phenotypes. For example, cancer biologists may use an appropriate Boolean network model with the two phenotypes proliferation and apoptosis to identify drug targets (i.e., edge or node controls), which force the system to always undergo apoptosis. The following example illustrates this specific control aspect, described in Corollary 3.6.

Example 3.11. *Consider again the network $F(x_1, x_2, x_3, x_4) = (x_2, x_1, x_2 x_4, x_3) = F_1 \rtimes F_2$ from Example 3.10 with $F_1(x_1, x_2) = (x_2, x_1)$ and $F_2(x_3, x_4) = (x_4, x_3)$. Suppose we want to stabilize F in 1111, which is an attractor of F (but not the only one). Note that the non-autonomous network $\overline{F}_2(x_1, x_2, x_3, x_4) = (x_2 x_4, x_3)$ and $\mathcal{D}(F_1) = \{00, 11, \{01, 10\}\}$. Let $\mathcal{C}_1 = \{11\} \in \mathcal{D}(F_1)$.*

- *The edge control $\mu_1 : x_1 \xrightarrow{1} x_2$ (that is, the control that constantly expresses the edge from x_1 to x_2) stabilizes F_1 at $\mathcal{C}_1 = \{11\}$. The space of attractors for $F_2^{\mathcal{C}_1}$ is then $\mathcal{D}(F_2^{\mathcal{C}_1}) = \{00, 11, \{01, 10\}\}$. Note that $x_2 \xrightarrow{1} x_1$ would be an alternative control.*
- *The edge control $\mu_2 : x_4 \xrightarrow{1} x_3$ (that is, the control that constantly expresses the edge from x_4 to x_3) stabilizes $F_2^{\mathcal{C}_1}$ at $\mathcal{C}_2 = \{11\} \in \mathcal{D}(F_2^{\mathcal{C}_1})$. Again, note that $x_3 \xrightarrow{1} x_4$ would be an alternative control.*
- *Now, the control $\mu = (\mu_1, \mu_2) = (x_1 \xrightarrow{1} x_2, x_4 \xrightarrow{1} x_3)$ stabilizes F at $\mathcal{C} = \mathcal{C}_1 \oplus \mathcal{C}_2 = \{1111\}$.*

4 Control via Modularity and Canalization

In addition to using the modular structure of the network, we can take advantage of the canalizing structure of the regulatory functions to identify control targets.

We first review some concepts and definitions, and introduce the concept of *canalization.*

Definition 4.1. *A Boolean function $f(x_1, \dots, x_n)$ is* essential *in the variable x_i if there exists an $\mathbf{x} \in \{0,1\}^n$ such that*

$$f(\mathbf{x}) \neq f(\mathbf{x} \oplus e_i),$$

where e_i is the ith unit vector. In that case, we also say f depends *on x_i.*

Definition 4.2. *A Boolean function $f(x_1, \dots, x_n)$ is* canalizing *if there exists a variable x_i, a Boolean function $g(x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_n)$ and a, $b \in \{0,\ 1\}$ such that*

$$f(x_1, x_2, ..., x_n) = \begin{cases} b, & \text{if } x_i = a \\ g(x_1, x_2, ..., x_{i-1}, x_{i+1}, ..., x_n), & \text{if } x_i \neq a \end{cases}$$

In that case, we say that x_i canalizes *f (to b) and call a the* canalizing input *(of x_i) and b the* canalized output.

Definition 4.3. *A Boolean function $f(x_1, \dots, x_n)$ is* nested canalizing *with respect to the permutation $\sigma \in \mathcal{S}_n$, inputs $a_1, \dots, a_n$ and outputs $b_1, \dots, b_n$, if*

$$f(x_1, \dots, x_n) = \begin{cases} b_1 & x_{\sigma(1)} = a_1, \\ b_2 & x_{\sigma(1)} \neq a_1, x_{\sigma(2)} = a_2, \\ b_3 & x_{\sigma(1)} \neq a_1, x_{\sigma(2)} \neq a_2, x_{\sigma(3)} = a_3, \\ \vdots & \vdots \\ b_n & x_{\sigma(1)} \neq a_1, \dots, x_{\sigma(n-1)} \neq a_{n-1}, x_{\sigma(n)} = a_n, \\ 1 \oplus b_n & x_{\sigma(1)} \neq a_1, \dots, x_{\sigma(n-1)} \neq a_{n-1}, x_{\sigma(n)} \neq a_n. \end{cases}$$

The last line ensures that f actually depends on all n variables.

To account for partial canalization, we also define k-canalizing functions which were first introduced in [36].

Definition 4.4. *A Boolean function $f(x_1, \dots, x_n)$ is* k-canalizing*, where $1 \leq k \leq n$, with respect to the permutation $\sigma \in \mathcal{S}_n$, inputs $a_1, \dots, a_k$, and outputs $b_1, \dots, b_k$ if*

$$f(x_1, \dots, x_n) = \\ \begin{cases} b_1 & x_{\sigma(1)} = a_1, \\ b_2 & x_{\sigma(1)} \neq a_1, x_{\sigma(2)} = a_2, \\ b_3 & x_{\sigma(1)} \neq a_1, x_{\sigma(2)} \neq a_2, x_{\sigma(3)} = a_3, \\ \vdots & \vdots \\ b_k & x_{\sigma(1)} \neq a_1, \dots, x_{\sigma(k-1)} \neq a_{k-1}, x_{\sigma(k)} = a_k, \\ f_C \not\equiv b_k & x_{\sigma(1)} \neq a_1, \dots, x_{\sigma(k-1)} \neq a_{k-1}, x_{\sigma(k)} \neq a_k, \end{cases}$$

where $f_C = f_C(x_{\sigma(k+1)}, \dots, x_{\sigma(n)})$ is the core function*, a Boolean function on $n - k$ variables. When f_C is not canalizing, then the integer k is the* canalizing depth *of f [37]. Note that an n-canalizing function (i.e., a function where all variables become eventually canalizing) is also called a* nested canalizing function *(NCF).*

We restate the following stratification theorem for reference.

Theorem 4.5 ([36]). *Every Boolean function $f(x_1, \dots, x_n) \not\equiv 0$ can be uniquely written as*

$$f(x_1, \dots, x_n) = M_1(M_2(\cdots(M_{r-1}(M_r p_C + 1) + 1)\cdots) + 1) + q, \tag{3}$$

where each $M_i = \prod_{j=1}^{k_i}(x_{i_j} + a_{i_j})$ is a nonconstant extended monomial, p_C is the core polynomial *of f, and $k = \sum_{i=1}^{r} k_i$ is the canalizing depth. Each x_i appears in exactly one of $\{M_1, \dots, M_r, p_C\}$, and the only restrictions are the following "exceptional cases":*

1. *If $p_C \equiv 1$ and $r \neq 1$, then $k_r \geq 2$;*
2. *If $p_C \equiv 1$ and $r = 1$ and $k_1 = 1$, then $q = 0$.*

When f is not canalizing (i.e., when $k = 0$), we simply have $p_C = f$.

Definition 4.6. *Given a Boolean function $f(x_1, \dots, x_n)$ represented as in Equation 3, we call the extended monomials M_i the* layers *of f and define, as in [38], the* layer structure *as the vector $(k_1, \dots, k_r)$, which describes the number of variables in each layer. Note that f is nested canalizing if and only if $k_1 + \cdots + k_r = n$.*

Remark 4.7. *Here we note the following important properties of layers of canalization.*

(a) Theorem 4.5 shows that any Boolean function has a unique extended monomial form given by Equation 3, in which the variables are partitioned into different layers based on their dominance. Any variable that is canalizing (independent of the values of other variables) is in the first layer. Any variable that "becomes" canalizing when excluding all variables from the first layer is in the second layer, etc. All remaining variables that never become canalizing are part of the core polynomial. The number of variables that eventually become canalizing is the canalizing depth of the function. NCFs are exactly those functions where all variables become eventually canalizing (note not all variables of an NCF must be in the first layer).

(b) While variables in the same layer may have different canalizing input values, they all share the same canalized output value, i.e., they all canalize a function to the same output. On the other hand, the outputs of two consecutive layers are distinct. Therefore, the number of layers of a k-canalizing function, expressed as in Definition 4.4, is simply one plus the number of changes in the vector of canalized outputs, $(b_1, b_2, \dots, b_k)$.

Example 4.8. *The Boolean functions $f(x_1, x_2, x_3, x_4) = x_1 \wedge (\neg x_2 \vee (x_3 \wedge x_4))$ and $g(x_1, x_2, x_3, x_4) = x_1 \wedge (\neg x_2 \vee x_3 \vee x_4)$ are nested canalizing. The function f has layer structure $(1, 1, 2)$ because*

- *x_1 canalizes f to 0 if it receives its canalizing input 0.*
- *If this is not the case, $x_2 = 0$ canalizes f to 1.*
- *Finally, $x_3 = 0$ or $x_4 = 0$ (or both) canalizes f to 0.*

On the other hand, g has layer structure $(1, 3)$. As for f, $x_1 = 0$ canlizes g to 0. If this does not happen, any of the following, $x_2 = 0, x_3 = 1, x_4 = 1$, canalizes g to 1.

While finding the layer structure of a Boolean function is an **NP**-hard problem, there exist several algorithmic implementations [39].

In meaningful biological networks, the attractors correspond to phenotypes. Typically a small subset of all Boolean variables is used to define phenotypes.

Definition 4.9. *Given a Boolean network F with attractor space $\mathcal{D}(F)$ and phenotype-defining variables $\mathcal{P} \subset \{x_1, \dots, x_n\}$, we associate the same phenotype to all attractors $\mathcal{C} \in \mathcal{D}(F)$ that are identical in $\mathcal{P}$. The states in $\mathcal{P}$ will be called markers of the phenotype.*

Suppose $F = F_1 \rtimes_P F_2$ is a decomposable network, and that there is a phenotype that depends on variables in F_2 only (that is, all markers of the phenotype are part of F_2), and that we wish to control the phenotype through F_2. The most straightforward approach is to set the variables that the phenotype depends on to the appropriate values that result in the desired phenotype. However, such intervention may not be experimentally possible. Instead, we can exploit the canalizing properties of the functions corresponding to the nodes connecting the modules F_1 and F_2 to identify control targets.

By Theorem 4.5, the variables of any Boolean update function can be ordered by importance/dominance, based on which layer they appear in (with variables in outer layers being more important). Thus, once we control a variable in a certain layer (by setting it to its canalizing input value), any further control of variables in less dominant layers will have no effect on the function (and thus on the network). We state this fact in the following lemma.

Lemma 4.10. *Suppose $F = F_1 \rtimes_P F_2$ is a decomposable network. Suppose further that only one node $x \in F_2$ with update function f_x is regulated by nodes in F_1. If f_x is canalizing with r layers, let $\ell \in \{1, \dots, r\}$ be the lowest (i.e., most important) layer of f_x, which contains nodes from F_1. If all regulators of x from F_1 appear in the core polynomial, we set $\ell = r + 1$. Then, setting $y \notin F_1$ to its canalizing value decouples the systems F_1 and F_2, as long as y appears in a layer $< \ell$.*

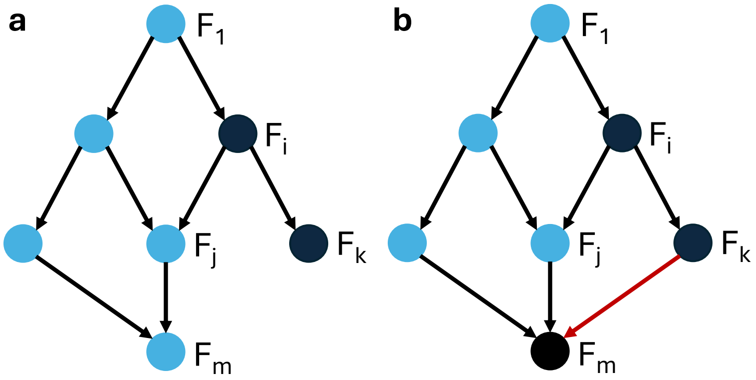


Figure 4: Example of a modular directed acylic graph structure to illustrate Condition (iii) in Theorems 4.11 and 4.12. (a) Module F_i can be removed from the control search as long as conditions (i) and (ii) are satisfied, and the phenotype of interest depends only on any subset of variables that are part of the blue modules. (b) When a node in module F_m is regulated by a node in module F_k (indicated by the red edge in the directed acyclic graph), the phenotype may no longer depend on nodes in F_m, in order for module F_i to be removable from the control search.

Proof. The lemma is a direct consequence of Theorem 4.5. If y receives its canalizing input and is in a more important layer of f_x than all variables in F_1, then none of these variables can affect f_x anymore. Thus, controlling y to receive its canalizing input eliminates the link between F_1 and F_2. □

The modularity approach in [25] yields an acyclic directed graph after one collapses each module into a single node. This endows a natural partial ordering on the collection modules of a network where one module precedes the next if there is a path for every node in the first module which ends in the second module. While not all modules are comparable, we can speak of chains of modules which consist of subsets of the partial ordering which are totally ordered. Furthermore, one can rank the modules based on the percentile scores (i.e., rank module k out of m modules). This type of ranking has been studied in [40], where it was shown that the importance of the modules is strongly correlated with the aggressiveness of mutations occurring within those modules and the effectiveness of interventions.

Theorem 4.11. *Suppose $F = F_1 \rtimes_{P_1} F_2 \rtimes_{P_2} \cdots \rtimes_{P_{n-1}} F_m$ is a decomposable network. If for some $i < j$,*

(i) only one node $x \in F_j$ with update function f_x is regulated by nodes in F_i, and

(ii) f_x is a canalizing function, which possesses none of the variables from F_i in its most important layer, and

(iii) the phenotype of interest only depends on variables in F_j as well as modules F_k, for which any chain containing F_i and F_k also contains F_j. (see Fig. 4 for an example)

then the module F_i can be excluded from the control search by setting any node $y \notin F_i$ to its canalizing input, as long as this node appears in a more dominant layer of f_x than all inputs of f_x that are part of F_i.

Proof. By Lemma 4.10, setting y to its canalizing value results in decoupling F_i and F_j. F_i will no longer have any effect on F_j, and thus, due to condition (iii), on the phenotype of interest. F_i can therefore be removed from the control search. □

Theorem 4.11 is illustrated in Fig. 5. Note that node y can be in F_j or some other module as in the figure. In this theorem, we assumed that none of the variables of F_i are in the most dominant layer in the update rules of variables in F_j. If some of the variables of F_i are in the most dominant layer, we can still remove module F_i from the control search using an edge control, as shown in the following theorem.

Theorem 4.12. *Suppose $F = F_1 \rtimes_{P_1} F_2 \rtimes_{P_2} \cdots \rtimes_{P_{n-1}} F_m$ is a decomposable network. If for some $i < j$,*

(i) only one node $x \in F_j$ with update function f_x is regulated by nodes in F_i, and

(ii) f_x is a canalizing function with some variables from F_i in its most important layer, and

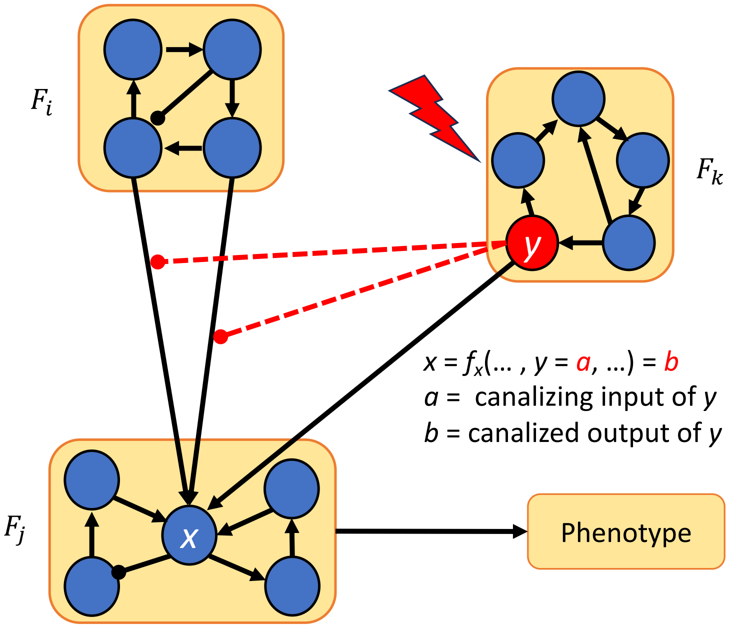

F_i
F_k
y
$x = f_x(\ldots, y = a, \ldots) = b$
a = canalizing input of y
b = canalized output of y
F_j
x
Phenotype

Figure 5: Control via modularity and canalization. Once the network is decomposed into modules $F_1, \ldots, F_m$, we can override the effect of module F_i by the using another module (F_k in this case) whose variables are inputs of f_x that are located in a higher canalizing hierarchy than the layers containing the variables of F_i.

(iii) the phenotype of interest only depends on variables in F_j as well as modules F_k, for which any chain containing F_i and F_k also contains F_j. (see Fig. 4 for an example)

then the module F_i can be excluded from the control search by applying an edge control to any input in the most dominant layer of f_x.

Proof. Let $y \in F_i$ such that $y \in \text{supp}(f_x)$, and that y is located in the most dominant layer f_x. Then, setting y to its canalizing value results in decoupling the subnetworks F_i and F_j. Thus, F_i will no longer have any effect on F_j and thus it can be removed from the control search. □

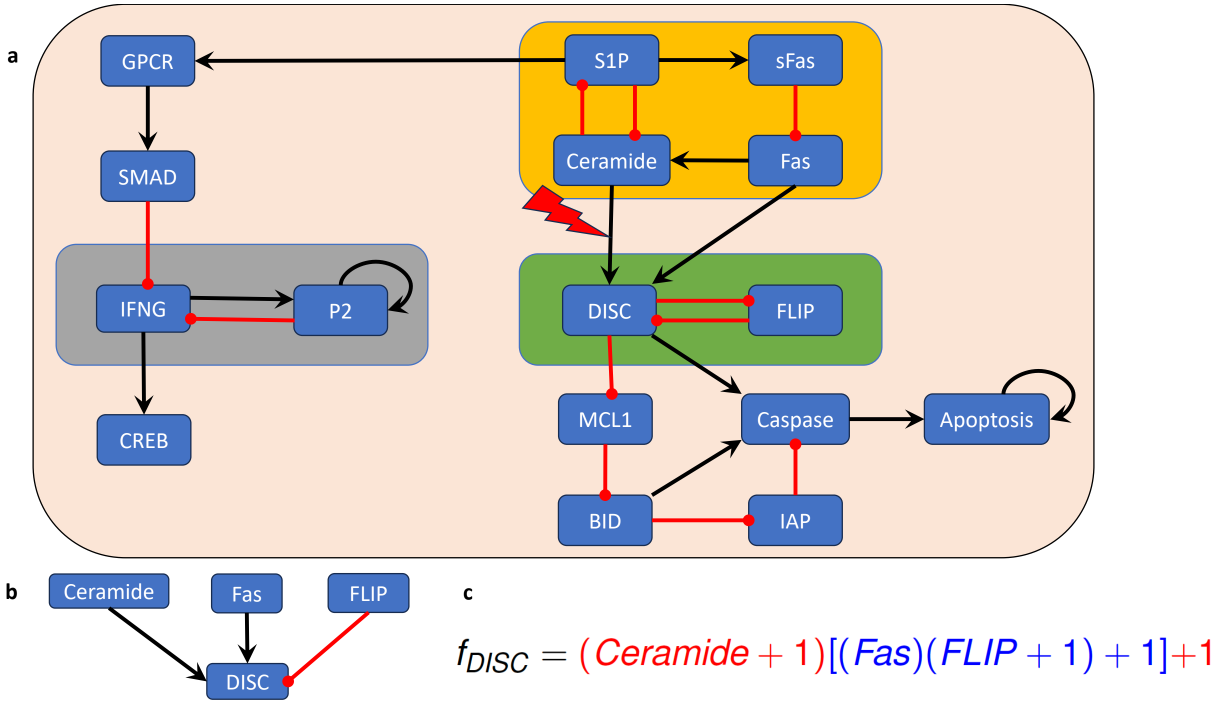
Remark 4.13. *The method can be extended to the case when F_i and F_j are connected via multiple nodes (that is, condition (i) in Theorem 4.11 and Theorem 4.12 can be relaxed. In that case decoupling is achieved through the same procedure presented above, applied to each node in F_j with regulators from F_i.*

We further note that condition (ii) in the theorems above is generally very restrictive as only a small proportion of Boolean functions in $n > 3$ variables is canalizing or even nested canalizing. However, as shown in [26], most biological Boolean network models are almost entirely governed by nested canalizing functions.

To showcase these methods, we will now decompose a published Boolean network model into its modules, and then identify the minimal set of controls for the entire network by exploiting the canalizing structure of the regulatory functions within the modules. The identified set of controls will force the entire system into a desired attractor.

Example 4.14. *We consider a Boolean network model for the blood cancer large granular lymphocyte (T-LGL) leukemia, which was published in [27]. T-LGL leukemia is a clonal hematological disorder characterized by persistent increases of large granular lymphocytes in the absence of reactive cause [41]. The wiring diagram of this model is depicted in Figure 6a. This network has 16 nodes and three nontrivial modules (highlighted by the amber, green, and gray boxes in Figure 6a). The control objective here is to identify control targets that lead the system to programmed cell death. In other words, we aim to direct the system into an attractor that has the marker apoptosis ON.*

Since the model has three nontrivial modules, the approach described in Section 3 would require us to identify control targets for three modules. However, an exploitation of the canalizing structure and common sense reveals



$$f_{DISC} = (Ceramide + 1)[(Fas)(FLIP + 1) + 1] + 1$$

Figure 6: (a) Wiring diagram of the T-LGL model, published in [27], which describes the mechanisms that regulate apoptosis. The non-trivial modules are highlighted by amber, green, and gray boxes. (b) The regulatory inputs of the node DISC. (c) Writing the regulatory function corresponding to node DISC in its standard monomial form (Theorem 4.5) reveals its canalizing structure.

that we do not need to control every module to ensure apoptosis, the desired control objective. First, irrespective of canalization, the module highlighted in gray in Figure 6a does not affect the phenotype apoptosis. Therefore, we can focus on the modules "upstream" of apoptosis (i.e., the amber and green modules in Figure 6a).

In this case, we will apply Theorem 4.12 to identify control targets for this model. We note that the edges from the upstream module (amber box in Figure 6a) to the downstream module (green box in Figure 6a) all end in the node DISC. Therefore, we will investigate the canalizing properties of the regulatory function of DISC (see Figure 6b),

$$f_{DISC} = Ceramide \vee (Fas \wedge \overline{FLIP}).$$

Using the approach described in [39], we find that f_{DISC} has two canalizing layers, $L_1 = \{Ceramide\}$ and $L_2 = \{Fas, FLIP\}$ and its layering structure is given by (see Figure 6c)

$$f_{DISC} = (Ceramide + 1)[(Fas)(FLIP + 1) + 1]{+}1$$

We note that the only variable in the most important canalizing layer, $Ceramide$, is in the upstream module. Thus, we can decouple the modules via an edge control on the connection between the upstream and downstream modules. That is, the constant expression of the edge from Ceramide to DISC will decouple the two modules and will lead to constant expression of DISC. We can check that this control is effective at stabilizing the system in the desired attractor and the control set obviously has minimal size.

In summary, in this example we used an edge control to decouple the upstream and downstream modules and then identified a control target in the downstream module which contains the markers of the phenotype of interest.

5 Conclusion

Model-based control is a mainstay of industrial engineering, and there is a well-developed mathematical theory of optimal control that can be applied to models consisting of systems of ordinary differential equations. While this model type is also commonly used in biology, for instance in biochemical network modeling or epidemiology and ecology, there are many biological systems that are more suitably modeled in other ways. Boolean network models provide a way to encode regulatory rules in networks that can be used to capture qualitative properties of biological networks, when it is unfeasible or unnecessary to determine kinetic information. While they are intuitive to build, they have the drawback that there is very little mathematical theory available that can be used for model analysis, beyond simulation approaches. And for large networks, simulation quickly becomes ineffective.

The results in this paper, building on those in [25], can be considered as a contribution to a mathematical control theory for Boolean networks, incorporating key features of biological networks. There are many open problems that remain, and we hope that this work will inspire additional developments.

Our concrete contributions here are as follows. The modularization method makes the control search far more efficient and allows us to combine controls at the module level obtained with different control methods. For example, methods based on computational algebra [33, 35] can identify controllers that can create new (desired) steady states, which other methods cannot. Feedback vertex set [14, 13] is a structure-based method that identifies a subset of nodes whose removal makes the graph acyclic. Stable motifs [11] are based on identifying strongly connected subgraphs in the extended graph representation of the Boolean network. Other control methods include [42, 43, 44]. We can use any combination of these methods to identify the controls in each module.

6 Acknowledgments

Author Matthew Wheeler was supported by The American Association of Immunologists through an Intersect Fellowship for Computational Scientists and Immunologists. This work was further supported by the Simons foundation [grant numbers 712537 (to C.K.), 850896 (to D.M.), 516088 (to A.V.)]; the American Mathematical Society and the Simons Foundation [Enhancement Grant for PUI faculty (to A.V.)]; the National Institute of Health [grant number 1 R01 HL169974-01 (to R.L.)]; and the Defense Advanced Research Projects Agency [grant number HR00112220038 (to R.L.)]. The authors also thank the Banff International Research Station for support through its Focused Research Group program during the week of May 29, 2022 (22frg001), which was of great help in framing initial ideas of this paper.

References

[1] Boris Aguilar, David L Gibbs, David J Reiss, Mark McConnell, Samuel A Danziger, Andrew Dervan, Matthew Trotter, Douglas Bassett, Robert Hershberg, Alexander V Ratushny, et al. A generalizable data-driven multicellular model of pancreatic ductal adenocarcinoma. *GigaScience*, 9(7):giaa075, 2020.

[2] Daniel Plaugher and David Murrugarra. Modeling the pancreatic cancer microenvironment in search of control targets. *Bulletin of Mathematical Biology*, 83(11):1–26, 2021.

[3] Jordan Rozum and Réka Albert. Leveraging network structure in nonlinear control. *NPJ systems biology and applications*, 8(1):36, 2022.

[4] Daniel Plaugher and David Murrugarra. Phenotype control techniques for boolean gene regulatory networks. *Bulletin of Mathematical Biology*, 85(10):1–36, 2023.

[5] Julian D Schwab, Silke D Kühlwein, Nensi Ikonomi, Michael Kühl, and Hans A Kestler. Concepts in boolean network modeling: What do they all mean? *Computational and structural biotechnology journal*, 18:571–582, 2020.

[6] Alan Veliz-Cuba, Joseph Arthur, Laura Hochstetler, Victoria Klomps, and Erikka Korpi. On the relationship of steady states of continuous and discrete models arising from biology. *Bulletin of mathematical biology*, 74:2779–2792, 2012.

[7] Vidisha Singh, Aurelien Naldi, Sylvain Soliman, and Anna Niarakis. A large-scale boolean model of the rheumatoid arthritis fibroblast-like synoviocytes predicts drug synergies in the arthritic joint. *NPJ systems biology and applications*, 9, 2023.

[8] Sara Sadat Aghamiri, Vidisha Singh, Aurélien Naldi, Tomáš Helikar, Sylvain Soliman, and Anna Niarakis. Automated inference of boolean models from molecular interaction maps using casq. *Bioinformatics*, 36(16):4473–4482, 2020.

[9] A. Veliz-Cuba. Reduction of Boolean network models. *Journal of Theoretical Biology*, 289:167–172, 2011.

[10] Assieh Saadatpour, Réka Albert, and Timothy Reluga. A reduction method for boolean network models proven to conserve attractors. *SIAM Journal on Applied Dynamical Systems*, 12:1997–2011, 01 2013.

[11] Jorge GT Zanudo and Réka Albert. Cell fate reprogramming by control of intracellular network dynamics. *PLoS computational biology*, 11(4):e1004193, 2015.

[12] Jordan C Rozum, Dávid Deritei, Kyu Hyong Park, Jorge Gómez Tejeda Zañudo, and Réka Albert. pystablemotifs: Python library for attractor identification and control in boolean networks. *Bioinformatics*, 38(5):1465–1466, 2022.

[13] Jorge Gomez Tejeda Zañudo, Gang Yang, and Réka Albert. Structure-based control of complex networks with nonlinear dynamics. *Proceedings of the National Academy of Sciences*, 114(28):7234–7239, 2017.

[14] Atsushi Mochizuki, Bernold Fiedler, Gen Kurosawa, and Daisuke Saito. Dynamics and control at feedback vertex sets. ii: A faithful monitor to determine the diversity of molecular activities in regulatory networks. *Journal of theoretical biology*, 335:130–146, 2013.

[15] Laura Cifuentes-Fontanals, Elisa Tonello, and Heike Siebert. Control in boolean networks with model checking. *Frontiers in Applied Mathematics and Statistics*, 8:838546, 2022.

[16] Hannes Klarner and Heike Siebert. Approximating attractors of boolean networks by iterative ctl model checking. *Frontiers in bioengineering and biotechnology*, 3:130, 2015.

[17] Roland Kaminski, Torsten Schaub, Anne Siegel, and Santiago Videla. Minimal intervention strategies in logical signaling networks with asp. *Theory and Practice of Logic Programming*, 13(4-5):675–690, 2013.

[18] Regina Samaga, Axel Von Kamp, and Steffen Klamt. Computing combinatorial intervention strategies and failure modes in signaling networks. *Journal of Computational Biology*, 17(1):39–53, 2010.

[19] Gang Wu, Lisha Zhu, Jennifer E Dent, and Christine Nardini. A comprehensive molecular interaction map for rheumatoid arthritis. *PLoS One*, 5(4):e10137, 2010.

[20] Abdul Salam Jarrah, Reinhard Laubenbacher, and Alan Veliz-Cuba. The dynamics of conjunctive and disjunctive boolean network models. *Bulletin of mathematical biology*, 72:1425–1447, 2010.

[21] Nadav Kashtan and Uri Alon. Spontaneous evolution of modularity and network motifs. *Proceedings of the National Academy of Sciences*, 102(39):13773–13778, 2005.

[22] Leland H Hartwell, John J Hopfield, Stanislas Leibler, and Andrew W Murray. From molecular to modular cell biology. *Nature*, 402(Suppl 6761):C47–C52, 1999.

[23] Hiroaki Kitano. Biological robustness. *Nature Reviews Genetics*, 5(11):826–837, 2004.

[24] Dirk M Lorenz, Alice Jeng, and Michael W Deem. The emergence of modularity in biological systems. *Physics of life reviews*, 8(2):129–160, 2011.

[25] Claus Kadelka, Matthew Wheeler, Alan Veliz-Cuba, David Murrugarra, and Reinhard Laubenbacher. Modularity of biological systems: a link between structure and function. *Journal of the Royal Society Interface*, 20(207):20230505, 2023.

[26] Claus Kadelka, Taras-Michael Butrie, Evan Hilton, Jack Kinseth, Addison Schmidt, and Haris Serdarevic. A meta-analysis of boolean network models reveals design principles of gene regulatory networks. *Science Advances*, 10(2):eadj0822, 2024.

[27] Assieh Saadatpour, Rui-Sheng Wang, Aijun Liao, Xin Liu, Thomas P Loughran, István Albert, and Réka Albert. Dynamical and structural analysis of a t cell survival network identifies novel candidate therapeutic targets for large granular lymphocyte leukemia. *PLoS Comput Biol*, 7(11):e1002267, Nov 2011.

[28] Conrad Hal Waddington. *The strategy of the genes*. Routledge, 2014.

[29] S A Kauffman. Metabolic stability and epigenesis in randomly constructed genetic nets. *J Theor Biol*, 22(3):437–67, Mar 1969.

[30] Daniel Plaugher, Boris Aguilar, and David Murrugarra. Uncovering potential interventions for pancreatic cancer patients via mathematical modeling. *Journal of theoretical biology*, 548:111197, 2022.

[31] Minsoo Choi, Jue Shi, Sung Hoon Jung, Xi Chen, and Kwang-Hyun Cho. Attractor landscape analysis reveals feedback loops in the p53 network that control the cellular response to dna damage. *Science signaling*, 5(251):ra83–ra83, 2012.

[32] David J Wooten, Jorge Gómez Tejeda Zañudo, David Murrugarra, Austin M Perry, Anna Dongari-Bagtzoglou, Reinhard Laubenbacher, Clarissa J Nobile, and Réka Albert. Mathematical modeling of the candida albicans yeast to hyphal transition reveals novel control strategies. *PLoS computational biology*, 17(3):e1008690, 2021.

[33] David Murrugarra, Alan Veliz-Cuba, Boris Aguilar, and Reinhard Laubenbacher. Identification of control targets in boolean molecular network models via computational algebra. *BMC systems biology*, 10:1–11, 2016.

[34] D Murrugarra and ES Dimitrova. Molecular network control through boolean canalization. eurasip j. *Bioinformatics Syst. Biol*, 9, 2015.

[35] Luis Sordo Vieira, Reinhard C Laubenbacher, and David Murrugarra. Control of intracellular molecular networks using algebraic methods. *Bulletin of mathematical biology*, 82(1):2, 2020.

[36] Qijun He and Matthew Macauley. Stratification and enumeration of boolean functions by canalizing depth. *Physica D: Nonlinear Phenomena*, 314:1–8, 2016.

[37] Lori Layne, Elena S Dimitrova, and Matthew Macauley. Nested canalyzing depth and network stability. *Bulletin of Mathematical Biology*, 74(2):422–433, 2012.

[38] Claus Kadelka, Jack Kuipers, and Reinhard Laubenbacher. The influence of canalization on the robustness of boolean networks. *Physica D: Nonlinear Phenomena*, 353:39–47, 2017.

[39] Elena Dimitrova, Brandilyn Stigler, Claus Kadelka, and David Murrugarra. Revealing the canalizing structure of boolean functions: Algorithms and applications. *Automatica*, 146:110630, 2022.

[40] Daniel Plaugher and David Murrugarra. Pancreatic cancer mutationscape: revealing the link between modular restructuring and intervention efficacy amidst common mutations. *bioRxiv*, 2024.

[41] Anila Rashid, Mohammad Khurshid, and Arsalan Ahmed. T-cell large granular lymphocytic leukemia: 4 cases. *Blood research*, 49(3):203–205, 2014.

[42] Sang-Mok Choo, Byunghyun Ban, Jae Il Joo, and Kwang-Hyun Cho. The phenotype control kernel of a biomolecular regulatory network. *BMC systems biology*, 12(1):1–15, 2018.

[43] Laura Cifuentes-Fontanals, Elisa Tonello, and Heike Siebert. Control in boolean networks with model checking. *Frontiers in Applied Mathematics and Statistics*, 8, 2022.

[44] Enrico Borriello and Bryan C Daniels. The basis of easy controllability in boolean networks. *Nature communications*, 12(1):1–15, 2021.