A Formal Framework for Composing Qualitative Models of Biological Systems

Hanadi Alkhudhayr[1,2] and Jason Steggles[1(✉)]

[1] School of Computing, Newcastle University, Newcastle upon Tyne, UK
jason.steggles@ncl.ac.uk
[2] Faculty of Computing and Information Technology, King Abdulaziz University, Rabigh, Saudi Arabia
h.alkhudhayr1@ncl.ac.uk

Abstract. Boolean networks are a widely used qualitative modelling approach which allows the abstract description of a biological system. One issue with the application of Boolean networks is the state space explosion problem which limits the applicability of the approach to large realistic systems. In this paper we investigate developing a compositional framework for Boolean networks to facilitate the construction and analysis of large scale models. The compositional approach we present is based on merging entities between Boolean networks using conjunction and we introduce the notion of compatibility which formalises the preservation of behaviour under composition. We investigate characterising compatibility and develop a notion of trace alignment which is sufficient to ensure compatibility. The compositional framework developed is supported by a prototype tool that automates composition and analysis.

Keywords: Qualitative models · Boolean network · Model composition

1 Introduction

In order to study and synthesize complex biological systems a range of qualitative modelling techniques have emerged [3,4]. *Boolean networks* [8,9] are one such approach which are based on abstractly representing the state of a regulatory entity as a Boolean value, where 1 represents the entity is active and 0 inactive. The state of each entity is then regulated by other entities based on a defined next–state function and their dynamic behaviour results in attractor cycles that can then be associated with biological phenomena. Entities can either be updated *synchronously*, where the state of all entities is updated simultaneously, or *asynchronously*, where entities update their state independently.

Despite their simplicity, Boolean networks have been shown to allow a range of interesting biological analysis to be performed and have been widely considered in the literature (for example, see [1,3,10,11,13]). Indeed, it can be seen that they have an important role to play in advancing our understanding and engineering capability of complex biological systems. However, one important

C. Martín-Vide et al. (Eds.): TPNC 2017, LNCS 10687, pp. 25–36, 2017.
https://doi.org/10.1007/978-3-319-71069-3_2

issue that limits the scalable application of Boolean networks is the well–known state space explosion problem.

In this paper we investigate developing a formal framework for the composition of Boolean networks to facilitate the construction and analysis of large scale models. The compositional approach we present is based on merging entities in Boolean networks using conjunction (though the results presented hold for other logical connectives). We introduce the notion of compatibility which formalises the idea of preserving the underlying behaviour of models that are composed. The compatibility property is problematic as it references the composed model and so we develop a notion of trace alignment which we show is sufficient to ensure compatibility. We illustrate the alignment property by presenting results about the compatibility of composing duplicate copies of a Boolean network. The compositional framework developed is supported by a prototype tool that automates the composition process and associated analysis.

This paper is organized as follows. In Sect. 2 we provide a brief introduction to Boolean networks. In Sect. 3 we develop a compositional framework for Boolean networks and consider the preservation of behaviour under composition which we formalise by a notion of compatibility. In Sect. 4 we investigate characterising compatibility and introduce the property of alignment which avoids directly considering the composed model. Finally, in Sect. 5 we present some concluding remarks and discuss future work.

2 Boolean Networks

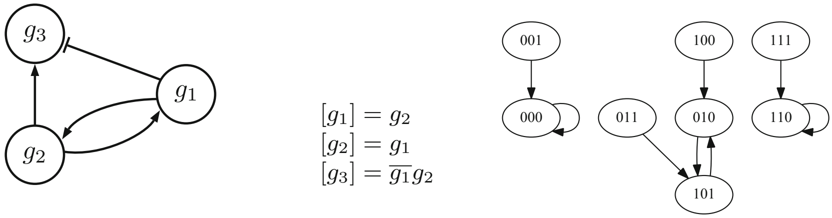
Boolean networks [8,9] are a widely used qualitative modelling approach for biological control systems (see for example [1,3,10,11,13]). In this section we introduce the basic definitions for Boolean networks needed in the sequel and provide illustrative examples.

A Boolean network consists of a set of regulatory entities $G = \{g_1, \ldots, g_n\}$ which can be in one of two possible states, either 1 representing the entity is active (e.g. a gene is expressed or a protein is present) or 0 representing the entity is inactive (e.g. a gene is not expressed or a protein is absent). The state of each entity is regulated by a subset of entities in the Boolean network and we refer to this subset as the *neighbourhood* of an entity (an entity may or may not be in its own neighbourhood). An entity updates its state by applying a logical *next–state function* to the current states of the entities in its neighbourhood.

We can define a Boolean network more formally as follows.

Definition 1. *A* B*oolean Network $\mathcal{BN}$ is a tuple $\mathcal{BN} = (G, N, F)$ where:*

(i) $G = \{g_1, \ldots, g_k\}$ *is a non-empty, finite set of entities;*
(ii) $N = (N(g_1), \ldots, N(g_k))$ *is a tuple of neighbourhoods, such that* $N(g_i) \subseteq G$ *is the neighbourhood of* g_i*; and*
(iii) $F = (F(g_1), \ldots, F(g_n))$ *is a tuple of next-state functions, such that the function* $F(g_i) : \mathbb{B}^{|N(g_i)|} \to \mathbb{B}$ *defines the next state of* g_i*.*



(A) (B) (C)

Fig. 1. Example of a Boolean network $\mathcal{BN}_{Ex1}$ consisting of: (A) Wiring diagram; (B) Equational definition of next–state functions for $\mathcal{BN}_{Ex1}$; (C) Synchronous state graph

As an example, consider the Boolean network $\mathcal{BN}_{Ex1} = (G_{Ex1}, N_{Ex1}, F_{Ex1})$ defined in Fig. 1. It consists of three entities $G_{Ex1} = \{g_1, g_2, g_3\}$ with neighbourhoods $N_{Ex1}(g_1) = \{g_2\}$, $N_{Ex1}(g_2) = \{g_1\}$, and $N_{Ex1}(g_3) = \{g_1, g_2\}$. The next-state functions F_{Ex1} are defined equational in Fig. 1(B), where we use $[g_i]$ to represent the next state of an entity g_i.

A *global state* of a Boolean network $\mathcal{BN}$ with n entities is represented by a tuple of Boolean states $(s_1, \ldots, s_n)$, where $s_i \in \mathbb{B}$ represents the state of entity $g_i \in \mathcal{BN}$. Note as a notational convenience we often use $s_1 \ldots s_n$ to represent a global state $(s_1, \ldots, s_n)$. When the current state of a Boolean network is clear from the context we allow g_i to denote both the name of an entity and its corresponding current state. The state space of a Boolean network $\mathcal{BN}$, denoted $S_{\mathcal{BN}}$, is therefore the set of all possible global states $S_{\mathcal{BN}} = \mathbb{B}^{|G|}$.

The state of a Boolean network can be updated either *synchronously* [9,16], where the state of all entities is updated simultaneously in a single update step, or *asynchronously* [6], where entities update their state independently. In the following we focus on the synchronous update semantics which has received considerable attention in the literature (see for example [1,2,8,9,12,16]). Given two states $S_1, S_2 \in S_{\mathcal{BN}}$, let $S_1 \rightarrow S_2$ represent a *(synchronous) update step* such that S_2 is the state that results from simultaneously updating the state of each entity g_i using its associated update function $F(g_i)$ and the appropriate neighbourhood of states from S_1. As an example, consider the global state 011 for $\mathcal{BN}_{Ex1}$ (see Fig. 1), where entity $g_1 = 0$, $g_2 = 1$, and $g_3 = 1$. Then $011 \rightarrow 101$ is an update step in $\mathcal{BN}_{Ex1}$.

The sequence of global states through $S_{\mathcal{BN}}$ from some initial state is called a *trace*. Note that in the case of the synchronous update semantics such traces are deterministic and infinite. However, given that the global state space is finite, this implies that a trace must eventually enter a cycle, known formally as an *attractor cycle* [9,14]. Attractor cycles are very important biologically where they are seen as representing different biological states or functions (e.g. different cellular types such as proliferation, apoptosis and differentiation [7]). We define a finite canonical representation for synchronous traces $\sigma(S)$, for $S \in S_{\mathcal{BN}}$, which specifies the infinite behaviour of a trace up to the first repeated state. The set of all traces $Tr(\mathcal{BN}) = \{\sigma(S) \mid S \in S_{\mathcal{BN}}\}$ therefore completely characterizes the behaviour of a Boolean network $\mathcal{BN}$ under the (synchronous) update semantics. For example, in $\mathcal{BN}_{Ex1}$ the trace $\sigma(011) = \langle 011, 101, 010, 101, 010, 101, \ldots \rangle$ is denoted by

$$\sigma(011) = \langle 011, 101, 010, 101 \rangle$$

It can be seen that $\mathcal{BN}_{Ex1}$ has three attractors: two point attractors $\langle 000, 000 \rangle$ and $\langle 110, 110 \rangle$; and a cyclic attractor $\langle 101, 010, 101 \rangle$.

The behaviour of a Boolean network can be concisely represented by a *state graph* in which the nodes are the global states and the edges are precisely the synchronous update steps allowed. We let $SG(\mathcal{BN}) = (S_{\mathcal{BN}}, \rightarrow)$ denote the state graph for a Boolean network $\mathcal{BN}$ under the synchronous trace semantics. As an example, consider the synchronous state graph $SG(\mathcal{BN}_{Ex1})$ for $\mathcal{BN}_{Ex1}$ presented in Fig. 1(C).

3 Compositional Framework

In this section we introduce definitions for composing two Boolean networks by merging entities and prove some simple results such as commutativity. We then consider what it means for the behaviour of an individual Boolean network in a composed model to be preserved and formulate a notion of *compatibility*.

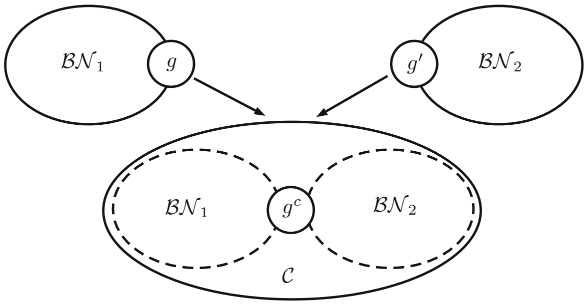


Fig. 2. Pictorial representation of composing $\mathcal{BN}_1$ and $\mathcal{BN}_2$ to form a new Boolean network $\mathcal{C}$ by merging entities $g \in \mathcal{BN}_1$ and $g' \in \mathcal{BN}_2$ into a new entity g^c

In the sequel, let $\mathcal{BN}_1 = (G_1, N_1, F_1)$ and $\mathcal{BN}_2 = (G_2, N_2, F_2)$ be two Boolean networks such that $G_1 = \{g, g_1, \ldots, g_n\}$ and $G_2 = \{g', g'_1, \ldots, g'_m\}$ are disjoint sets, for some $n, m \in \mathbb{N}$.

We formally define the composition of two Boolean networks $\mathcal{BN}_1$ and $\mathcal{BN}_2$ based on using conjunction (see Fig. 2). (Note all results presented also hold using disjunction.)

Definition 2. *(Composition) Let $\mathcal{C}\,(\mathcal{BN}_1, \mathcal{BN}_2, g, g')$ denote the Boolean network constructed by merging $\mathcal{BN}_1$ and $\mathcal{BN}_2$ on entities g and g' defined as follows:*

1. ***Entities:*** *the finite set of entities* $G = (G_1/\{g\}) \cup (G_2/\{g'\}) \cup \{g^c\}$, *where* g^c *denotes the new entity created by merging* g *and* g'.
2. ***Neighbourhood:*** *for any entity* $h_i \in G$, *the neighbourhood* $N(h_i)$ *is defined as follows:*

$$N(h_i) = \begin{cases} N_1(h_i)[g/g^c], & if\ h_i \in G_1 \\ N_2(h_i)[g'/g^c], & if\ h_i \in G_2 \\ N_1(g)[g/g^c] \cup N_2(g')[g'/g^c], & if\ h_i = g^c \end{cases}$$

where $S[f/e]$ *represents set* S *with all occurrences of element* f *replaced by* e.

3. ***Functions:*** *for any* $h_i \in G$, *the next-state function* $F(h_i)$ *is defined:*

$$F(h_i) = \begin{cases} F_1(h_i), & if\ h_i \in G_1 \\ F_2(h_i), & if\ h_i \in G_2 \\ \mathcal{F}, & if\ h_i = g^c \end{cases}$$

where $\mathcal{F} : \mathbb{B}^{|N(g^c)|} \to \mathbb{B}$ *is defined using four cases as follows:*

(i) If $g \notin N_1(g)$ *and* $g' \notin N_2(g')$, *where* $N_1(g) = \{l_1, ..., l_p\}$ *and* $N_2(g') = \{l'_1, ..., l'_q\}$, *then* $\mathcal{F}(l_1, ..., l_p, l'_1, ..., l'_q) = F_1(g)(l_1, ..., l_p) \wedge F_2(g')(l'_1, ..., l'_q)$;

(ii) If $g \in N_1(g)$ *and* $g' \notin N_2(g')$, *where* $N_1(g) = \{g, l_1, ..., l_p\}$ *and* $N_2(g') = \{l'_1, ..., l'_q\}$, *then* $\mathcal{F}(g^c, l_1, ..., l_p, l'_1, ..., l'_q) = F_1(g)(g^c, l_1, ..., l_p) \wedge F_2(g')(l'_1, ..., l'_q)$;

(iii) If $g \notin N_1(g)$ *and* $g' \in N_2(g')$, *where* $N_1(g) = \{l_1, ..., l_p\}$ *and* $N_2(g') = \{g', l'_1, ..., l'_q\}$, *then* $\mathcal{F}(g^c, l_1, ..., l_p, l'_1, ..., l'_q) = F_1(g)(l_1, ..., l_p) \wedge F_2(g')(g^c, l'_1, ..., l'_q)$;

(iv) If $g \in N_1(g)$ *and* $g' \in N_2(g')$, *where* $N_1(g) = \{g, l_1, ..., l_p\}$ *and* $N_2(g') = \{g', l'_1, ..., l'_q\}$, *then*

$$\mathcal{F}(g^c, l_1, ..., l_p, l'_1, ..., l'_q) = F_1(g)(g^c, l_1, ..., l_p) \wedge F_2(g')(g^c, l'_1, ..., l'_q).$$

In the sequel, we let g^c denote the new entity created by merging g and g' and assume that $\mathcal{C}\,(\mathcal{BN}_1, \mathcal{BN}_2, g, g')$ has global states $(g^c\ g_1\ ...\ g_n\ g'_1\ ...\ g'_m) \in \mathcal{S}_{\mathcal{C}}$.

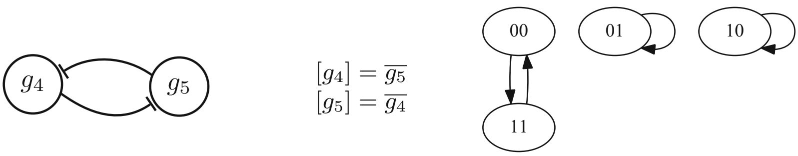


Fig. 3. A second Boolean network example $\mathcal{BN}_{Ex2}$ containing the wiring diagram, next–state equations, and state graph

As an example, consider composing $\mathcal{BN}_{Ex1}$ (Fig. 1) and $\mathcal{BN}_{Ex2}$ (Fig. 3) on entities g_1 and g_4. The resulting Boolean network $\mathcal{C}(\mathcal{BN}_{Ex1}, \mathcal{BN}_{Ex2}, g_1, g_4)$ is depicted in Fig. 4.

The following results shows that composition is commutative.

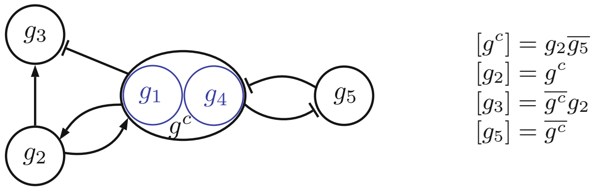


Fig. 4. Boolean network $\mathcal{C}(\mathcal{BN}_{Ex1}, \mathcal{BN}_{Ex2}, g_1, g_4)$ resulting from the composition of $\mathcal{BN}_{Ex1}$ and $\mathcal{BN}_{Ex2}$ on entities g_1 and g_4

Lemma 3. *For any Boolean networks $\mathcal{BN}_1$ and $\mathcal{BN}_2$ and entities $g \in \mathcal{BN}_1$ and $g' \in \mathcal{BN}_2$ we have $\mathcal{C}\,(\mathcal{BN}_1, \mathcal{BN}_2, g, g') = \mathcal{C}(\mathcal{BN}_2, \mathcal{BN}_1, g', g)$.*

Proof. Straightforward based on the commutativity of conjunction. □

Composition gives a means of constructing new Boolean networks from well–understood and analysed Boolean networks. In particular, we would like to be able to infer properties and behaviour of a composed system from the underlying Boolean networks that have been composed. Being able to do this would allow us to construct large Boolean models with known properties without the limitations imposed by the state space explosion problem. The following definitions formalize the idea that the original behaviour of the underlying Boolean networks can be preserved in their composition.

We begin by defining *projection operators* which are able to extract states and traces from a composed system.

Definition 4. *(Projections) Let $\mathcal{C} = \mathcal{C}\,(\mathcal{BN}_1, \mathcal{BN}_2, g, g')$ be the new Boolean network constructed by composing $\mathcal{BN}_1$ and $\mathcal{BN}_2$ on entities g and g'. Let $S = (g^c\ g_1\ \ldots\ g_n\ g'_1\ \ldots\ g'_m) \in \mathcal{S}_{\mathcal{C}}$ be a global state in the composed system. Then we define the left $\mathcal{P}_1 : \mathcal{S}_{\mathcal{C}} \to \mathcal{S}_{\mathcal{BN}_1}$ and right $\mathcal{P}_2 : \mathcal{S}_{\mathcal{C}} \to \mathcal{S}_{\mathcal{BN}_2}$ projection operators by*

$$\mathcal{P}_1(S) = (g^c\ g_1\ \ldots\ g_n), \quad \mathcal{P}_2(S) = (g^c\ g'_1\ \ldots\ g'_m)$$

We can extend the projection operators to traces $\sigma = \langle S_1, S_2, \ldots \rangle \in Tr(\mathcal{C})$ by

$$\mathcal{P}_1(\sigma) = \langle \mathcal{P}_1(S_1), \mathcal{P}_1(S_2), \ldots \rangle, \quad \mathcal{P}_2(\sigma) = \langle \mathcal{P}_2(S_1), \mathcal{P}_2(S_2), \ldots \rangle$$

and let $\mathcal{P}_1(Tr(\mathcal{C}))$ and $\mathcal{P}_2(Tr(\mathcal{C}))$ represent the sets of projected traces derived by projecting each trace in $Tr(\mathcal{C})$.

Note that projected traces may not be well–defined traces in their corresponding Boolean network, i.e. $\mathcal{P}_j(Tr(\mathcal{C})) \not\subseteq Tr(\mathcal{BN}_j)$ may hold, for $j \in \{1, 2\}$.

We are interested in situations where composing two Boolean networks preserves their behaviour and define a notion of *compatibility.*

Definition 5. *(Compatibility) Let $\mathcal{C} = \mathcal{C}\,(\mathcal{BN}_1, \mathcal{BN}_2, g, g')$ be the Boolean network resulting from composing $\mathcal{BN}_1$ and $\mathcal{BN}_2$ on entities g and g'. Then we say that $\mathcal{BN}_1$ and $\mathcal{BN}_2$ are compatible on g and g' iff $Tr(\mathcal{BN}_1) \subseteq \mathcal{P}_1(Tr(\mathcal{C}))$ and $Tr(\mathcal{BN}_2) \subseteq \mathcal{P}_2(Tr(\mathcal{C}))$.*

To illustrate the definition of compatibility consider composing $\mathcal{BN}_{Ex1}$ and $\mathcal{BN}_{Ex2}$ to produce $\mathcal{C} = \mathcal{C}(\mathcal{BN}_{Ex1}, \mathcal{BN}_{Ex2}, g_1, g_4)$ (see Fig. 4). Then examples of projected traces in $\mathcal{P}_2(Tr(\mathcal{C}))$ (assuming state order $(g^c\ g_2\ g_3\ g_5)$) will be

$$\mathcal{P}_2(\langle 0100, 1011, 0100\rangle) = \langle 00, 11, 00\rangle \qquad \mathcal{P}_2(\langle 0001, 0001\rangle) = \langle 01, 01\rangle$$
$$\mathcal{P}_2(\langle 1001, 0100, 1011, 0100\rangle) = \langle 11, 00, 11\rangle \qquad \mathcal{P}_2(\langle 1100, 1100\rangle) = \langle 10, 10\rangle$$

It can be seen that $Tr(\mathcal{BN}_{Ex2}) \subseteq \mathcal{P}_2(Tr(\mathcal{C}))$ and so since we can also show $Tr(\mathcal{BN}_{Ex1}) \subseteq \mathcal{P}_1(Tr(\mathcal{C}))$ we know $\mathcal{BN}_{Ex1}$ and $\mathcal{BN}_{Ex2}$ are compatible on g_1 and g_4.

The following results show that composition is associative and so given Lemma 3 (commutativity) this means that the order in which multiple Boolean networks are composed does not affect the resulting model.

Lemma 6. *Let $\mathcal{BN}_1$, $\mathcal{BN}_2$ and $\mathcal{BN}_3$ be three Boolean networks, and let $g_1 \in \mathcal{BN}_1$, $g_2, g_3 \in \mathcal{BN}_2$, $g_2 \neq g_3$, and $g_4 \in \mathcal{BN}_3$. Then we have*

$$\mathcal{C}(\mathcal{C}\left(\mathcal{BN}_1, \mathcal{BN}_2, g_1, g_2\right), \mathcal{BN}_3, g_3, g_4) = \mathcal{C}(\mathcal{BN}_1, \mathcal{C}(\mathcal{BN}_2, \mathcal{BN}_3, g_3, g_4), g_1, g_2)$$

Proof. Let $\mathcal{C}_2 = \mathcal{C}\left(\mathcal{BN}_1, \mathcal{BN}_2, g_1, g_2\right)$, $\mathcal{C}_3 = \mathcal{C}(\mathcal{C}_2, \mathcal{BN}_3, g_3, g_4)$, and let $\mathcal{C}_4 = \mathcal{C}(\mathcal{BN}_2, \mathcal{BN}_3, g_3, g_4)$, $\mathcal{C}_5 = \mathcal{C}(\mathcal{BN}_1, \mathcal{C}_4, g_1, g_2)$. Let g_2^c be the entity representing the merge of g_1 and g_2, and g_4^c the merge of g_3 and g_4. Then by Definition 2 it suffices to show: (1) $F_{\mathcal{C}_3}(g_2^c) = F_{\mathcal{C}_5}(g_2^c)$; and (2) $F_{\mathcal{C}_3}(g_4^c) = F_{\mathcal{C}_5}(g_4^c)$.

We prove (1) as follows. By Definition 2 we know

$$F_{\mathcal{C}_3}(g_2^c) = F_{\mathcal{C}_2}(g_2^c), \quad \text{and} \quad F_{\mathcal{C}_2}(g_2^c) = F_1(g_1) \wedge F_2(g_2)$$

where $F_1(g_1) \wedge F_2(g_2)$ represents the function formed by the conjunction of the results of the two subfunctions $F_1(g_1)$ and $F_2(g_2)$. Then it follows from above that

$$F_{\mathcal{C}_3}(g_2^c) = F_1(g_1) \wedge F_2(g_2) \tag{I}$$

Again, by Definition 2 we know

$$F_{c_4}(g_2) = F_2(g_2), \quad \text{and} \quad F_{c_5}(g_2^c) = F_1(g_1) \wedge F_{c_4}(g_2)$$

and so it follows that

$$F_{c_5}(g_2^c) = F_1(g_1) \wedge F_2(g_2) \tag{II}$$

The result therefore follows by (I) and (II). The proof of (2) follows along similar lines to above. □

Lemma 7. *Let $\mathcal{BN}_1$, $\mathcal{BN}_2$ and $\mathcal{BN}_3$ be three Boolean networks, and let $g_1 \in \mathcal{BN}_1$, $g_2 \in \mathcal{BN}_2$, and $g_4 \in \mathcal{BN}_3$. Then we have*

$$\mathcal{C}(\mathcal{C}\left(\mathcal{BN}_1, \mathcal{BN}_2, g_1, g_2\right), \mathcal{BN}_3, g_2^c, g_4) = \mathcal{C}(\mathcal{BN}_1, \mathcal{C}(\mathcal{BN}_2, \mathcal{BN}_3, g_2, g_4), g_1, g_4^c)$$

where g_2^c is the entity representing the merge of g_1 and g_2, and g_4^c the merge of g_2 and g_4.

Proof. Let $\mathcal{C}_2 = \mathcal{C}(\mathcal{BN}_1, \mathcal{BN}_2, g_1, g_2)$, $\mathcal{C}_3 = \mathcal{C}(\mathcal{C}_2, \mathcal{BN}_3, g_2^c, g_4)$, and g_3^c be the entity representing the merge of g_2^c and g_4. Let $\mathcal{C}_4 = \mathcal{C}(\mathcal{BN}_2, \mathcal{BN}_3, g_2, g_4)$, $\mathcal{C}_5 = \mathcal{C}(\mathcal{BN}_1, \mathcal{C}_4, g_1, g_4^c)$, and g_5^c be the entity representing the merge of g_1 and g_4^c. To show that $\mathcal{C}_3 = \mathcal{C}_5$ we need to show that $F_{c_3}(g_3^c) = F_{c_5}(g_5^c)$. By Definition 2 we know

$$F_{c_3}(g_3^c) = F_{c_2}(g_2^c) \wedge F_3(g_4), \quad \text{and} \quad F_{c_2}(g_2^c) = F_1(g_1) \wedge F_2(g_2)$$

and so it follows that

$$F_{c_3}(g_3^c) = (F_1(g_1) \wedge F_2(g_2)) \wedge F_3(g_4) \tag{III}$$

Again, by Definition 2 we know

$$F_{c_5}(g_5^c) = F_1(g_1) \wedge F_{c_4}(g_4^c), \quad \text{and} \quad F_{c_4}(g_4^c) = F_2(g_2) \wedge F_3(g_4)$$

and so it follows that

$$F_{c_5}(g_5^c) = F_1(g_1) \wedge (F_2(g_2) \wedge F_3(g_4)) \tag{IV}$$

Then the result follows by (III), (IV) and the associativity of $\wedge$. □

4 Compatibility and Alignment

In this section we investigate how to infer compatibility without using the composed model. We formalise the property of *alignment* which we show is sufficient for obtaining compatibility. We use this result to show that duplicate Boolean networks are compatible under composition of corresponding entities.

For any Boolean network $\mathcal{BN}$ with entities $G = \{g_1, \ldots, g_n\}$, global state $S = (s_1 \ldots s_n) \in S_{\mathcal{BN}}$ and any entity $g_i \in \mathcal{BN}$ we define $\rho_{g_i}(S) = s_i$. Then $\rho_{g_i}(\sigma)$ denotes the *projected trace* of entity $g_i \in \mathcal{BN}$ on trace $\sigma = \langle S_1, S_2, \ldots \rangle \in Tr(\mathcal{BN})$ defined by $\rho_{g_i}(\sigma) = \langle \rho_{g_i}(S_1), \rho_{g_i}(S_2), \ldots \rangle$. We let $\rho_{g_i}(Tr(\mathcal{BN})) = \{\rho_{g_i}(\sigma) | \ \sigma \in Tr(\mathcal{BN})\}$. As an example, consider projecting the traces of $\mathcal{BN}_{Ex2}$ (Fig. 3) on g_4 which gives $\rho_{g_4}(Tr(\mathcal{BN}_{Ex2})) = \{\langle 0,1,0\rangle, \langle 0,0\rangle, \langle 1,1\rangle, \langle 1,0,1\rangle\}$.

We can now define the property of *alignment* as follows.

Definition 8. *(Alignment) Let $\mathcal{BN}_1$ and $\mathcal{BN}_2$ be two Boolean networks and let $g \in \mathcal{BN}_1$ and $g' \in \mathcal{BN}_2$. Then we say that $\mathcal{BN}_1$ and $\mathcal{BN}_2$ are aligned on g and g' iff $\rho_g(Tr(\mathcal{BN}_1)) = \rho_{g'}(Tr(\mathcal{BN}_2))$.*

Let $\mathcal{C} = \mathcal{C}(\mathcal{BN}_1, \mathcal{BN}_2, g, g')$, and $S^1 = (g \ g_1 \ ... \ g_n) \in \mathcal{S}_{\mathcal{BN}_1}$ and $S^2 = (g' \ g'_1 \ ... \ g'_m) \in \mathcal{S}_{\mathcal{BN}_2}$. Then we define $S^1 \wedge S^2 \in \mathcal{S}_{\mathcal{C}}$ by merging the state of g with g', that is $S^1 \wedge S^2 = (g \wedge g' \ g_1 \ ... \ g_n \ g'_1 \ ... \ g'_m)$. Let $\sigma_1 = \langle S_1^1, S_2^1, ...\rangle \in Tr(\mathcal{BN}_1)$ and $\sigma_2 = \langle S_1^2, S_2^2, ...\rangle \in Tr(\mathcal{BN}_2)$ be two traces. Then we define $\sigma_1 \wedge \sigma_2 = \langle S_1^1 \wedge S_1^2, S_2^1 \wedge S_2^2, ...\rangle$. Note that for any $\sigma_1 \in Tr(\mathcal{BN}_1)$ and $\sigma_2 \in Tr(\mathcal{BN}_2)$ we may have that $\sigma_1 \wedge \sigma_2 \notin Tr(\mathcal{C})$.

We now prove some useful results about merging aligned traces.

Lemma 9. *Let $\mathcal{BN}_1$ and $\mathcal{BN}_2$ be Boolean networks with $G_1 = \{g, g_1, \ldots, g_n\}$ and $G_2 = \{g', g'_1, \ldots, g'_m\}$. Let $\mathcal{C} = \mathcal{C}(\mathcal{BN}_1, \mathcal{BN}_2, g, g')$, and let $\sigma^1 \in Tr(\mathcal{BN}_1)$ and $\sigma^2 \in Tr(\mathcal{BN}_2)$ such that $\rho_g(\sigma^1) = \rho_{g'}(\sigma^2)$. Then we have:*

(i) $\sigma^1 \wedge \sigma^2 \in Tr(\mathcal{C})$*; and*
(ii) $\mathcal{P}_1(\sigma^1 \wedge \sigma^2) = \sigma^1$ *and* $\mathcal{P}_2(\sigma^1 \wedge \sigma^2) = \sigma^2$*.*

Proof. Let $\sigma^1 = \langle S_1, S_2, ...\rangle \in Tr(\mathcal{BN}_1)$ and $\sigma^2 = \langle T_1, T_2, ...\rangle \in Tr(\mathcal{BN}_2)$ such that $\rho_g(\sigma^1) = \rho_{g'}(\sigma^2)$. In the following we consider an arbitrary synchronous update step in the above traces: $S_i \rightarrow S_{i+1}$ and $T_i \rightarrow T_{i+1}$, where $S_i = (s^i\ s_1^i\ ...\ s_n^i)$, $S_{i+1} = (s^{i+1}\ s_1^{i+1}\ ...\ s_n^{i+1})$, $T_i = (t^i\ t_1^i\ ...\ t_m^i)$, and $T_{i+1} = (t^{i+1}\ t_1^{i+1}\ ...\ t_m^{i+1})$. Note that by our assumption $\rho_g(\sigma^1) = \rho_{g'}(\sigma^2)$ we know $s^i = t^i$ and so by idempotency of $\wedge$ we have

$$s^i \wedge t^i = s^i = t^i \tag{V}$$

(i) To show $\sigma^1 \wedge \sigma^2 \in Tr(\mathcal{C})$, it suffices to show

$$(s^i\ s_1^i\ ...\ s_n^i) \wedge (t^i\ t_1^i\ ...\ t_m^i) \rightarrow (s^{i+1}\ s_1^{i+1}\ ...\ s_n^{i+1}) \wedge (t^{i+1}\ t_1^{i+1}\ ...\ t_m^{i+1})$$

is a synchronous update step in $\mathcal{C}$. We do this in three stages by considering each possible entity $h \in \mathcal{C}$. (Note to simplify the proof we assume $N_1(h^1) = G_1$ and $N_2(h^2) = G_2$, for any $h^1 \in G_1$ and $h^2 \in G_2$.)

(1) Suppose $h = g_j \in \mathcal{BN}_1$, for some $j \in \{1, \ldots, n\}$. Then by the definition of merging states and (V) above we have

$$F(g_j)(s^i \wedge t^i, s_1^i, ..., s_n^i) = F_1(g_j)(s^i, s_1^i, ..., s_n^i)$$

By definition of σ^1 we know $F_1(g_j)(s^i, s_1^i, ..., s_n^i) = s_j^{i+1}$ and so it follows that

$$F(g_j)(s^i \wedge t^i, s_1^i, ..., s_n^i) = s_j^{i+1}$$

as required.

(2) Suppose $h = g'_j \in \mathcal{BN}_2$, for some $j \in \{1, \ldots, m\}$. Then we can prove

$$F(g'_j)(s^i \wedge t^i, t_1^i, ..., t_m^i) = t_j^{i+1}$$

using a similar approach to (1) above.

(3) Suppose $h = g^c \in \mathcal{C}$. Then by Definition 2 and (I) above we have

$$F(g^c)(s^i \wedge t^i, s_1^i, ..., s_n^i, t_1^i, ..., t_m^i) = F_1(g)(s^i, s_1^i, ..., s_n^i) \wedge F_2(g')(t^i, t_1^i, ..., t_m^i)$$

Then by our assumptions on σ^1 and σ^2 we have

$$F_1(g)(s^i, s_1^i, ..., s_n^i) \wedge F_2(g')(t^i, t_1^i, ..., t_m^i) = s^{i+1} \wedge t^{i+1}$$

and so the result follows as required.

(ii) By definition of merging traces it suffices to show that

$$\mathcal{P}_1(S_i \wedge T_i) = S_i \text{ and } \mathcal{P}_2(S_i \wedge T_i) = T_i$$

for any $i \in \mathbb{N}$. By definition of merging states we have

$$\mathcal{P}_1(S_i \wedge T_i) = (s^i \wedge t^i\ s_1^i\ ...\ s_n^i) \text{ and } \mathcal{P}_2(S_i \wedge T_i) = (s^i \wedge t^i\ t_1^i\ ...\ t_m^i)$$

Then the result follows by (V) above. □

We can now prove that *alignment* is a sufficient property for *compatibility.*

Theorem 10. *Let $\mathcal{BN}_1$ and $\mathcal{BN}_2$ be two BNs with $g \in \mathcal{BN}_1$ and $g' \in \mathcal{BN}_2$. Then if $\mathcal{BN}_1$ and $\mathcal{BN}_2$ are aligned on g and g' then $\mathcal{BN}_1$ and $\mathcal{BN}_2$ are compatible on g and g'.*

Proof. Let $\mathcal{C} = \mathcal{C}(\mathcal{BN}_1, \mathcal{BN}_2, g, g')$. By Definition 5 we need to show the following: (i) $Tr(\mathcal{BN}_1) \subseteq \mathcal{P}_1(Tr(\mathcal{C}))$; and (ii) $Tr(\mathcal{BN}_2) \subseteq \mathcal{P}_2(Tr(\mathcal{C}))$.

(i) Since g *aligns* with g' we know that for each trace $\sigma^1 \in Tr(\mathcal{BN}_1)$ there exists $\sigma^2 \in Tr(\mathcal{BN}_2)$ such that $\rho_g(\sigma^1) = \rho_{g'}(\sigma^2)$. Then we need to show that $\sigma^1 \in \mathcal{P}_1(Tr(\mathcal{C}))$. By our assumption above on σ^1 and σ^2 and Lemma 9. (i) we know that $\sigma^1 \wedge \sigma^2 \in Tr(\mathcal{C})$ must hold. Then by Lemma 9. (ii) we have $\mathcal{P}_1(\sigma^1 \wedge \sigma^2) = \sigma^1$ and so $\sigma^1 \in \mathcal{P}_1(Tr(\mathcal{C}))$ as required.

(ii) The proof follows along similar lines to (i) above. □

The above result provides a means of ensuring compatibility holds without requiring the composed system to be considered. This is important since a composed model will be larger and so more affected by the state space explosion problem. Note that while alignment is a sufficient condition for compatibility it can be shown that it is not a necessary property for it. In future work we intend to investigate strengthening alignment so that it completely characterises compatibility (see Sect. 5).

We say that $\mathcal{BN}_1$ and $\mathcal{BN}_2$ are *duplicates* if they are the same Boolean network up to the renaming of entities (i.e. they are isomorphic). It is interesting to consider what happens when duplicate Boolean networks are merged on corresponding entities (where *corresponding* is defined in the obvious way based on the underlying isomorphism). As an illustration, consider the example presented in Fig. 5 based on composing two duplicate copies of $\mathcal{BN}_{Ex1}$ (Fig. 1).

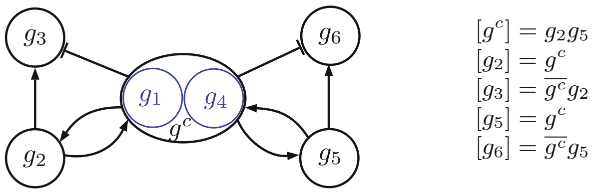


Fig. 5. Composing two duplicate copies of $\mathcal{BN}_{Ex1}$ on corresponding entities g_1 and g_4

We now use the alignment property to show that duplicate Boolean networks are compatible when composed on corresponding entities.

Theorem 11. *Let $\mathcal{BN}_1$ and $\mathcal{BN}_2$ be two duplicate Boolean networks and let $g \in \mathcal{BN}_1$ and $g' \in \mathcal{BN}_2$ be corresponding entities. Then $\mathcal{BN}_1$ and $\mathcal{BN}_2$ are compatible on g and g'.*

Proof. Since $\mathcal{BN}_1$ and $\mathcal{BN}_2$ are duplicates it follows (assuming a corresponding state order) that $Tr(\mathcal{BN}_1) = Tr(\mathcal{BN}_2)$. Thus by Definition 8 we know that $\mathcal{BN}_1$ and $\mathcal{BN}_2$ are aligned on corresponding entities g and g', and so by Theorem 10 we have that $\mathcal{BN}_1$ and $\mathcal{BN}_2$ are *compatible* on g and g' as required. □

5 Conclusions

In this paper we set out to develop a compositional framework for Boolean networks in order to facilitate the construction and analysis of large scale models. This work was motivated by interesting interactions with the synthetic biology group at Newcastle[1] and their search for formal tools and techniques to support their work on engineering biological systems. We have formally defined our compositional approach and introduced the notion of compatibility to formalize the preservation of a Boolean network's behaviour within a composed model. We formulated the alignment property which we showed was a sufficient condition for ensuring compatibility and used it to investigate the composition of duplicate models. Importantly, the alignment property makes no reference to the composed model and so helps avoid potentially limiting state space explosion issues. The compositional framework developed is supported by a prototype tool that automates the composition process and associated analysis.

A range of related work on composing Boolean networks can be found in the literature. For example, the properties of composing random Boolean network by computing the attractors compositionally is considered in [5]. Other work includes [15] in which a compositional approach is used to study a large-scale network. Our approach based on merging entities and characterising the preservation of model behaviour appears to be new.

In future work we intend to extend the alignment property to provide a complete characterisation of compatibility. Initial work in this area has focused on using a state graph to model the interference that can occur between Boolean networks in a composed model. We are also interested in using our compositional framework as the basis for decomposing large Boolean network models to aid analysis. Further, we intend to undertake a series of large case studies to investigate the applicability of the techniques and tools we have developed.

Acknowledgments. We would like to thank Will Peckham for his work on developing tool support for our framework. We also acknowledge the financial support provided by Faculty of Computing and Information Technology, King Abdulaziz University.

[1] www.ncl.ac.uk/csbb/.

References

1. Akutsu, T., Miyano, S., Kuhara, S., et al.: Identification of genetic networks from a small number of gene expression patterns under the boolean network model. Pacific Symp. Biocomputing **4**, 17–28 (1999)
2. Banks, R., Steggles, L.J.: An abstraction theory for qualitative models of biological systems. Theoret. Comput. Sci. **431**, 207–218 (2012)
3. Bartocci, E., Lió, P.: Computational modeling, formal analysis, and tools for systems biology. PLoS Comput. Biol. **12**(1), e1004591 (2016)
4. De Jong, H.: Modeling and simulation of genetic regulatory systems: a literature review. J. Comput. Biol. **9**(1), 67–103 (2002)
5. Dubrova, E., Teslenko, M.: Compositional properties of random boolean networks. Phys. Rev. E **71**, 056116 (2005). http://link.aps.org/doi/10.1103/PhysRevE.71.056116
6. Harvey, I., Bossomaier, T.: Time out of joint: attractors in asynchronous random boolean networks. In: Proceedings of the Fourth European Conference on Artificial Life, pp. 67–75. MIT Press, Cambridge (1997)
7. Huang, S., Ingber, D.E.: Shape-dependent control of cell growth, differentiation, and apoptosis: switching between attractors in cell regulatory networks. Exp. Cell Res. **261**(1), 91–103 (2000)
8. Kauffman, S.A.: Metabolic stability and epigenesis in randomly constructed genetic nets. J. Theor. Biol. **22**(3), 437–467 (1969)
9. Kauffman, S.A.: The Origins of OIrder: Self Organization and Selection in Evolution. Oxford University Press, USA (1993)
10. Rosenblueth, D.A., Muñoz, S., Carrillo, M., Azpeitia, E.: Inference of boolean networks from gene interaction graphs using a SAT solver. In: Dediu, A.-H., Martín-Vide, C., Truthe, B. (eds.) AlCoB 2014. LNCS, vol. 8542, pp. 235–246. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-07953-0_19
11. Saadatpour, A., Albert, R.: Boolean modeling of biological regulatory networks: a methodology tutorial. Methods **62**(1), 3–12 (2013)
12. Schaub, M.A., Henzinger, T.A., Fisher, J.: Qualitative networks: a symbolic approach to analyze biological signaling networks. BMC Syst. Biol. **1**(4) (2007)
13. Steggles, L.J., Banks, R., Shaw, O., Wipat, A.: Qualitatively modelling and analysing genetic regulatory networks: a petri net approach. Bioinformatics **23**(3), 336–343 (2007). http://bioinformatics.oxfordjournals.org/content/23/3/336
14. Thieffry, D., Thomas, R.: Dynamical behaviour of biological regulatory networks–II. Immunity control in bacteriophage lambda. Bull. Math. Biol. **57**(2), 277–297 (1995)
15. Tournier, L., Chaves, M.: Interconnection of asynchronous boolean networks, asymptotic and transient dynamics. Automatica **49**(4), 884–893 (2013)
16. Wuensche, A.: Basins of attraction in network dynamics: a conceptual framework for biomolecular networks. In: Schlosser, G., Wagner, G.P. (eds.) Modularity in Development and Evolution, chap. 13, pp. 288–311. University of Chicago Press, Chicago (2004)

http://www.springer.com/978-3-319-71068-6

Theory and Practice of Natural Computing
6th International Conference, TPNC 2017, Prague,
Czech Republic, December 18-20, 2017, Proceedings
Martín-Vide, C.; Neruda, R.; Vega-Rodríguez, M.A. (Eds.)
2017, XI, 319 p. 89 illus., Softcover
ISBN: 978-3-319-71068-6