Compositional Analysis of Boolean Networks Using Local Fixed-Point Iterations

RP 2016, Aalborg, Denmark

Adrien Le Coënt [1], Laurent Fribourg [2], Romain Soulat [3]

September 22, 2016

[1]CMLA Centre de Mathématiques et de Leurs Applications
[2]LSV Laboratoire de Spécification et Vérification
[3]Thales Research & Technology

Outline

Outline

Synchronous Boolean Networks

A Boolean network (BN) is a discrete-time dynamical system subject to the rules

$$x(t+1) = f(x(t)) \tag{1}$$

where

- x is a vector of n Boolean variables and
- f is a vector of n Boolean functions on these variables.

Let $S = \{0,1\}^n$ be the set of all instances of x. We have: $f : S \to S$.

Decomposition

We suppose x decomposed into two vectors x_1 and x_2 of n_1 and n_2 components resp. ($n = n_1 + n_2$).

Let $S_1 = \{0,1\}^{n_1}$ and $S_2 = \{0,1\}^{n_2}$ be the sets of all instances of x_1 and x_2 resp. We have: $S = S_1 \times S_2$.

The BN can be written:

$$x_1(t+1) = f_1(x_1(t), x_2(t)) \quad (2)$$

$$x_2(t+1) = f_2(x_1(t), x_2(t)) \quad (3)$$

with $f_1 : S_1 \times S_2 \to S_1$ and $f_2 : S_1 \times S_2 \to S_2$.

NB: Way of partioning x, e.g., by exploiting the oriented graph associated to f, out of the scope of this work (see, e.g., [Akutsu]).

Example

Consider $x = (A, B, C, E, F, G, H, I)$,
$x_1 = (A, F, G, H, I)$, $x_2 = (B, C, E)$
and the functions f_1 and f_2 defined by the systems:

$$\boxed{\begin{array}{l} A(t+1) = 1 \wedge H(t) \\ F(t+1) = E(t) \wedge (E(t) \vee G(t)) \\ G(t+1) = 1 \wedge (B(t) \vee E(t)) \\ H(t+1) = F(t) \wedge (F(t) \vee G(t)) \\ I(t+1) = H(t) \wedge (H(t) \vee I(t)) \end{array}}$$

$$\boxed{\begin{array}{l} B(t+1) = A(t) \wedge (A(t) \vee C(t)) \\ C(t+1) = I(t) \\ E(t+1) = 1 \wedge C(t) \wedge (C(t) \vee F(t)) \end{array}}$$

Attractors

- Major interest of BNs: Finding its cycles.
  In industrial case studies, such as railway interlocking, one for example wants to show no cycle of length >1.

- Since f deterministic and S finite, every derivation ends to a cycle.
  The set of elements composing this cycle is called an attractor.
  An attractor of length 1 is called a stationary state

- Complexity of finding all the attractors of a BN is NP-hard [zhang2007].

Let F^* denote the union of all the attractors of the BN defined by f.

Outline

Lifting f to the powerset

- The "lifted" version $f : 2^S \to 2^S$ is defined, for all $X \subseteq S$, by:

$$f(X) = \{y \mid y = f(x) \text{ for } x \text{ in } X\}$$

- A fixed-point of f is a set $X \subseteq S$ s.t. $f(X) = X$. By Knaster-Tarski th., there is a greatest fixed-point of f given by:

$$gfp(f) = \bigcap_{k \geq 0} f^k(S)$$

- <u>Proposition:</u>
  1. The set F^* of attractors is given by: $F^* = gfp(f)$
  2. If $X \supseteq F^*$ then $F^* = \bigcap_{k \geq 0} f^k(X)$

Constructing an overapproximation of F^*

- Abstraction ("separation"): $\alpha : 2^S \to 2^{S_1} \times 2^{S_2}$ is defined, for all set $X \subseteq S$, by:

$$\alpha(X) = (\pi_1(X), \pi_2(X))$$

  where π_1 and π_2 are the 1st and 2nd projection of S to S_1 and S_2

- Concretization ("glueing"): $\gamma : 2^{S_1} \times 2^{S_2} \to 2^S$ is defined, for all sets $X_1 \subseteq S_1$, $X_2 \subseteq S_2$ by:

$$\gamma(X_1, X_2) = X_1 \times X_2$$

Proposition: α and γ satisfy the properties of a Galois connection:

- $\alpha(\gamma(X_1, X_2)) \subseteq (X_1, X_2)$, for all $X_1 \subseteq S_1$, $X_2 \subseteq S_2$
- $\gamma(\alpha(X)) \supseteq X$, for all $X \subseteq S$

Constructing an overapproximation of F^* (cont'd)

- Abstract function $\tilde{f}$
  Let $\tilde{f} : 2^{S_1} \times 2^{S_2} \to 2^{S_1} \times 2^{S_2}$ defined by $\tilde{f} = \alpha f \gamma$, i.e., for all $X_1 \subseteq S_1$ and $X_2 \subseteq S_2$:

$$\tilde{f}(X_1, X_2) = (f_1(X_1, X_2), f_2(X_1, X_2))$$

- By Cousot's theorem, we have:

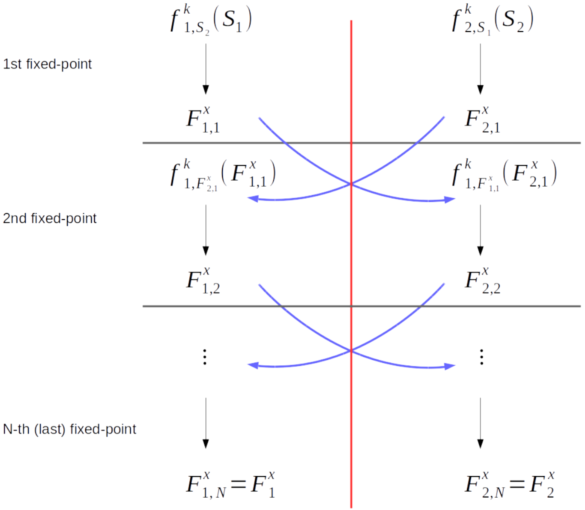
$\gamma(gfp(\tilde{f})) \supseteq gfp(f)$, hence: $\gamma(\bigcap_{k \geq 0} \tilde{f}^k((S_1, S_2))) \supseteq F^*$

Let us denote $\bigcap_{k \geq 0} \tilde{f}^k((S_1, S_2))$ by (F_1^*, F_2^*). We have:

Proposition:

1. $F_1^* \times F_2^* \supseteq F^*$
2. $F^* = \bigcap_{k \geq 0} f^k(F_1^* \times F_2^*)$

Construction of F_1^*, F_2^* by "iterative reduction"



Extensions

1. Decomposition of the BN into more than 2 sub-systems
2. Basins of attraction
3. Controlled Boolean networks ($x(t+1) = f(x(t), u(t))$)
4. Use of the ℓ-th power $g = f^\ell$ of f in order to refine the over-approximation, i.e., use of $\tilde{g} = \tilde{f}^\ell$ and computation of $G_1^* \times G_2^* \subseteq F_1^* \times F_2^*$

Example of (A,B,C,E,F,G,H,I) system

S has 256 elements. Let $\ell = 2$

$G_{1,0} = S_1$	$G_{2,0} = S_2$
$G_{1,1} = \{00000, 00001, 00100, \ldots, 11110\}$	$G_{2,1} = \{000, 001, 011, 101\}$
$G_{1,2} = \{10000, 11000, 11010, \ldots, 10100\}$	$G_{2,2} = G_{2,1}$
$G_{1,3} = G_{1,2}$	$G_{2,3} = G_{2,1}$

Hence $G_1^* \times G_2^*$ has $9 \times 4 = 36$ elements.

$F^* = f^{10}(G_1^* \times G_2^*)$ has 8 elements (unique cycle).

Comparison with other compositional methods for BNs

Our method

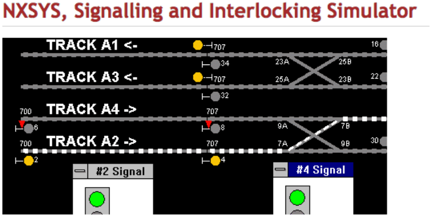
1. Partitioning the set of variables
2. Iterative reduction: repeated computation of local fixed-points (F_1^* and F_2^*)
3. Global f-reduction of $F_1^* \times F_2^*$

Closer related method (e.g. [Guo-Yang-Wu-Le-Sun 2014])

1. Decomposing the set of variables with common shared variables
2. Finding local attractors for each subsystem
3. Recombining local attractors in order to maintain compatibility for shared variables

Outline

Application to railways interlocking (NXSYS)



- 50 Boolean variables
- Decomposition into 4 subsystems
- Computation of $F_1^* \times F_2^* \times F_3^* \times F_4^*$ takes 2 hours
- Computation of F^* by subsequent f-applications takes 12 hours.
- $|F^*| = 24M$

NB: Other tools of the state-of-art seem slower/unable to finding the attractors of such a BN with high node connectivity (indegree > 7)

Contribution and future work

Contribution

- First application of "iterative reduction" compositionality method to compute attractors of BNs
- Successful application to a railways example with 50 variables

Future work

- parallel implementation
- symbolic representation (BDD, SAT)
- application to a French railways station model (200 variables)