Update digraphs and Boolean networks

Julio B. Aracena Lucero
(J. Demongeot, E. Fanchon, E. Goles, L. Gómez, M. Montalva, A. Moreira, M. Noual and L. Salinas)

Departamento de Ingeniería Matemática
Facultad de Ciencias Físicas y Matemáticas
Universidad de Concepción

Contents

1. Boolean Networks
   - Definition
   - Connection Digraph
   - Deterministic Update Schedule
2. Update Digraph
   - Necessary conditions
   - Sufficient conditions
3. Some combinatorics problems about update digraphs
4. Inverse Problem
   - Complexity
   - Open Problems

Contents

1. Boolean Networks
   - Definition
   - Connection Digraph
   - Deterministic Update Schedule
2. Update Digraph
   - Necessary conditions
   - Sufficient conditions
3. Some combinatorics problems about update digraphs
4. Inverse Problem
   - Complexity
   - Open Problems

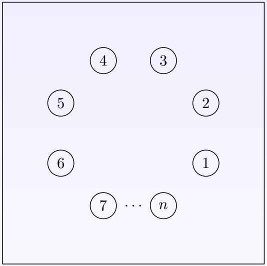
Contents

1. Boolean Networks
   - Definition
   - Connection Digraph
   - Deterministic Update Schedule
2. Update Digraph
   - Necessary conditions
   - Sufficient conditions
3. Some combinatorics problems about update digraphs
4. Inverse Problem
   - Complexity
   - Open Problems

Contents

1. Boolean Networks
   - Definition
   - Connection Digraph
   - Deterministic Update Schedule
2. Update Digraph
   - Necessary conditions
   - Sufficient conditions
3. Some combinatorics problems about update digraphs
4. Inverse Problem
   - Complexity
   - Open Problems

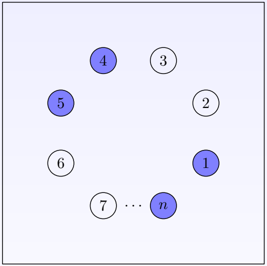
Boolean Network (BN)



A BN $N = (F, s)$ is defined by:

- A global transition function $F = (f_1, \ldots, f_n) : \{0,1\}^n \to \{0,1\}^n$. f_i local activation function.
- An update schedule s
- $x_i(t) \in \{0,1\}$ is the node state i on time t.

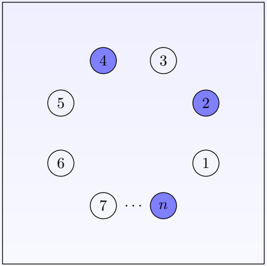
Boolean Network (BN)



A BN $N = (F, s)$ is defined by:

- A global transition function $F = (f_1, \ldots, f_n) : \{0,1\}^n \to \{0,1\}^n$. f_i local activation function.
- An update schedule s
- $x_i(t) \in \{0,1\}$ is the node state i on time t.

Boolean Network (BN)



A BN $N = (F, s)$ is defined by:

- A global transition function $F = (f_1, \ldots, f_n) : \{0,1\}^n \rightarrow \{0,1\}^n$. f_i local activation function.
- An update schedule s
- $x_i(t) \in \{0,1\}$ is the node state i on time t.

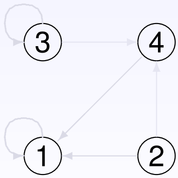
Connection Digraph

Given $N = (F, s)$, the connection digraph $G^F = (V, A)$ is defined as:

- $V = \{1, \ldots, n\}$,
- $(i, j) \in A \iff f_j$ depends on x_i,

$$f_j(x_1, \ldots, x_{i-1}, 0, x_{i+1}, \ldots, x_n) \neq f_j(x_1, \ldots, x_{i-1}, 1, x_{i+1}, \ldots, x_n)$$

$$f_1(x) = (x_1 \wedge x_2) \vee x_4$$
$$f_2(x) = 0$$
$$f_3(x) = x_3 \wedge (x_4 \vee \bar{x}_4)$$
$$f_4(x) = x_2 \wedge \bar{x}_3$$



$$V^-(j) = \{i : (i, j) \in A\}$$

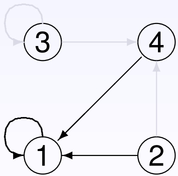
Connection Digraph

Given $N = (F, s)$, the connection digraph $G^F = (V, A)$ is defined as:

- $V = \{1, \ldots, n\}$,
- $(i, j) \in A \iff f_j$ depends on x_i,

$$f_j(x_1, \ldots, x_{i-1}, 0, x_{i+1}, \ldots, x_n) \neq f_j(x_1, \ldots, x_{i-1}, 1, x_{i+1}, \ldots, x_n)$$

$$f_1(x) = (x_1 \wedge x_2) \vee x_4$$
$$f_2(x) = 0$$
$$f_3(x) = x_3 \wedge (x_4 \vee \bar{x}_4)$$
$$f_4(x) = x_2 \wedge \bar{x}_3$$



$$V^-(j) = \{i : (i, j) \in A\}$$

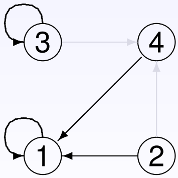
Connection Digraph

Given $N = (F, s)$, the connection digraph $G^F = (V, A)$ is defined as:

- $V = \{1, \ldots, n\}$,
- $(i, j) \in A \iff f_j$ depends on x_i,

$$f_j(x_1, \ldots, x_{i-1}, 0, x_{i+1}, \ldots, x_n) \neq f_j(x_1, \ldots, x_{i-1}, 1, x_{i+1}, \ldots, x_n)$$

$$f_1(x) = (x_1 \wedge x_2) \vee x_4$$
$$f_2(x) = 0$$
$$f_3(x) = x_3 \wedge (x_4 \vee \bar{x}_4)$$
$$f_4(x) = x_2 \wedge \bar{x}_3$$



$$V^-(j) = \{i : (i, j) \in A\}$$

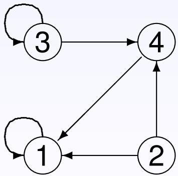
Connection Digraph

Given $N = (F, s)$, the connection digraph $G^F = (V, A)$ is defined as:

- $V = \{1, \ldots, n\}$,
- $(i, j) \in A \iff f_j$ depends on x_i,

$$f_j(x_1, \ldots, x_{i-1}, 0, x_{i+1}, \ldots, x_n) \neq f_j(x_1, \ldots, x_{i-1}, 1, x_{i+1}, \ldots, x_n)$$

$$f_1(x) = (x_1 \wedge x_2) \vee x_4$$
$$f_2(x) = 0$$
$$f_3(x) = x_3 \wedge (x_4 \vee \bar{x}_4)$$
$$f_4(x) = x_2 \wedge \bar{x}_3$$

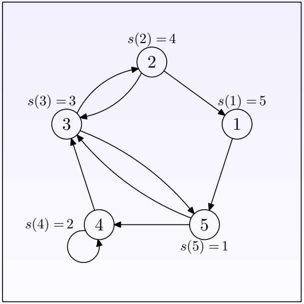


$$V^-(j) = \{i : (i, j) \in A\}$$

Deterministic Update Schedule

A deterministic update schedule is a function $s : \{1, \ldots, n\} \to \{1, \ldots, n\}$ such that $s(\{1, \ldots, n\}) = \{1, \ldots, m\},\ m \leq n$.

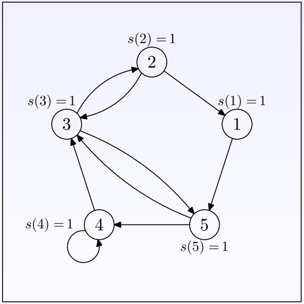
- Sequential: $s(\{1, \ldots, n\}) = \{1, \ldots, n\}$.
- Parallel (s_p): $s(\{1, \ldots, n\}) = \{1\}$
- Block-Sequential: $s(\{1, \ldots, n\}) = \{1, \ldots, m\}, m < n$.



Deterministic Update Schedule

A deterministic update schedule is a function $s : \{1, \ldots, n\} \to \{1, \ldots, n\}$ such that $s(\{1, \ldots, n\}) = \{1, \ldots, m\},\ m \leq n$.

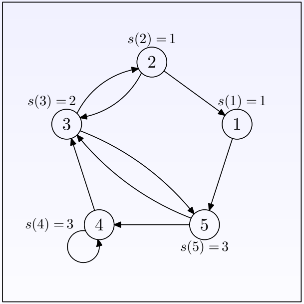
- Sequential: $s(\{1, \ldots, n\}) = \{1, \ldots, n\}$.
- Parallel (s_p): $s(\{1, \ldots, n\}) = \{1\}$
- Block-Sequential: $s(\{1, \ldots, n\}) = \{1, \ldots, m\},\ m < n$.



Deterministic Update Schedule

A deterministic update schedule is a function $s : \{1, \ldots, n\} \to \{1, \ldots, n\}$ such that $s(\{1, \ldots, n\}) = \{1, \ldots, m\},\ m \leq n$.

- Sequential: $s(\{1, \ldots, n\}) = \{1, \ldots, n\}$.
- Parallel (s_p): $s(\{1, \ldots, n\}) = \{1\}$
- Block-Sequential: $s(\{1, \ldots, n\}) = \{1, \ldots, m\}, m < n$.



Dynamical bevahior

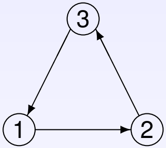
The dynamics of the Boolean network $N = (F, s)$ is given by:

$$x_i(t+1) = f_i(x_1(t_1), \ldots, x_j(t_j), \ldots, x_n(t_n)),$$

where $t_j = \begin{cases} t & s(i) \leq s(j) \\ t+1 & s(i) > s(j) \end{cases}$

Thus, $\exists\ F^s : \{0,1\}^n \to \{0,1\}^n$, with $F^s(x) = (f_1^s(x), \ldots, f_n^s(x))$ such that: $x(t+1) = F^s(x(t))$. Hence, $N^p = (F^s, s_p)$ is equivalent to N.

Example:



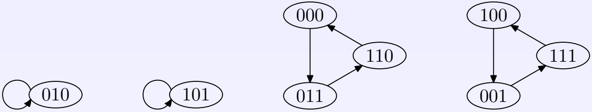
$F : \{0,1\}^3 \to \{0,1\}^3$

$$
\begin{aligned}
f_1(x_1, x_2, x_3) &= x_3 \\
f_2(x_1, x_2, x_3) &= \bar{x}_1 \\
f_3(x_1, x_2, x_3) &= \bar{x}_2
\end{aligned}
$$

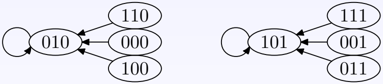
$x(t)$	s=(1)(2)(3) $F^s(x(t))$	s=(1,2,3) $x(t+1)$	s=(2)(1,3) $x(t+1)$
000	010	011	010
001	101	111	110
010	010	010	010
011	101	110	110
100	010	001	001
101	101	101	101
110	010	000	001
111	101	100	101

Iteration Graph

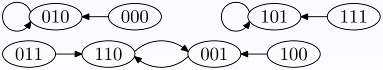
$s = (1,2,3)$



$s = (1)(2)(3)$



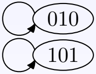
$s = (2)(1,3)$



Attractors

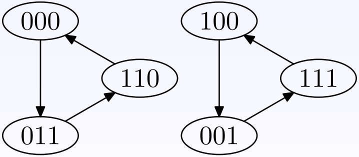
Fixed Point:

$x \in \{0,1\}^n$, $F^s(x) = x$



Limit cycle:

$x^0, x^1, \ldots, x^{p-1}, x^p \in \{0,1\}^n$,
$F^s(x^i) = x^{i+1}$,
$\forall i = 0, \ldots, p-1$, $x^0 = x^p$.

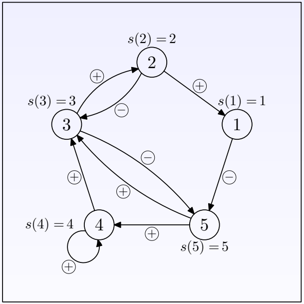


Update Digraph

Given $N = (F, s)$ a BN and G^F its connection digraph, we define:

- Update digraph: $G_s^F = (G^F, \text{lab}_s)$
- $\text{lab}_s : E(G^F) \to \{\oplus, \ominus\}$,

$$\text{lab}_s(i,j) = \begin{cases} \oplus & ; \quad s(i) \geq s(j) \\ \ominus & ; \quad s(i) < s(j) \end{cases}$$



Equivalent update schedules

Theorem

Let $N_1 = (F, s_1)$ and $N_2 = (F, s_2)$ be two Boolean networks which are different only in the update schedule. If $G^F_{s_1} = G^F_{s_2}$, then both dynamical behaviors are equal.

We define the equivalence relation between update schedules:

$$s_1 \sim_N s_2 \Longleftrightarrow G^F_{s_1} = G^F_{s_2}.$$

Hence, we denote $[s]_N = \{s' : s \sim_N s'\}$.

Example

$s_1 = (1)(2)(3)(4)$ $\qquad$ $s_2 = (1,2)(3)(4)$

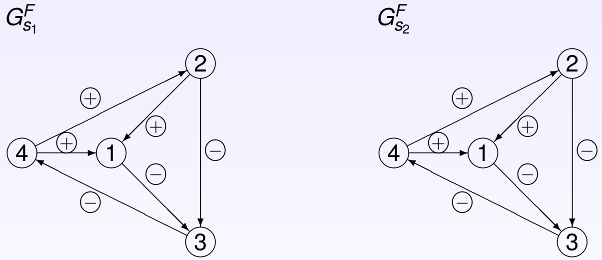


Figure: OR-Boolean Network with two equivalent schedules that yield the same dynamical behavior.

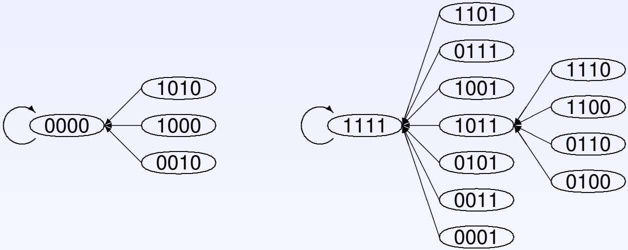


Figure: Dynamical behavior of both OR-Boolean networks.

Equivalent update schedules

The converse of previous theorem is not true.

$s_1 = (1)(2)(3)(4)$ $\qquad$ $s_3 = (1)(3)(2)(4)$

$G^F_{s_1}$ $\qquad$ $G^F_{s_3}$

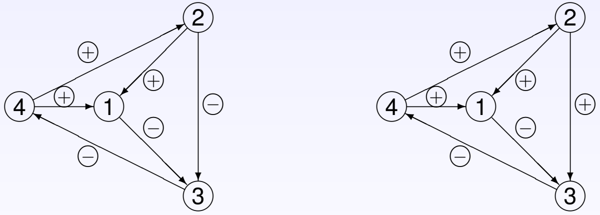


Figure: OR-Boolean Networks with two non-equivalent schedules that have the same dynamical behavior.

Dynamics and update digraphs

Theorem

Let $N = (F, s)$ be a Boolean network. There exists s' an update schedule such that $N' = (F, s')$ does not preserve the limit cycles of $N = (F, s)$.

Proof(Idea)
Let $\{i_1, i_2, \ldots, i_n\}$ with $s(i_1) \leq s(i_2) \leq \cdots \leq s(i_n)$. Then, $s'(i_j) = n + 1 - j$, ie $s'(i_1) > s'(i_2) > \cdots > s'(i_n)$, verifies the property.

Limit cycles in BNs with parallel and sequential schedules

Theorem

Let $N_p = (F, s_p)$ and $N_q = (F, s_q)$ be two BNs where the loops are monotonic and such that s_p and s_q are the parallel and a sequential update, respectively. Then, $LC(N_p) \cap LC(N_q) = \emptyset$.

Complexity of problems about the limit cycles

Limit Cycle Problem

Given a Boolean network $N = (F, s)$ and $C \in LC(N)$. There exists $\hat{s} \notin [s]_{G^F}$ such that $C \in LC(\hat{N} = (F, \hat{s}))$?

Limit Cycle Set Problem

Given a Boolean network $N = (F, s)$. There exists $\hat{s} \notin [s]_{G^F}$ such that $LC(N) = LC(\hat{N} = (F, \hat{s}))$?

Complexity of problems about the limit cycles

Limit Cycle Problem

Given a Boolean network $N = (F, s)$ and $C \in LC(N)$. There exists $\hat{s} \notin [s]_{G^F}$ such that $C \in LC(\hat{N} = (F, \hat{s}))$?

Limit Cycle Set Problem

Given a Boolean network $N = (F, s)$. There exists $\hat{s} \notin [s]_{G^F}$ such that $LC(N) = LC(\hat{N} = (F, \hat{s}))$?

Complexity of problems about the limit cycles

Limit Cycle Problem

Given a Boolean network $N = (F, s)$ and $C \in LC(N)$. There exists $\hat{s} \notin [s]_{G^F}$ such that $C \in LC(\hat{N} = (F, \hat{s}))$?

Limit Cycle Set Problem

Given a Boolean network $N = (F, s)$. There exists $\hat{s} \notin [s]_{G^F}$ such that $LC(N) = LC(\hat{N} = (F, \hat{s}))$?

Common Limit Cycle Problem

Given a Boolean network $N = (F, s)$. There exists $\hat{s} \notin [s]_{G^F}$ such that $LC(N) \cap LC(\hat{N} = (F, \hat{s})) \neq \emptyset$?

We proved that all these problems are NP-hard.

Algorithm

We designed a polynomial algorithm that works as a necessary condition to share limit cycles.

- Given: (G, s_1, s_2)
- $Test(G, s_1, s_2) = TRUE$, then no matter the global function F used, $N_1 = (F, s_1)$ and $N_2 = (F, s_2)$ never share any limit cycle.
- Polynomial ($O(n^2)$)

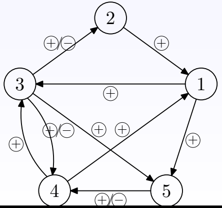
Algorithm

We designed a polynomial algorithm that works as a necessary condition to share limit cycles.

- Given: (G, s_1, s_2)
- $Test(G, s_1, s_2) = TRUE$, then no matter the global function F used, $N_1 = (F, s_1)$ and $N_2 = (F, s_2)$ never share any limit cycle.
- Polynomial ($O(n^2)$)

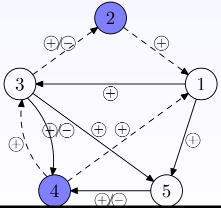
Test 1

$M = \emptyset$;
$N = V$
While $\exists i \in N$ such that
        $((V^-(i) \cap N = \emptyset)$ or
        $(\exists j \in M, V^-(j) = \{i\}))$ or
        $(\forall j \in V^-(i) \cap N, ((\text{lab}_{s_1}(j,i) = \oplus \wedge \text{lab}_{s_2}(j,i) = \ominus))$ or
        $(\forall j \in V^-(i) \cap N, ((\text{lab}_{s_1}(j,i) = \ominus \wedge \text{lab}_{s_2}(j,i) = \oplus))$
    $M \longleftarrow M \cup \{i\}$
    $N \longleftarrow N \setminus \{i\}$
end While
if $M = V$ then return TRUE
else return FALSE



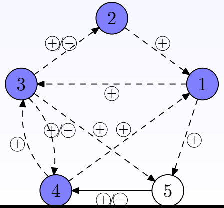
Test 1

$M = \emptyset$;
$N = V$
While $\exists i \in N$ such that
    $((V^{-}(i) \cap N = \emptyset)$ or
    $(\exists j \in M, V^{-}(j) = \{i\}))$ or
    $(\forall j \in V^{-}(i) \cap N, ((\text{lab}_{s_1}(j,i) = \oplus \wedge \text{lab}_{s_2}(j,i) = \ominus))$ or
    $(\forall j \in V^{-}(i) \cap N, ((\text{lab}_{s_1}(j,i) = \ominus \wedge \text{lab}_{s_2}(j,i) = \oplus))$
  $M \longleftarrow M \cup \{i\}$
  $N \longleftarrow N \setminus \{i\}$
end While
if $M = V$ then return TRUE
else return FALSE



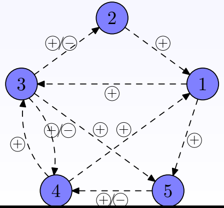
Test 1

$M = \emptyset$;
$N = V$
While $\exists i \in N$ such that
    $((V^-(i) \cap N = \emptyset)$ or
    $(\exists j \in M, V^-(j) = \{i\}))$ or
    $(\forall j \in V^-(i) \cap N, ((\text{lab}_{s_1}(j,i) = \oplus \wedge \text{lab}_{s_2}(j,i) = \ominus))$ or
    $(\forall j \in V^-(i) \cap N, ((\text{lab}_{s_1}(j,i) = \ominus \wedge \text{lab}_{s_2}(j,i) = \oplus))$
  $M \longleftarrow M \cup \{i\}$
  $N \longleftarrow N \setminus \{i\}$
end While
if $M = V$ then return TRUE
else return FALSE



Test 1

$M = \emptyset$;
$N = V$
While $\exists i \in N$ such that
  $((V^{-}(i) \cap N = \emptyset)$ or
  $(\exists j \in M, V^{-}(j) = \{i\}))$ or
  $(\forall j \in V^{-}(i) \cap N, ((\mathrm{lab}_{s_1}(j,i) = \oplus \wedge \mathrm{lab}_{s_2}(j,i) = \ominus))$ or
  $(\forall j \in V^{-}(i) \cap N, ((\mathrm{lab}_{s_1}(j,i) = \ominus \wedge \mathrm{lab}_{s_2}(j,i) = \oplus))$
 $M \longleftarrow M \cup \{i\}$
 $N \longleftarrow N \setminus \{i\}$
end While
if $M = V$ then return TRUE
else return FALSE



Sufficient conditions

Under certain conditions on the constant nodes of a given set of limit cycles, we proved that several equivalences classes can be built such that they share the given set of limit cycles.

Some combinatorics results about the update digraphs

Theorem

Let G be a digraph. Then,

$$|MFAS(G)| < |U(G)| \leq |FAS(G)|.$$

Proof (Idea)
$g : U(G) \to FAS(G), \forall lab \in U(G)$,
$g(lab) = \{a \in A(G) : lab(a) = \oplus\}$.
Besides, we define $h : MFAS(G) \to U(G), \forall F \in MFAS(G)$,
$h(F) = lab_F$, where $lab_F(a) = \oplus, \forall a \in F$ and
$lab_F(a) = \ominus, \forall a \in A(G) \setminus F$.

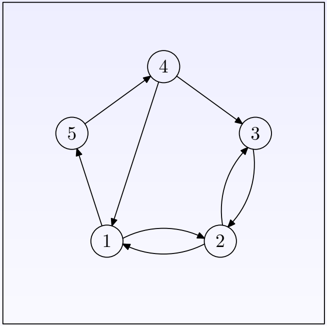
Theorem

Let G be a digraph with $|V(G)| = n$. Then, $F \subseteq A(G)$ is a minimal feedback arc set of G if and only if (G, lab_F) is an update digraph with a maximal number of negative arcs, where $lab_F(u, v) = \oplus \Leftrightarrow (u, v) \in F$.

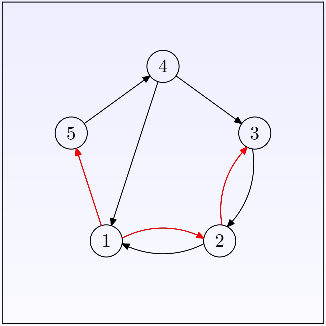
Proposition

Let (G, lab) be an update digraph with a maximal number of negative arcs. Then, there is not a positive cycle.

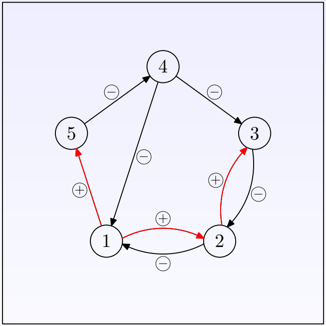
Example



Example



Example



Inverse Problem

Objetive

Given a certain characteristic of the dynamical behavior of a Boolean network, we want to find an update schedule that satisfies that particular property.

Complexity Problems

UPDATE SCHEDULE EXISTENCE PROBLEM (USE)

Given F a Boolean function and $x, y \in \{0,1\}^n$. There exists an update schedule s such that: $F^s(y) = x$?

BOOLEAN NETWORK PREDECESOR PROBLEM (BNPP)

Given $N = (F, s)$ a Boolean network and $x \in \{0,1\}^n$. There exists $y \in \{0,1\}^n$ such that: $F^s(y) = x$?.

We proved that these problems and some other variants are NP-Hard.

Complexity Problems

UPDATE SCHEDULE EXISTENCE PROBLEM (USE)

Given F a Boolean function and $x, y \in \{0,1\}^n$. There exists an update schedule s such that: $F^s(y) = x$?

BOOLEAN NETWORK PREDECESOR PROBLEM (BNPP)

Given $N = (F, s)$ a Boolean network and $x \in \{0,1\}^n$. There exists $y \in \{0,1\}^n$ such that: $F^s(y) = x$?.

We proved that these problems and some other variants are NP-Hard.

Complexity Problems

UPDATE SCHEDULE EXISTENCE PROBLEM (USE)

Given F a Boolean function and $x, y \in \{0,1\}^n$. There exists an update schedule s such that: $F^s(y) = x$?

BOOLEAN NETWORK PREDECESOR PROBLEM (BNPP)

Given $N = (F, s)$ a Boolean network and $x \in \{0,1\}^n$. There exists $y \in \{0,1\}^n$ such that: $F^s(y) = x$?.

We proved that these problems and some other variants are NP-Hard.

Complexity Problems

UPDATE SCHEDULE EXISTENCE PROBLEM (USE)

Given F a Boolean function and $x, y \in \{0,1\}^n$. There exists an update schedule s such that: $F^s(y) = x$?

BOOLEAN NETWORK PREDECESOR PROBLEM (BNPP)

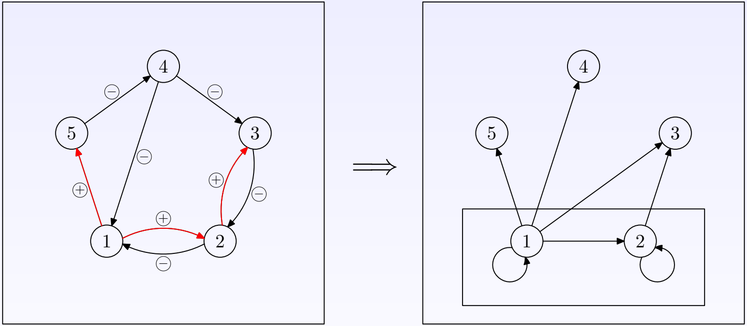
Given $N = (F, s)$ a Boolean network and $x \in \{0,1\}^n$. There exists $y \in \{0,1\}^n$ such that: $F^s(y) = x$?.

We proved that these problems and some other variants are NP-Hard.

Theorem

Let F such that for every i f_i is an AND (OR) function. Then, there exists a sequential update schedule s_q such that $N = (F, s_q)$ has no limit cycles.

Example



Monotonic Boolean Network Conjecture

Let $N' = (F, s')$ be a monotonic Boolean Network. Then, there exists an update schedule s such that $N = (F, s)$ has only fixed points as attractors.

Known Cases

- Networks with all their loops (Mortveit and Reidys, 2007): Any sequential update schedule.
- Symmetric Threshold Networks (Goles, 1987): Any sequential update schedule.
- AND(OR) network.

Monotonic Boolean Network Conjecture

Let $N' = (F, s')$ be a monotonic Boolean Network. Then, there exists an update schedule s such that $N = (F, s)$ has only fixed points as attractors.

Known Cases

- Networks with all their loops (Mortveit and Reidys, 2007): Any sequential update schedule.
- Symmetric Threshold Networks (Goles, 1987): Any sequential update schedule.
- AND(OR) network.

Monotonic Boolean Network Conjecture

Let $N' = (F, s')$ be a monotonic Boolean Network. Then, there exists an update schedule s such that $N = (F, s)$ has only fixed points as attractors.

Known Cases

- Networks with all their loops (Mortveit and Reidys, 2007): Any sequential update schedule.
- Symmetric Threshold Networks (Goles, 1987): Any sequential update schedule.
- AND(OR) network.

¡Feliz Cumpleaños Eric!

Thank you!

References

1. J. Aracena,E. Goles, M. Moreira, S. Salinas. On the robustness of update schedules in Boolean networks. *Biosystems* 97 (2009), 1-8.
2. J. Aracena, L. Gómez, L. Salinas. Limit cycles and update digraphs in Boolean networks. (2011), submitted.
3. J. Aracena, E. Fanchon, M.Montalva and M. Noual. Cominatorics on update digraphs in Boolean networks. *Discrete Applied Mathematics* 159 (2011), 401-409.
4. E. Goles and L. Salinas. Comparison between parallel and serial dynamics of Boolean networks. *Theoretical Computer Science*, 2008 (396), 247-253.
5. S. Kauffman. Metabolic stability and epigenesis in randomly connected nets. *Journal of Theoretical Biology* 22 (1969), 437-467.
6. H. Mortveit, C. Reidys. An introduction to sequential dynamical systems. *Springer Series in Computational Mathematics* (2007).