UNIVERSIDAD DE CONCEPCION

CI²MA Centro de Investigación en Ingeniería Matemática

# Non Primitive Update Digraph problem

Julio Aracena[1] Luis Gómez Lilian Salinas[2]

[1]CI$^2$MA and Departamento de Ingeniería Matemática, Universidad de Concepción.

[2]CI$^2$MA and Departamento de Ingeniería Informática y Ciencias de la Computación, Universidad de Concepción.

April 2016

# Summary

# Boolean Networks

$N = (G^F, F, s)$, Boolean Network.

$G^F = (V, A)$ a digraph.

$V$ a set of $n$ elements.

$F : \{0,1\}^n \to \{0,1\}^n$, global activation function, with interaction digraph $G^F$.

$f_v(x) := F(x)_v$, $\forall v \in V$, local activation functions.

# Boolean Networks

$N = (G^F, F, s)$, Boolean Network.

$G^F = (V, A)$ a digraph.

$V$ a set of $n$ elements.

$F : \{0,1\}^n \to \{0,1\}^n$, global activation function, with interaction digraph $G^F$.

$f_v(x) := F(x)_v$, $\forall v \in V$, local activation functions.

# Boolean Networks

$N = (G^F, F, s)$, Boolean Network.

$G^F = (V, A)$ a digraph.

$V$ a set of $n$ elements.

$F : \{0,1\}^n \rightarrow \{0,1\}^n$, global activation function, with interaction digraph $G^F$.

$f_v(x) := F(x)_v$, $\forall v \in V$, local activation functions.

# Boolean Networks

$N = (G^F, F, s)$, Boolean Network.

$G^F = (V, A)$ a digraph.

$V$ a set of $n$ elements.

$F\colon \{0,1\}^n \to \{0,1\}^n$, global activation function, with interaction digraph $G^F$.

$f_v(x) := F(x)_v$, $\forall v \in V$, local activation functions.

# Boolean Networks

$N = (G^F, F, s)$, Boolean Network.

$G^F = (V, A)$ a digraph.

$V$ a set of $n$ elements.

$F\colon \{0,1\}^n \to \{0,1\}^n$, global activation function, with interaction digraph $G^F$.

$f_v(x) := F(x)_v$, $\forall v \in V$, local activation functions.

# Boolean Networks

$N = (G^F, F, s)$, Boolean Network.

$G^F = (V, A)$ a digraph.

$V$ a set of $n$ elements.

$F\colon \{0,1\}^n \to \{0,1\}^n$, global activation function, with interaction digraph $G^F$.

$f_v(x) := F(x)_v$, $\forall v \in V$, local activation functions.

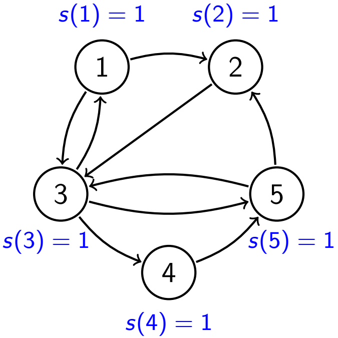
# Update Schedule

$s\colon V \to \{1, \dots, n\}$, function.

# Update Schedule

$s\colon V \to \{1,\ldots,n\}$, function.

$s(V) = \{1\}$, parallel.

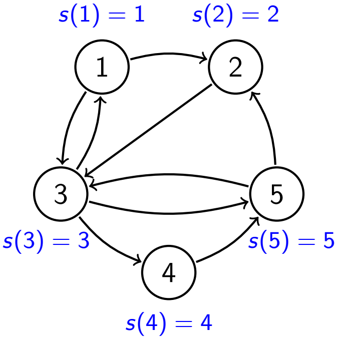


# Update Schedule

$s\colon V \to \{1, \ldots, n\}$, function.

$s(V) = \{1\}$, parallel.

$s(V) = \{1, \ldots, n\}$, sequential.



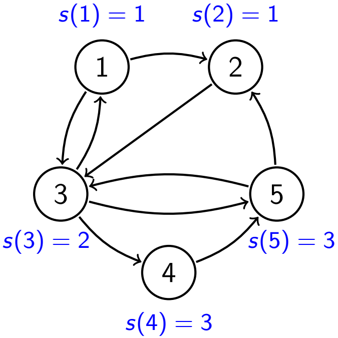
# Update Schedule

$s\colon V \to \{1, \ldots, n\}$, function.

$s(V) = \{1\}$, parallel.

$s(V) = \{1, \ldots, n\}$, sequential.

$s(V) = \{1, \ldots, m\}$, $1 < m < n$, block-sequential.



# Iteration

- Given $x = (x_v)_{v \in V} \in \{0,1\}^n$, the $(k+1)$-iteration of $x$ by $F$ according to $s$ is given by:

$$x_v^{k+1} = f_v(x_u^{l_u} : u \in V)$$

- Where:

$$l_u = \begin{cases} k & \text{if } s(v) \leq s(u) \\ k+1 & \text{if } s(v) > s(u) \end{cases}$$

# Iteration

- Given $x = (x_v)_{v \in V} \in \{0,1\}^n$, the $(k+1)$-iteration of $x$ by $F$ according to $s$ is given by:

$$x_v^{k+1} = f_v(x_u^{l_u} : u \in V)$$

- Where:

$$l_u = \begin{cases} k & \text{if } s(v) \leq s(u) \\ k+1 & \text{if } s(v) > s(u) \end{cases}$$

# Dynamical Behavior

- We can define $f_v^s(x) = f_v(g_{v,u}^s(x) \colon u \in V)$
- Where:

$$g_{v,u}^s(x) = \begin{cases} x_u & \text{if } s(v) \leq s(u) \\ f_u^s(x) & \text{if } s(v) > s(u) \end{cases}$$

- $F^s$ is the dynamical behavior of $N = (F, s)$.

# Dynamical Behavior

- We can define $f_v^s(x) = f_v(g_{v,u}^s(x) \colon u \in V)$
- Where:

$$g_{v,u}^s(x) = \begin{cases} x_u & \text{if } s(v) \leq s(u) \\ f_u^s(x) & \text{if } s(v) > s(u) \end{cases}$$

- $F^s$ is the dynamical behavior of $N = (F, s)$.

# Dynamical Behavior

- We can define $f_v^s(x) = f_v(g_{v,u}^s(x) \colon u \in V)$
- Where:

$$g_{v,u}^s(x) = \begin{cases} x_u & \text{if } s(v) \leq s(u) \\ f_u^s(x) & \text{if } s(v) > s(u) \end{cases}$$

- $F^s$ is the dynamical behavior of $N = (F, s)$.

## Limit Behavior

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

Limit Cycles: $C = \left[x^k\right]_{k=0}^{p}$, $x^k \in \{0,1\}^n$, $p > 1$:

$$x^{k+1} = F^s(x^k) \quad \wedge \quad x^p \equiv x^0$$

$LC(N)$: set of limit cycles of $N$.

# Limit Behavior

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

Limit Cycles: $C = \left[x^k\right]_{k=0}^{p}$, $x^k \in \{0,1\}^n$, $p > 1$:

$$x^{k+1} = F^s(x^k) \quad \wedge \quad x^p \equiv x^0$$

$LC(N)$: set of limit cycles of $N$.

# Update Digraph

Given a Boolean network $(F, s)$, we define the associated labeled digraph $G_s^F = (G^F, \mathrm{lab}_s)$, called *update digraph*, where $\mathrm{lab}_s : A(G^F) \to \{\ominus, \oplus\}$ is defined as:

$$\mathrm{lab}_s(u, v) = \begin{cases} \oplus & \text{if } s(u) \geq s(v) \\ \ominus & \text{if } s(u) < s(v) \end{cases}$$

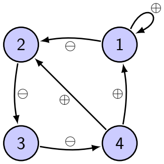
Example: 

$s(i) = i, \quad \forall i \in \{1, \ldots, n\}.$

# Update Digraph

Given a Boolean network $(F, s)$, we define the associated labeled digraph $G^F_s = (G^F, \mathrm{lab}_s)$, called *update digraph*, where $\mathrm{lab}_s : A(G^F) \to \{\ominus, \oplus\}$ is defined as:

$$\mathrm{lab}_s(u, v) = \begin{cases} \oplus & \text{if } s(u) \geq s(v) \\ \ominus & \text{if } s(u) < s(v) \end{cases}$$

Example: 

$s(i) = i, \quad \forall i \in \{1, \ldots, n\}.$

# Update Digraph

## Remark

It was proven in (Salinas, L., 2008. Estudio de modelos discretos: estructura y dinámica, PhD thesis. Universidad de Chile, Santiago, Chile), that if two different updates schedules have the same update digraph, then they also have the same dynamical behavior.

## Equivalence classes

$$s_1 \sim_{G^F} s_2 \iff [s_1]_{G^F} = [s_2]_{G^F} \iff G^F_{s_1} = G^F_{s_2}$$

# Update Digraph

## Remark

It was proven in (Salinas, L., 2008. Estudio de modelos discretos: estructura y dinámica, PhD thesis. Universidad de Chile, Santiago, Chile), that if two different updates schedules have the same update digraph, then they also have the same dynamical behavior.

## Equivalence classes

$$s_1 \sim_{G^F} s_2 \iff [s_1]_{G^F} = [s_2]_{G^F} \iff G^F_{s_1} = G^F_{s_2}$$

# Parallel Operator

Given an update digraph $G_{\mathrm{lab}} = (G, \mathrm{lab})$, with $G = (V, A)$, we define the operator $\mathcal{P}$ as $\mathcal{P}(G_{\mathrm{lab}}) = (V', A')$, where $V' = V$ and $(u, v) \in A'$ if and only if:

I.- $(u, v) \in A$ and $\mathrm{lab}(u, v) = \oplus$ or,

II.- there exists $w \in V$ such that $(w, v) \in A$, $\mathrm{lab}(w, v) = \ominus$ and $(u, w) \in A'$.

Remark: $G^{F^s}$ is a sub-digraph of $\mathcal{P}(G_s^F)$, and in the case of OR functions they are equal.

# Parallel Operator

Given an update digraph $G_{\text{lab}} = (G, \text{lab})$, with $G = (V, A)$, we define the operator $\mathcal{P}$ as $\mathcal{P}(G_{\text{lab}}) = (V', A')$, where $V' = V$ and $(u, v) \in A'$ if and only if:

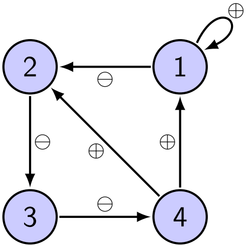
I.- $(u, v) \in A$ and $\text{lab}(u, v) = \oplus$ or,

II.- there exists $w \in V$ such that $(w, v) \in A$, $\text{lab}(w, v) = \ominus$ and $(u, w) \in A'$.

Remark: $G^{F^s}$ is a sub-digraph of $\mathcal{P}(G^F_s)$, and in the case of OR functions they are equal.

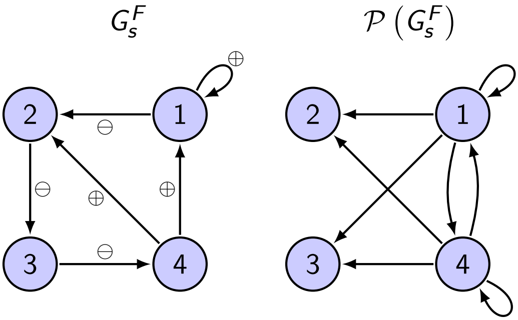
# Example

$G_s^F$



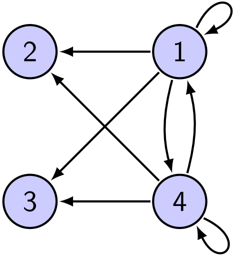

$$
\begin{aligned}
f_1(x) &= x_1 \wedge x_4 \\
f_2(x) &= x_1 \vee x_4 \\
f_3(x) &= x_2 \\
f_4(x) &= x_3 \\
s &= \{1\}\,\{2\}\,\{3\}\,\{4\}
\end{aligned}
$$

# Example

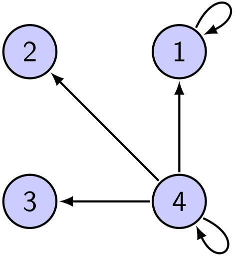


# Example

$\mathcal{P}\left(G_s^F\right)$




$G^{F^s}$



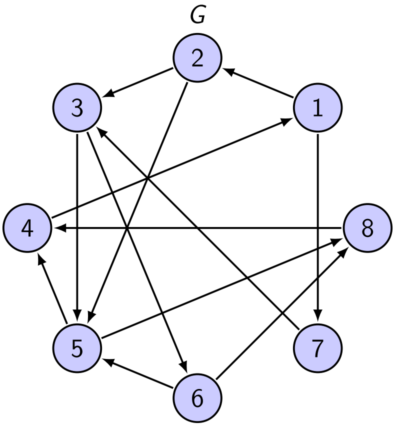

$$f_1^s(x) = x_1 \wedge x_4$$
$$f_2^s(x) = x_4$$
$$f_3^s(x) = x_4$$
$$f_4^s(x) = x_4$$

# NPUD

**Non-Primitive Update Digraph problem (NPUD)**

Given $G$ a digraph. Does there exists an integer $2 \leq k \leq g(G)$ and lab: $A(G) \rightarrow \{\oplus, \ominus\}$ such that $G_{\text{lab}}$ is an update digraph and each cycle in $G_{lab}$ has a multiple of $k$ $\oplus$-labeled arcs?

# Example



# Summary

# LCE

## LIMIT CYCLE EXISTENCE PROBLEM (LCE)

Given $F : \{0,1\}^n \to \{0,1\}^n$. Does there exist an update schedule $s$ such that $LC(F,s) \neq \emptyset$?

## OR LCE

Given $F : \{0,1\}^n \to \{0,1\}^n$ an OR function. Does there exist an update schedule $s$ such that $LC(F,s) \neq \emptyset$?

# LCE

## LIMIT CYCLE EXISTENCE PROBLEM (LCE)

Given $F : \{0,1\}^n \to \{0,1\}^n$. Does there exist an update schedule $s$ such that $LC(F,s) \neq \emptyset$?

## OR LCE

Given $F : \{0,1\}^n \to \{0,1\}^n$ an OR function. Does there exist an update schedule $s$ such that $LC(F,s) \neq \emptyset$?

# LCE

**Theorem**

AND-OR LCE is NP-hard.

**Theorem**

SYMMETRIC LCE is NP-Hard.

**Proposition**

SYMMETRIC AND-OR LCE is polynomial.

# LCE

**Theorem**

AND-OR LCE is NP-hard.

**Theorem**

SYMMETRIC LCE is NP-Hard.

**Proposition**

SYMMETRIC AND-OR LCE is polynomial.

# LCE

**Theorem**

AND-OR LCE is NP-hard.

**Theorem**

SYMMETRIC LCE is NP-Hard.

**Proposition**

SYMMETRIC AND-OR LCE is polynomial.

# Summary

# Known

**Theorem (Jarrah et al. (2010))**

Let $F$ be an OR function and let us consider $N = (F, s^p)$, where $s^p$ is the parallel update schedule.

1. If $G^F$ is strongly connected then:
$$LC(N) = \emptyset \text{ if and only if } G^F \text{ is primitive.}$$
2. If $G^F$ is not strongly connected then:
$$LC(N) = \emptyset \text{ if and only if } G^F \text{ is primitive or it does not have cycles.}$$

Definition: $G^F$ is primitive if and only if $\rho\left(G^F\right) = 1$, where $\rho\left(G^F\right)$ is defined as the greatest common divisor of the lengths of the cycles of $G$.

# Known

**Theorem (Jarrah et al. (2010))**

Let $F$ be an OR function and let us consider $N = (F, s^p)$, where $s^p$ is the parallel update schedule.

1. If $G^F$ is strongly connected then:
$$LC(N) = \emptyset \text{ if and only if } G^F \text{ is primitive.}$$
2. If $G^F$ is not strongly connected then:
$$LC(N) = \emptyset \text{ if and only if } G^F \text{ is primitive or it does not have cycles.}$$

Definition: $G^F$ is primitive if and only if $\rho\left(G^F\right) = 1$, where $\rho\left(G^F\right)$ is defined as the greatest common divisor of the lengths of the cycles of $G$.

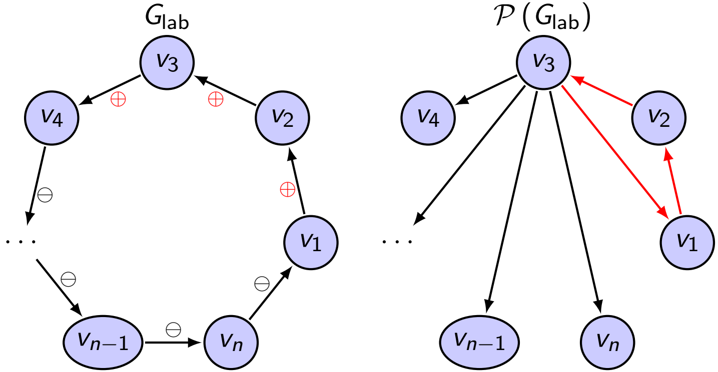
# OR LCE

**Lemma**

Given an update digraph $G_{\text{lab}}$. Then, every circuit in $G_{\text{lab}}$ produce a circuit in $\mathcal{P}(G_{\text{lab}})$ with length the number of the $\oplus$-labeled arcs of it. Conversely, every circuit in $\mathcal{P}(G_{\text{lab}})$ comes from a circuit in $G_{\text{lab}}$ with a number of $\oplus$-labeled arcs equal to the length of the cycle.

# OR LCE

Proof: (Idea)



# OR LCE

## Proposition

Let $N = (F, s)$ be an OR network with strongly connected $G^F$. Then, $LC(N) \neq \emptyset$ if and only if there exists a number $2 \leq k \leq g(G^F)$, such that each cycle in $G_s^F$ has a multiple of $k$ $\oplus$-labeled arcs.

Remark: the condition each cycle in $G_s^F$ has a multiple of $k$ $\oplus$-labeled arcs is equivalent to $\rho(\mathcal{P}(G_s^F)) = k$

# OR LCE

**Proposition**

Let $N = (F, s)$ be an OR network with strongly connected $G^F$. Then, $LC(N) \neq \emptyset$ if and only if there exists a number $2 \leq k \leq g(G^F)$, such that each cycle in $G_s^F$ has a multiple of $k$ $\oplus$-labeled arcs.

Remark: the condition each cycle in $G_s^F$ has a multiple of $k$ $\oplus$-labeled arcs is equivalent to $\rho(\mathcal{P}(G_s^F)) = k$

# Summary

# NPUD

**Non-Primitive Update Digraph problem (NPUD)**

Given $G$ a digraph. Does there exists an integer $2 \leq k \leq g(G)$ and $\mathrm{lab}\colon A(G) \to \{\oplus, \ominus\}$ such that $G_{\mathrm{lab}}$ is an update digraph and each cycle in $G_{lab}$ has a multiple of $k$ $\oplus$-labeled arcs?

# NPUD

**Theorem**

OR LCE and NPUD are equivalent problems in strongly connected digraphs.

Remark: In non strongly connected digraphs, it is sufficient to solve NPUD in a sole strongly connected component to have a solution for OR LCE.

# NPUD

**Theorem**

OR LCE and NPUD are equivalent problems in strongly connected digraphs.

Remark: In non strongly connected digraphs, it is sufficient to solve NPUD in a sole strongly connected component to have a solution for OR LCE.

# NPUD

**Remark**

Each condition of the problem:

- $G_{\mathrm{lab}}$ is an update digraph,
- each cycle in $G_{lab}$ has a multiple of $k$ $\oplus$-labeled arcs,

can be solved independently in polynomial time.

# NPUD

**Lemma**

If $G$ is a non strongly connected digraph, then NPUD has a solution in $G$ if and only if it has a solution in every strongly connected component of $G$.

# NPUD

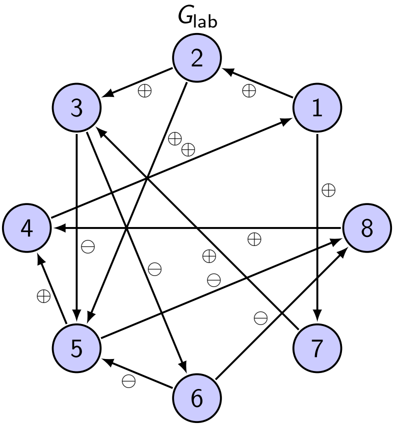
## Proposition

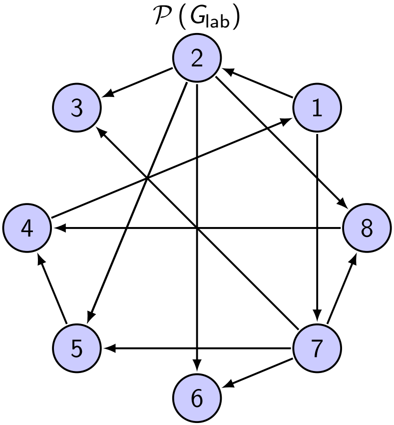
Let $G_{\text{lab}}$ be an update digraph with $G$ strongly connected. Then, $\rho\left(\mathcal{P}\left(G_{\text{lab}}\right)\right) = k > 1$ if and only if there exists a partition of $V$, $\{V_1, \ldots, V_k\}$, such that:

I) For each $(u, v) \in A(G), u \in V_i, v \in V_j$: $\text{lab}(u, v) = \ominus \implies i = j$.

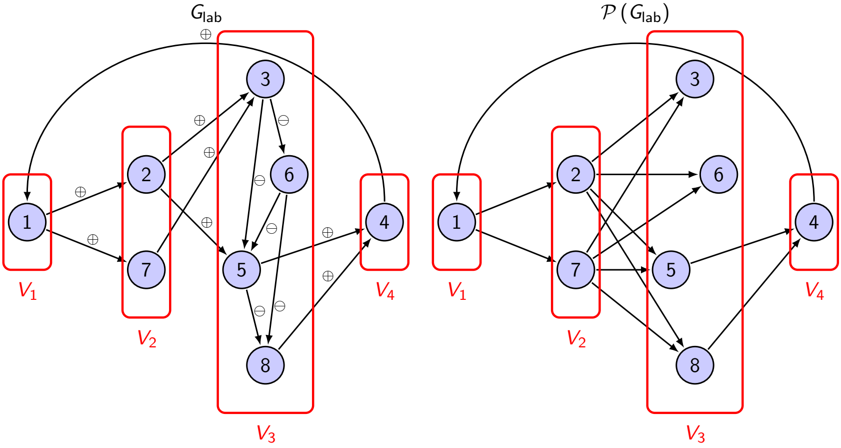
II) For each $(u, v) \in A(G)$ such that $\text{lab}(u, v) = \oplus$, then there exists $i \in \{1, \ldots, k\}$ such that $u \in V_i$ and $v \in V_{i+1}$.

## Example



# Example




Bang-Jensen and Gutin (2007)

# NPUD

## Corollary

Let be $G$ an strongly connected digraph. Then, NPUD has a solution in $G$ if and only if there exist a partition $V_1$, $V_2$ of $V$ such that $G_{\text{lab}}$ is an update digraph, where the label function lab is defined as:

$$\forall (u,v) \in A(G) \colon \ \text{lab}(u,v) = \begin{cases} \ominus & \text{if } u, v \in V_i, i \in \{1,2\} \\ \oplus & \text{if } u \in V_i, v \in V \setminus V_i, i \in \{1,2\} \end{cases}$$

Remark: If such a partition exists, then $G[V_1]$ and $G[V_2]$ are acyclic. Therefore, $V_1$ and $V_2$ form an acyclic 2-partitioning of $V$ (Nobibon et al., (2010)).

# NPUD

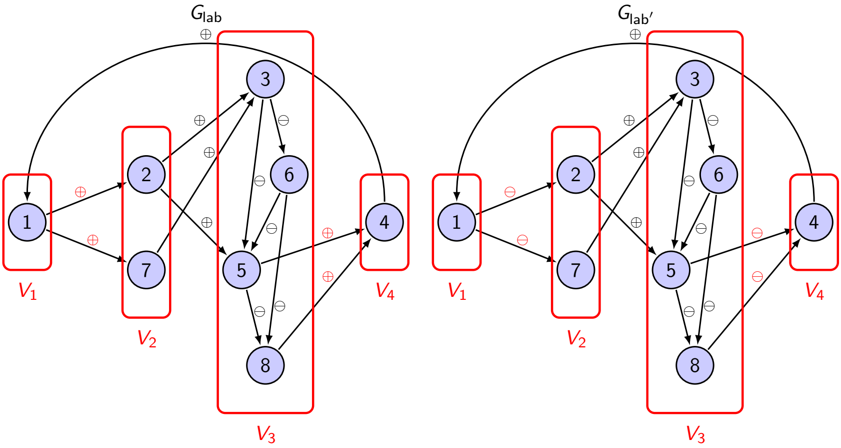
## Corollary

Let be $G$ an strongly connected digraph. Then, NPUD has a solution in $G$ if and only if there exist a partition $V_1$, $V_2$ of $V$ such that $G_{\mathrm{lab}}$ is an update digraph, where the label function lab is defined as:

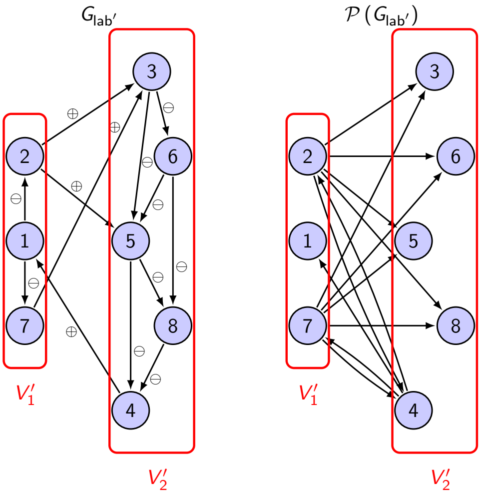
$$\forall (u,v) \in A(G)\colon\ \mathrm{lab}(u,v) = \begin{cases} \ominus & \text{if } u, v \in V_i, i \in \{1,2\} \\ \oplus & \text{if } u \in V_i, v \in V \setminus V_i, i \in \{1,2\} \end{cases}$$

Remark: If such a partition exists, then $G[V_1]$ and $G[V_2]$ are acyclic. Therefore, $V_1$ and $V_2$ form an acyclic 2-partitioning of $V$ (Nobibon et al., (2010)).

# Example 5



# Example 5



# NPUD

- Algorithm that solves NPUD (2-partition version, non polynomial).
- Forbidden configuration approach.

# NPUD

- Algorithm that solves NPUD (2-partition version, non polynomial).
- Forbidden configuration approach.

# NPUD

- Algorithm that solves NPUD (2-partition version, non polynomial).
- Forbidden configuration approach.

# NPUD

**Open problem**

Complexity.

# Bibliography I


- S.A. Kauffman, *Metabolic stability and epigenesis in randomly connected nets.* Journal of Theoretical Biology, Volume 22, Pages 437-467, 1969.
- J. Aracena, E. Goles, A. Moreira, L. Salinas, *On the robustness of update schedules in Boolean networks.* Biosystems, Volume 97, Pages 1-8, 2009.
- J. Aracena, L. Gómez, L. Salinas, *Limit cycles and update digraphs in Boolean networks.* Discrete Applied Mathematics, Volume 161, Issues 1-2, Pages 1-12, 2013.
- A. Jarrah, R. Laubenbacher, A. Veliz-Cuba, *The dynamics of conjunctive and disjunctive Boolean networks.* Bulletin of Mathematical Biology, Volume 72, Pages 1425-1447, 2010.

# Bibliography II


- E. Goles, M. Noual, *Disjunctive networks and update schedules.* Advances in Applied Mathematics, Volume 48, Pages 646-662, 2012.
- J. Bang-Jensen, G. Gutin, *Digraphs Theory, Algorithms and Applications.* Springer-Verlag, Berlin, 2007.
- F. Nobibon, C. Hurkens, R. Leus, F, Spieksma, *Exact algorithms for coloring graphs while avoiding monochromatic cycles.* Proceeding 6th Internat. Conference on Algorithmic Aspects in Information and Management, Pages 229-242.

# The End

Thanks

Thanks

Thanks

Thanks

# The End

Thanks

Thanks

Thanks

Thanks

# The End

Thanks

Thanks

Thanks

Thanks

# The End

Thanks

Thanks

Thanks

Thanks

# The End

Thanks

Thanks

Thanks

Thanks

## The End

Thanks

Thanks

Thanks

Thanks

# The End

Thanks

Thanks

Thanks

Thanks

# The End

Thanks

Thanks

Thanks

Thanks

# The End

Thanks

Thanks

Thanks

Thanks

# The End

Thanks

Thanks

Thanks

Thanks

# The End

Thanks

Thanks

Thanks

Thanks

# The End

Thanks

Thanks

Thanks

Thanks

# The End

Thanks

Thanks

Thanks

Thanks

# The End

Thanks

Thanks

Thanks

Thanks

# The End

Thanks

Thanks

Thanks

Thanks

# The End

Thanks

Thanks

Thanks

Thanks

# The End

Thanks

Thanks

Thanks

Thanks

# The End

Thanks

Thanks

Thanks

Thanks

# The End

Thanks

Thanks

Thanks

Thanks

# The End

Thanks

Thanks

Thanks

Thanks

# The End

Thanks

Thanks

Thanks

Thanks

# The End

Thanks

Thanks

Thanks

Thanks

# The End

Thanks

Thanks

Thanks

Thanks

# The End

Thanks

Thanks

Thanks

Thanks

# The End

Thanks

Thanks

Thanks

Thanks

# The End

Thanks

Thanks

Thanks

Thanks

# The End

Thanks

Thanks

Thanks

Thanks

# The End

Thanks

Thanks

Thanks

Thanks

# The End

Thanks

Thanks

Thanks

Thanks

# The End

Thanks

Thanks

Thanks

Thanks