UNIVERSIDAD DE CONCEPCIÓN

Centro de Investigación en
Ingeniería Matemática (CI²MA)

Combinatorics on update digraphs of discrete networks

Julio Aracena, Eric Fanchon,
Marco Montalva, Mathilde Noual

PREPRINT 2009-24

SERIE DE PRE-PUBLICACIONES

Combinatorics on update digraphs of discrete networks

J. Aracena [a,*] E. Fanchon [c] M. Montalva [a] M. Noual [b]

[a] *CI^{2}MA and Departamento de Ingeniería Matemática, Universidad de Concepción,*
Av. Esteban Iturra s/n, Casilla 160-C, Concepción, Chile.

[b] *Laboratoire d'Informatique du Parallélisme de L'ENS Lyon,*
46, allée d'Italie, 69364 Lyon Cedex 07, France.

[c] *TIMC-IMAG Laboratory, TIMB Team, Faculté de Médecine,*
38706 La Tronche cedex, France.

Abstract

Discrete networks have been used as models of gene regulation and other biological networks. One key element in these models is the update schedule, which indicates the order in which states have to be updated. In Aracena et al. (2009) was defined equivalence classes of deterministic update schedules according to the labeled digraph associated to the network (update digraph) and such that two schedules in the same class yield the same dynamical behavior. In this paper we study algorithmical and combinatorial aspects of update digraphs. We show a polinomial characterization of these digraphs, which enables to characterize the corresponding equivalence classes. We prove that the update digraphs are exactly the projections, on the respective subgraphs, of a complete update digraph. Finally, the exact number of complete update digraphs was determined, which provides upper and lower bounds on the number of equivalence classes.

Key words: Discrete network, update schedule, update digraph, feedback arc set.

* Corresponding author.
Email addresses: `jaracena@ing-mat.udec.cl` (J. Aracena), `Eric.Fanchon@imag.fr` (E. Fanchon), `mmontalv@udec.cl` (M. Montalva), `Mathilde.Noual@ens-lyon.fr` (M. Noual).

Preprint submitted to

1 Introduction

Discrete networks (DNs) are the most simple model for genetic regulatory networks, as well as for other simple distributed dynamical systems. Despite their simplicity, they provide a realistic model in which different phenomena can be reproduced and studied, and indeed, many regulatory models published in the biological literature fit within their framework Kauffman (1969); Thomas (1973); Shmulevich et al. (2003).

A DN is defined by the states set of the nodes of the network, its connection digraph, its local activation functions, and the type of update schedule used, which may range from the parallel update, the most common Kauffman (1969); Thomas (1991), to the sequential update, passing through all the combinations of block-sequential updates (which are sequential over the sets of a partition, but parallel inside of each set).

The effect in the dynamical behavior of of a discrete network against perturbations in the update schedule has been greatly studied, mainly from a statistical point of view, in random Boolean networks(RBN), where the local activation functions are probabilistically chosen Chaves et al. (2005).

Some analytical works about perturbations of update schedules have been made in a particular class of discrete dynamical networks, called sequential dynamical systems, where the connection digraph is symmetric or equivalently an undirected graph and the update schedule is sequential. For this class of networks, the team of Barrett, Mortveit and Reidys studied the set of sequential update schedules preserving the whole dynamical behavior of the network (2001) and the set of attractors in a certain class of Cellular Automata (2005).

In Aracena et al. (2009) was defined equivalence classes of deterministic update schedules in a particular kind of discrete networks (Boolean networks) according to the labeled digraph associated to the network (update digraph). It was proven that two update schedules in the same classe yield exactly the same dynamical behavior.

In this paper we focus on the study of the update digraphs and the number and size of equivalence classes of update schedules associated in a discrete network.

The main reason for our interest in update digraphs schedules is two-fold. On one hand, a necessary and good understanding of the objects we are dealing with. On the other hand, a better understanding of the relationships between the architecture of connection graph and the robustness of discrete network dynamics through the study of the equivalence classes of determinsitic update schedules definided by its aasociated updated digraphs.

2 Definitions

A *digraph* is an ordered pair of sets $G = (V, A)$ where $V = \{1, \dots, n\}$ is a set of elements called **vertices** (or **nodes**) and A is a set of ordered pairs (called **arcs**) of vertices of V. The vertex set of G is referred to as $V(G)$, its arc set as $A(G)$.

A **walk** from a vertex v_1 to a vertex v_m in a digraph G is a sequence of vertices $v_1, v_2, \dots, v_m$ of $V(G)$ such that $\forall k = 1, \dots, m-1$, $(v_k, v_{k+1}) \in A(G)$ or $(v_{k+1}, v_k) \in A(G)$. The vertices v_1 and v_m are the initial and terminal vertex of the walk. A walk is **elementary** if each vertex in the walk appears only once with the possible exception that the first and last vertex may coincide. A walk is closed if its initial and terminal vertices coincide. A **circuit** is a closed elementary walk. A walk $v_1, v_2 \dots, v_m$ is a **path** if $(v_k, v_{k+1}) \in A(G)$ for all $k = 1, \dots, m-1$. A **cycle** is a directed circuit, that is a closed elementary path.

A digraph G is said to be **connected** if there is a walk between every pair of its vertices, and **strongly connected** if there is a path between every pair of its vertices.

$G = (V, A)$ being a digraph and $i \in V$ one of its vertices, $N(i) = \{j \in V \mid (j, i) \in A\}$ denotes the input neighborhood of i and $d^-(i) = |N(i)|$ is the input degree of i. More terminology about digraph can be found in (West, 1996).

Also, in the sequel, we will write $[\![a, b]\!] = \{a, \dots, b\}$ and $[\![a, b[\![= \{a, \dots, b-1\}$, for any integers a and b.

Definition 1 *An* **update schedule** *of the vertices of a digraph $G = (V, A)$, with $|V| = n$, is a function $s : V \to \{1, \dots, n\}$ such that $s(V) = [\![1, m]\!]$ for some $m \leq n$. If $\forall i \in V,\ s(i) = 1$, the update schedule is said to be* **parallel**. *In this case, we will write $s = s_p$. If s is a permutation over the set $\{1, \dots, n\}$, s is said to be* **sequential**. *And in all other cases, s is said to be* **block sequential**.

As mentionned in Demongeot et al. (2008), the number of update schedules associated to a given digraph of order n is equal to the number of ordered partitions of a set of size n, that is

$$T_n = \sum_{k=0}^{n-1} \binom{n}{k} T_k.$$

Let $G = (V, A)$ be a digraph and s an update schedule. We denote $s^{-1}(r) = \{i \in V \mid s(i) = r\}$.

Definition 2 *Let $G = (V, A)$ be a digraph and s an update schedule, we define the labeling function $lab_s : A \to \{\ominus, \oplus\}$ in the following way :*

$$\forall (j,i) \in A, lab_s(j,i) = \begin{cases} \oplus & \text{if } s(j) \geq s(i) \\ \ominus & \text{if } s(j) < s(i). \end{cases}$$

An arc $a \in A$ such that $lab_s(a) = \oplus$ is called a **positive arc** *and an arc $a \in A$ such that $lab_s(a) = \ominus$ is called a* **negative arc**. *Labeling every arc a of A by $lab_s(a)$, we obtain a labeled digraph (G, lab_s) named* **update digraph**. *We write $N_s^+(i) = \{j \in N(i) \mid lab_s(j,i) = \oplus\}$ and $N_s^-(i) = \{j \in N(i) \mid lab_s(j,i) = \ominus\}$. Thus, we have $N(i) = N_s^+(i) \cup N_s^-(i)$.*

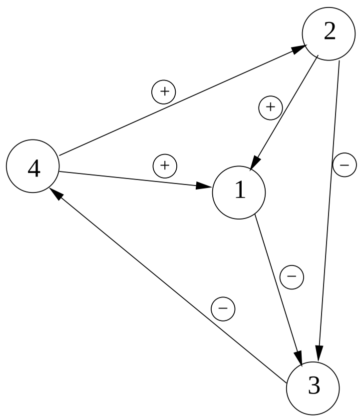


Fig. 1. A digraph $G = (V, A)$ labeled by the function lab_s where $\forall i \in V = \{1, \ldots, 4\}$, $s(i) = i$.

Definition 3 *A* **discrete network** *$N = (G, F, s)$ is defined by a finite set of variable states $x \in Q = [\![0, m-1]\!]$, a digraph G, a global activation function $F : Q^n \to Q^n$, where $F(x) = (f_1(x), \ldots, f_n(x))$ with $f_i : Q^n \to Q$* **local activation functions** *such that $j \in N(i)$ if and only if $\exists\, (x_1, \ldots, x_n) \in Q^n, a, b \in Q$, $a \neq b$,*

$$f_i(x_1, \ldots, x_{j-1}, a, x_{j+1}, \ldots, x_n) \neq f_i(x_1, \ldots, x_{j-1}, b, x_{j+1}, \ldots, x_n).$$

and s an update schedule of the vertices of G. In the particular case where $Q = \{0, 1\}$ the network is said to be Boolean network.

The iteration of the discrete network with an update schedule s is given by:

$$x_i^{r+1} = f_i(x_1^{l_1}, \ldots, x_n^{l_n}), \tag{1}$$

where $l_j = r$ if $s(i) \leq s(j)$ and $l_j = r + 1$ if $s(i) > s(j)$.

This is equivalent to applying a function $F^s : Q^n \to Q^n$ in a parallel way, with $F^s(x) = (f_1^s(x), \ldots, f_n^s(x))$ defined by:

$$f_i^s(x) = f_i(g_{i,1}^s(x), \ldots, g_{i,n}^s(x)),$$

where the function $g_{i,j}^s$ is defined by $g_{i,j}^s(x) = x_j$ if $s(i) \leq s(j)$ and $g_{i,j}^s(x) = f_j^s(x)$ if $s(i) > s(j)$. Thus, the function F^s corresponds to the dynamical behavior of the network N. We will say that two networks $N_1 = (G, F, s_1)$ and $N_2 = (G, F, s_2)$ have the same dynamics if $F^{s_1} = F^{s_2}$.

3 Preliminary results and motivations

Extending a result given in Aracena et al. (2009) for Boolean networks, the following holds:

Theorem 4 *Let $N_1 = (G, F, s_1)$ and $N_2 = (G, F, s_2)$ be two discrete networks that differ only in the update schedule. If $(G, lab_{s_1}) = (G, lab_{s_2})$, then N_1 and N_2 have the same dynamics.*

Theorem 4 allows us to define *equivalence classes with respect to labeled digraphs*: if s is an update schedule of the vertices of a digraph G, we write $[s]_G$ the set of update schedules s' such that $s \overset{G}{\sim} s'$, that is

$$[s]_G = \{s' : \ (G, lab_s) = (G, lab_{s'})\}.$$

Thus, an equivalence class, $[s]_G$, is a set of update schedules that all yield the same labeled digraph, and consequently, the same dynamics on networks.

In this work we study update digraphs and the equivalence classes of their update schedules. More precisely, Section 4 deals with the characterization of update digraphs. Sections 5 and 6 focus on the size and the number of equivalence classes of update schedules.

4 Characterization of update digraphs

In this section, we study the relation $\sim_G$ and the labelings of a given digraph G. First, we give a characterization of the labeling functions $lab : A(G) \to \{\oplus, \ominus\}$ that indeed correspond to labeling functions induced by update schedules. Then, we examine update schedules s which satisfy $lab = lab_s$. The section ends with some observations that where made to help determine the number of $[\cdot]_G$ classes. First, let us give some additional definitions.

Definition 5 *A labeled digraph (G, lab) is said to be an* **update digraph** *(UD) if there exists an update schedule s such that $lab = lab_s$, that is $\forall a \in A(G)$, $lab(a) = lab_s(a)$. (see example in figure 4)*

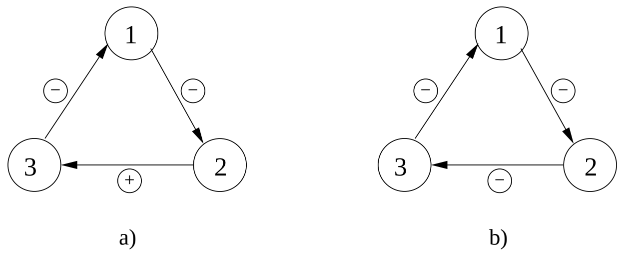


Fig. 2. a) A labeled digraph (G, lab) which is update digraph. b) A labeled digraph (G, lab') which is not an update digraph.

The goal of this section is to determine which labeled digraphs are update digraphs.

Definition 6 *Let (G, lab) be a labeled digraph and G' a subdigraph of G. We define the projection of (G, lab) onto G' by the labeled digraph $(G', lab_{G'})$, where $lab_{G'}(a) = lab(a)$, $\forall a \in A(G')$.*

Definition 7 *Let (G, lab) be a labeled digraph and G' be a non trivial strongly connected subdigraph of G, the projected labeled digraph (G', lab') is said to be a positive strongly connected component of (G, lab) if $\forall a \in A(G')$, $lab_{G'} = \oplus$ and it is maximal for this property. We will say that (G, lab) is reduced if it has no positive strongly connected components.*

Nothe that the fact of being (G, lab) an update digraph is independent of the presence or absence of positive strongly connected components, because the images of the vertices by an update schedule in a positive strongly connect component are identical. In the sequel and without loss of generality, we will work only with reduced labeled digraphs.

Definition 8 *Let (G, lab) be a labeled digraph. We define the* **labeled reoriented digraph** *associated to (G, lab), and write (G_R, lab_R), to refer to the labeled digraph in which all negative arcs are inverted:*

- $V(G_R) = V(G)$.
- $A(G_R) = \{(u, v) \in A(G) \mid lab(u, v) = \oplus\} \cup \{(v, u) \mid (u, v) \in A(G) \land lab(u, v) = \ominus\}$
- $lab_R(u, v) = \ominus$ *if* $(v, u) \in A(G) \land lab(v, u) = \ominus$ *and* $lab_R(u, v) = \oplus$ *if* $lab(u, v) = \oplus$.

A forbidden cycle in (G_R, lab_R) is a cycle containing a negative arc.

Let (G, lab) be a labeled digraph. We can determine if it is reduced in time $\mathcal{O}(|A|)$ with an algorithm that searches for strongly connected components of a digraph. We also can get (G_R, lab_R) in time $\mathcal{O}(|A|)$.

Definition 9 *Let (G, lab) be a labeled digraph and P a path in (G_R, lab_R), we denote by $l^-(P)$ the number of negative arcs of P. Thus, for every $v \in V(G)$ we define $L^-(v) = \max\{l^-(P_v) \mid P_v$ is a path in (G_R, lab_R) ending in $v\}$ and*

$$L^-(G_R, lab_R) = \max_{v \in V(G)} \{L^-(v)\},$$

the number of negative arcs of a path with the maximum number of negative arcs in (G_R, lab_R).

Theorem 10 *A labeled digraph (G, lab) is an update digraph if and only if (G_R, lab_R) does not contain any forbidden cycle.*

PROOF. $(\Rightarrow)$ Let us suppose that (G_R, lab_R) contains a forbidden cycle $C : v_1, \ldots, v_p = v_1$ such that (v_j, v_{j+1}) is a negative arc. Then any update schedule s such that $(G, lab) = (G, lab_s)$ must satisfy $s(v_j) > s(v_{j+1})$. It must also satisfy $s(v_j) \leq s(v_{j+1})$ since there exists in (G_R, lab_R) a path from v_{j+1} to v_j. Thus, we end up with a contradiction.

$(\Leftarrow)$ Let $L = L^-(G_R, lab_R)$. Observe first that if $P : v_1, \ldots, v_k$ is a path in G such that $l^-(P) = L$ with $\{(v_{i_1}, v_{i_2}), (v_{i_3}, v_{i_4}), \ldots, (v_{i_{2L-1}}, v_{i_{2L}})\}$ the set of negative arcs of P where $j > k \Rightarrow i_j > i_k$, and s is an update schedule such that $(G, lab) = (G, lab_s)$ then

$$s(v_{i_1}) > s(v_{i_2}) > s(v_{i_4}) > s(v_{i_6}) > \cdots > s(v_{i_{2L}}),$$

which implies $\max\{s(v) | \ v \in V(G)\} \geq L + 1$. Besides,

$$\forall\, i = 1, \ldots, k, \ L^-(v_i) = l^-(v_1, \ldots, v_i) \quad \text{and} \quad L^-(v_1) = 0.$$

Let $s : V(G) \to [\![1, L+1]\!]$ with

$$s(v) = L - L^-(v) + 1, \ \forall v \in V(G).$$

Since obervation mentioned above, $s(V(G)) = [\![1, L+1]\!]$, that is, s is an update schedule of $V(G)$. To check that s is also an update schedule satisfying $(G, lab) = (G, lab_s)$, we must show that $\forall a = (u, v) \in A(G_R), \ s(u) > s(v)$ or $L(v) > L(u)$. This follows from the fact that (u, v) being an arc of G_R, it necessarily holds that $L(v) \geq 1 + L(u)$. □

We can notice that if (G, lab) is a labeled digraph, the forbidden cycles of (G_R, lab_R) correspond to what we will refer to as alternating circuits of G. That is, they coincide with walks of G, $C = v_0, v_1, \ldots, v_k$, where $v_0 = v_k$ and either $(v_i, v_{i+1}) \in A$ in which case $lab_G(v_i, v_{i+1}) = \oplus$ or $(v_{i+1}, v_i) \in A$ in which case $lab_G(v_{i+1}, v_i) = \ominus$ (or vice-versa). Among these alternating circuits, are in particular circuits such that $\forall i \in [\![0, k-1]\!]$, $lab(v_i, v_{i+1}) = \ominus$ as well as sub-graphs containing two vertices u and v, a walk from u to v negatively labeled and another walk from u to v positively signed.

Incidently, let us notice that, as a consequence of Theorem 10, if $a = (u, v) \in A(G)$ is an arc not belonging to any circuit, then the fact that (G, lab) is an update digraph or not is independent of $lab(a)$.

Algorithm 1, given below, finds an update schedule corresponding to a given reduced labeled digraph as described in the proof of Theorem 10. It is adapted from the famous algorithm van Leeuwen (1990), giving a topological order on a digraph without cycles. For a given reduced labeled digraph (G, lab), algorithm 1 works on the labeled reoriented digraph (G_R, lab_R) without forbidden cycles. It returns in time $\mathcal{O}(|V| + |A|)$ an update schedule s such that $(G, lab) = (G, lab_s)$ and

$$max\{s(v) \mid v \in V\} = min\{max\{s'(v) \mid v \in V\} \mid s' \text{ is an update schedule of } G\}.$$

Figure 4 shows the different steps of the algorithm that returns an update schedule associated to an arbitrary possible labeled digraph (not necessarily reduced).

Corollary 11 *The following problems can be solved in polynomial time.*

(1) Determine whether a labeled digraph (G, lab) is an update digraph,

(2) Given (G, lab) a labeled digraph, find an update schedule s such that $(G, lab) = (G, lab_s)$.

Indeed, according to Theorem 10, a labeled digraph (G, lab) is an update one if and only if, in (G_R, lab_R) no negative-arc belongs to a strongly connected component. Thus, the first part of Corollary 11 holds since the strongly connected components of a digraph can be identified in polynomial time. The second part of Theorem 10 comes from the existence algorithm 1 whose run time is also polynomial.

5 Sizes of the equivalence classes $[\cdot]_G$

In this section we study the size of the equivalence classes $[\cdot]_G$.

Algorithm 1. `update schedule associated to a labeled digraph`
Input: $(G = (V, A), lab)$ a reduced labeled digraph such that (G_R, lab_R) has no forbidden cycle

```
begin
    ValMax ← table of size |V(G_R)| in which are stocked the maximal possible
    values of s(v), v ∈ V(G_R).
    n ← |V|;
    H ← G_R;
    forall v ∈ V do
        ValMax[v] = n;
    end

    while ∃v ∈ V, d⁻(v) = 0 in H do
        s(v) ← ValMax[v];
        forall u ∈ N(v) do
            if (u, v) is a negative arc then
                ValMax[u] ← min{ValMax[u], s(v) − 1; }
            else
                ValMax[u] ← min{ValMax[u], s(v); }
            end
            delete the arc (u, v) from H
        end
    end

    s_min ← min{s(v | v ∈ V); }
    forall v ∈ V do
        s(v) ← s(v) − s_min;
    end
end
```

Let us now consider the following question : given a digraph G and an update schedule s, does there exist any udpdate schedule $s' \neq s$ such that $(G, lab_s) = (G, lab_{s'})$? That is, what conditions need to be satisfied in order for $|[s]_G| > 1$ to hold?

Corollary 12 *Let (G, lab) be a reduced update digraph with $|V(G)| = n$ and $L = L^-(G_R, lab_R)$. Then, $\forall m \in [\![L, n-1]\!]$, there exists an update schedule s such that* $\max\{s(v)|\ v \in V\} = m + 1$ *and $(G, lab) = (G, lab_s)$.*

PROOF. We show the result by induction on m.

If $m = L$, the result was proved in Theorem 10.

If $L = n - 1$, we are done. Otherwise, let $m \in]\!]L, n-1]\!]$. By induction hypothesis, there exists an update schedule s such that $(G, lab) = (G, lab_s)$ and

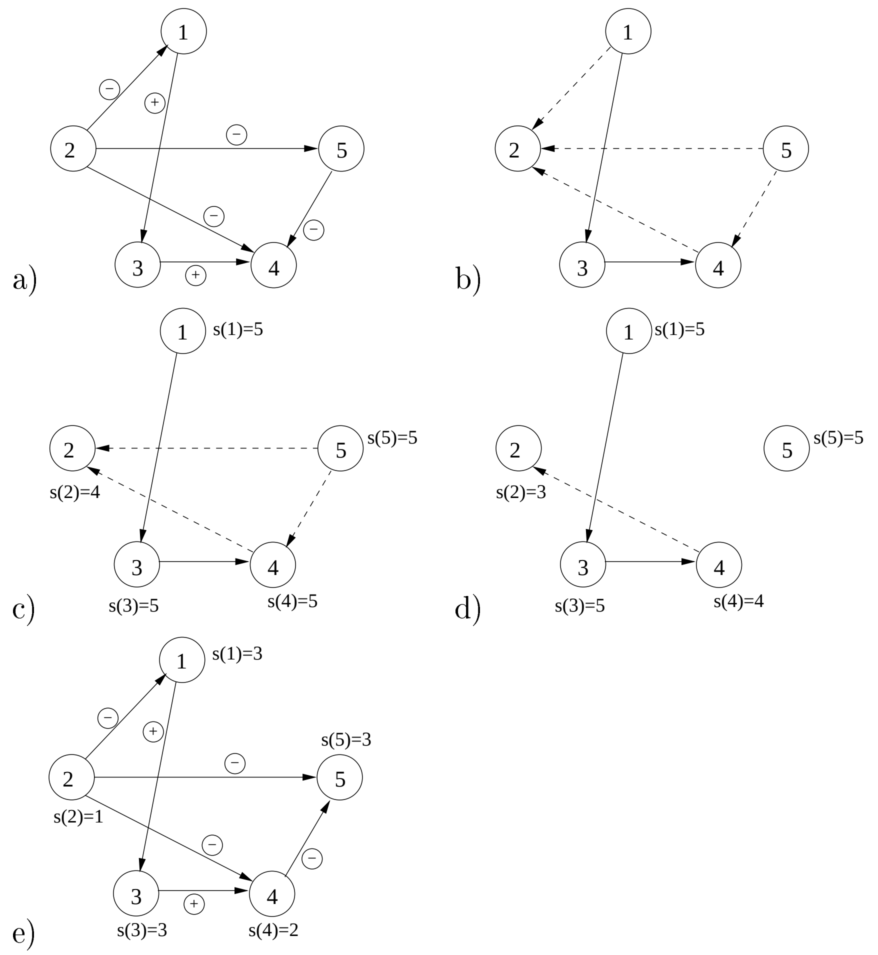


Fig. 3. *a*) A labeled digraph $G = (\{1, \ldots, 5\}, A)$. *b*) (G_R, lab_R). The arcs drawn in dotted lines are negative-arcs. The others are positive-arcs. *c*) and *d*) the update schedule computed by algorithm 1 after the *while loop*. *e*) The update schedule s such that $(G, lab) = (G, lab_s)$.

$\max\{s(v)|\ v \in V(G)\} = m$. Since $m < n$, there exists $i^* \in [\![L, n-1]\!]$ such that $|s^{-1}(i^*)| > 1$. Notice that $\forall (u,v) \in s^{-1}(i^*) \times s^{-1}(i^*) \cap A(G)$, $lab_s(u,v) = \oplus$. Besides, because there are not cycles in (G_R, lab_R), there exists $w \in s^{-1}(i^*)$ such that $\{v \in s^{-1}(i^*)|\ (w,v) \in A(G_R)\} = \emptyset$. Hence, let us define s' as follows:

$$s'(v) = \begin{cases} s(v) + 1 & \text{if } s(v) \geq s(w) \text{ and } v \neq w, \\ s(v) & \text{if } s(v) < s(w) \text{ or } v = w. \end{cases}$$

Hence obviously, $s'(V(G)) = [\![1, m+1]\!]$, i.e. s' in an update schedule of $V(G)$, and $(G, lab) = (G, lab_{s'})$. □

Corollary 13 *Let (G, lab) be a reduced update digraph and $L = L^-(G_R, lab_R)$. Then, $|[s]_G| \geq |V(G)| - L$, where s is an update schedule such that $(G, lab) = (G, lab_s)$.*

Corollary 14 *Let $(G = (V, A), lab_s)$ be an update digraph. $|[s]_G| > 1$ if and only if G is not strongly connected, neither such that (G_R, lab_R) is linear and negative.*

PROOF. If G is not strongly connected, neither such that (G_R, lab_R) is linear and negative, then $|[\![L, |V|]\!]| > 1$. Thus, by Theorem 12, $|[s]_G| > 1$, where $(G, lab) = (G, lab_s)$.

Conversely, there exists two particular cases in which $|[s]_G| = 1$:

(1) If G is linear digraph then $L = |V|$. Thus, in the case where $\forall a \in A$, $lab_G(a) = \ominus$, only one update schedule s satisfies $(G, lab) = (G, lab_s)$. On the contrary, if $\exists a \in A$, $lab_G(a) = \oplus$, there are 2^k such update schedules, where k is the number positive arcs of (G, lab).
(2) If G is strongly connected, then (G, lab) has only positive arcs. Thus, (G_C, lab_C) is reduced to one alone vertice. According to the previous Lemma, only one schedule satisfies $(G, lab) = (G, lab_s)$: the parallel update schedule, s_p.

□

As a consequence, because (G, lab_{s_p}) is not a negative linear digraph, $|[s_p]_G| > 1$ if and only if G is not strongly connected.

6 Number of update digraphs

In the previous section, given a labeled digraph (G, lab), we were interested by the existence of update schedules s such that $(G, lab) = (G, lab_s)$. And when there did exist such update schedules, we wanted to know how many there were.

In the present section, given a digraph G, we would like to determine how it can be labeled into an update digraph, that is, which are the labeling functions lab of G such that (G, lab) is indeed an update digraph. In particular, here, we focus on the number of equivalence classes $[\cdot]_G$ (rather than there sizes).

Definition 15 *We define the size of a labeled digraph (G, lab) by the number of its positive arcs.*

We define the following problem :

DIGRAPH UPDATE (DU) problem:

- **Input:** `A digraph` $G = (V, A)$ `and an integer` k;
- **Question:** `Does there exist a labeling function` $lab : A \rightarrow \{\oplus, \ominus\}$ `such that` (G, lab) `is an update digraph and its size is at most` k `?`

Theorem 16 *DIGRAPH UPDATE is NP-complete.*

PROOF. We are going to prove Theorem 16 by reduction to the FAS problem defined bellow and which is known to be NP-complete Garey and Johnson (1979):

FAS problem:

- **Input:** `A digraph` $G = (V, A)$ `and an integer` k;
- **Question:** `Does there exist a feedback arc set` F `of` G `such that` $|F| \leq k$ `?`

where a feedback arc set (FAS) F of G is a set of arcs such that the digraph $(V, A \backslash F)$ does not have any cycles. F is minimal if there does not exist $F' \subsetneq F$ FAS of G.

The reduction function we use to map an instance of FAS to an instance of DU is simply the identity. Next, for a given instance (G, k) we show that there exists a labeling function lab such that (G, lab) is update digraph of size at most k if and only if there exists a FAS F of G such that $|F| \leq k$.

($\Rightarrow$) Let lab be a labeling function such that (G, lab) is an update digraph of size at most k and let $F = \{a \in A(G) | \; lab(a) = \oplus\}$. Then, F is a FAS of size $|F| \leq k$. $G' = (V, A \setminus F)$ cannot contain any cycle since otherwise it would be negative cycle of (G, lab) which is not possible in an update digraph.

($\Leftarrow$) Let F be a minimal FAS of G such that $|F| \leq k$. Let $a \in F$. If every cycle of G containing a contains as well another arc of F then $F \setminus \{a\}$ is a FAS of G smaller than F. This contradicts the minimality of F. Thus, for every $a \in F$, there exists a cycle of G containing a and no other arc of F. Now, let us define the labeling function lab as follows:

$$\forall a \in F, \; lab(a) = \oplus \text{ and } \forall a \in A \setminus F, \; lab(a) = \ominus.$$

Note that because there are no cycles in $G' = (V, A \backslash F)$, there are not cycles in (G, lab). Suppose, however, that (G, lab) is not an update digraph. In (G, lab), there must thus be an alternating circuit (see Theorem 10 and the remarks made after) containing both positive and negative arcs. In other words, there is a forbidden cycle in (G_R, lab_R). The positive arcs in this cycle belong to F. Let $a \in A(G)$ be such a positive arc belonging to the forbidden cycle and to F. From the discussion above, we derive that there exists a cycle C_a of G that contains a and no other arc of F. All the arcs of C_a that are not a are thus negative in (G, lab). Concatenating the negative arcs of the alternating circuit and of all cycles C_a (a being an arc of F in the forbidden cycle) we obtain a cycle in $G' = (V, A(G) \setminus F)$ (see figure 3 below) which contradicts F being a FAS of G (as well as the fact that (G, lab) has no negative cycles). □

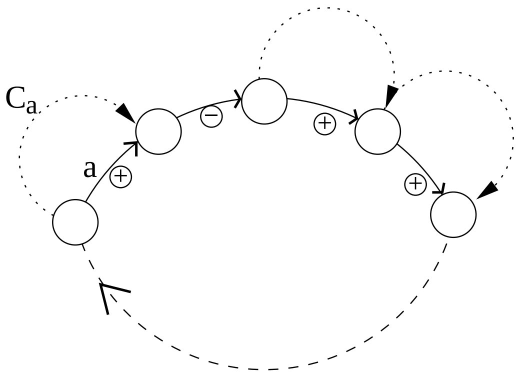


Fig. 4. A forbidden cycle in (G_R, lab_R) with, surrounding it, the negative cycles C_a mentioned in the proof of Theorem 16. Arrows in full line represent arcs, arrows in dashed lines reprensent paths.

Corollary 17 *Let (G, lab) be an update digraph. If N_{FAS} and N_{MFAS} are, respectively, the total number of FAS and minimal FAS of G. Then,*

$$N_{MFAS} \leq |\{[s]_G \mid s \text{ is un update schedule of } V(G)\}| \leq N_{FAS}$$

7 Projections

Theorem 18 *Let G be a digraph and G' a subdigraph of G. Then, (G', lab') is an update digraph if and only if there exists a labeling function lab of $A(G)$ such that $lab' = lab_{G'}$.*

PROOF. Obviously, if (G, lab) is an update digraph and $lab' = lab_{G'}$, then (G', lab') is also an update digraph by Theorem 10.

Conversely, if (G', lab') is an update digraph we will show that for all $a = (u, v) \in A(G) \setminus A(G')$, either $(G' + a, lab_a^+)$ or $(G' + a, lab_a^-)$ is an update digraph, where $V(G' + a) = V(G') \cup \{u, v\}$, $E(G' + a) = E(G') \cup \{a\}$ and lab_a^+

and lab_a^- are defined by $lab_a^+(e) = lab_a^-(e) = lab(e), \forall e \in A(G')$, $lab_a^+(a) = \oplus$ and $lab_a^-(a) = \ominus$.

Let us suppose that there exists $a = (u, v) \in A(G) \setminus A(G')$ such that neither $(G' + a, lab_a^+)$ nor $(G' + a, lab_a^-)$ are update digraphs. Then there exist a forbidden cycle $C_1 : x_1 = u, x_2 = v, x_3, \ldots, x_p = u$ with $lab_a^+(x_j, x_{j+1}) = \ominus$ in the reoriented labeled digraph $((G + a)_R, (lab_a^+)_R)$. In the same way, there exists a forbidden cycle $C_2 : y_1 = v, y_2 = u, y_3, \ldots, y_q = v$ in the reoriented labeled digraph $((G + a)_R, (lab_a^-)_R)$. Hence, la sequence of nodes $x_2 = v, \ldots, x_j, x_{j+1}, \ldots, x_p = u = y_2, \ldots, y_q = v$ in the reoriented labeled digraph (G_R, lab_R) contains a cycle including the arc (x_j, x_{j+1}) (see Fig...), that is a forbidden cycle. Thus (G, lab) is not un update digraph, which is a contradiction.

Therefore, if $A(G) \setminus A(G') = \{a_1, \ldots, a_r\}$, then by induction we can prove that for all k in $\{1, \ldots, r\}$ there exists a labeling function lab^k of the arcs of $G' + a_1 + \ldots + a_k$ such that $lab^k_{G'} = lab'$ and $(G' + a_1 + \ldots + a_k, lab_k)$ is an update digraph. In particular, there exists a labeling function lab in G such that (G, lab) is an update digraph and $lab' = lab_{G'}$. □

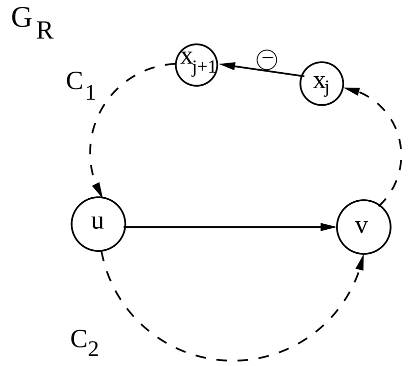


Fig. 5. Scheme of the forbidden cycle in (G_R, lab_R) mentioned in the proof of Theorem 18.

Corollary 19 *Let G be a connected digraph of $n > 1$ vertices. Then,*

$$2^{n-1} \;\leq\; |\{(G, lab) :\ (G, lab) \textit{ is update digraph }\}| \;\leq\; T_n$$

where $T_n = \sum_{k=0}^{n-1} \binom{n}{k} T_k$.

PROOF. From Theorem 18 for all digraph G and subdigraph G',

$$|\{(G', lab') :\ (G', lab') \text{ is UD}\}| \;\leq\; |\{(G, lab) :\ (G, lab) \text{ is UD}\}|.$$

On other hand, the connected digraph of n vertices with the least number of arcs, equal to $n - 1$, is an oriented tree. In this case, all labeling functions on the digraph yield an update digraph. Thus, there are 2^{n-1} update connected digraphs with the least number of arcs. In the same way, the connected digraph of n nodes with the greast number of arcs, equal to n^2 (including the arcs (u,u)), is the complete digraph. In this case, each labeling function on a complete digraph define a total preorder on the vertices. Besides, it is well known that the number of total preorders on a set of n elements is T_n defined as in the statement of this Theorem. Thus, T_n is the maximum number of update connected digraphs with n vertices. □

8 Acknowledgments

This work was partially supported by FONDECYT project 1061008 (J.A.), by FONDAP and BASAL projects CMM, Universidad de Chile, by Centro de Investigación en Ingeniería Matemática (CI^2MA), Universidad de Concepción, and by CONICYT fellowship (M.M.).

References

J. Aracena, E. Goles, A. Moreira, L. Salinas, On the robustness of update schedules in Boolean networks, Biosystems 97 (2009) 1–8.

S. Kauffman, Metabolic stability and epigenesis in randomly connected nets, Journal of Theoretical Biology 22 (1969) 437–67.

R. Thomas, Boolean formalization of genetic control circuits, Journal of Theoretical Biology 42 (1973) 563–585.

I. Shmulevich, H. Lähdesmäki, E. Dougherty, J. Astola, W. Zhang, The role of certain Post classes in Boolean network models of genetic networks, Proceedings of the National Academy od Sciences USA 100 (2003) 10734–10739.

R. Thomas, Regulatory networks seen as asynchronous automata: A logical description, Journal of Theoretical Biology 153 (1991) 1–23.

M. Chaves, R. Albert, E. Sontag, Robustness and fragility of Boolean models for genetic regulatory networks, Journal of Theoretical Biology 235 (2005) 431–449.

H. Mortveit, C. Reidys, Discrete, sequential dynamical systems, Discrete Mathematics 226 (2001) 281–295.

A. Hansson, H. Mortveit, C. Reidys, On asynchronous cellular automata, Advances in Complex Systems 8 (4) (2005) 521–538.

J. Demongeot, A. Elena, S. Sen, Robustness in Regulatory Networks: a Multi-Disciplinary Approach, Acta Biotheoretica 56 (1-2) (2008) 27–49.

J. van Leeuwen, Handbook of theoretical computer science, Elsevier, 1990.

M. Garey, D. Johnson, Computers and intractability: a guide to the theory of NP-completeness, W.H. Freeman, 1979.

Centro de Investigación en Ingeniería Matemática (CI²MA)

PRE-PUBLICACIONES 2009

2009-12 Alfredo Bermúdez, Luis Hervella-Nieto, Andres Prieto, Rodolfo Rodríguez: *Perfectly matched layers for time-harmonic second order elliptic problems*

2009-13 Ricardo Durán, Rodolfo Rodríguez, Frank Sanhueza: *Computation of the vibration modes of a Reissner-Mindlin laminated plate*

2009-14 Gabriel N. Gatica, Antonio Marquez, Manuel A. Sanchez: *Analysis of a velocity-pressure-pseudostress formulation for the stationary Stokes equations*

2009-15 Raimund Bürger, Kenneth H. KARLSEN, John D. Towers: *On some difference schemes and entropy conditions for a class of multi-species kinematic flow models with discontinuous flux*

2009-16 Gabriel N. Gatica, George C. Hsiao, Francisco J. Sayas: *Relaxing the hypotheses of the Bielak-MacCamy BEM-FEM coupling*

2009-17 Iuliu S. Pop, Florin A. Radu, Mauricio Sepúlveda, Octavio Vera: *Error estimates for the finite volume discretization for the porous medium equation*

2009-18 Raimund Bürger, Kenneth H. KARLSEN, Hector Torres, John D. Towers: *Second-order schemes for conservation laws with discontinuous flux modelling clarifier-thickener units*

2009-19 Maria G. Armentano, Claudio Padra, Rodolfo Rodríguez, Mario Scheble: *An hp finite element adaptive scheme to solve the Laplace model for fluid-solid vibrations*

2009-21 Carlos D. Acosta, Raimund Bürger, Carlos E. Mejia: *Monotone difference schemes stabilized by discrete mollification for strongly degenerate parabolic equations*

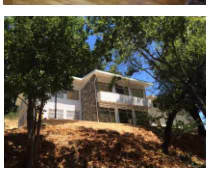
2009-22 Gabriel N. Gatica, Salim Meddahi: *Finite element analysis of a time harmonic Maxwell problem with an impedance boundary condition*

2009-23 Veronica Anaya, Mostafa Bendahmane, Mauricio Sepúlveda: *Mathematical and numerical analysis for predator-prey system in a polluted environment*

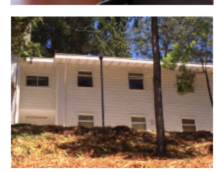
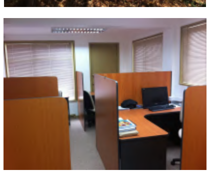
2009-24 Julio Aracena, Eric Fanchon, Marco Montalva, Mathilde Noual: *Combinatorics on update digraphs of discrete networks*

Para obtener copias de las Pre-Publicaciones, escribir o llamar a: Director, Centro de Investigación en Ingeniería Matemática, Universidad de Concepción, Casilla 160-C, Concepción, Chile, Tel.: 41-2661324, o bien, visitar la página web del centro: `http://www.ci2ma.udec.cl`



Centro de Investigación en
Ingeniería Matemática (CI^{2}MA)
Universidad de Concepción

Casilla 160-C, Concepción, Chile
Tel.: 56-41-2661324/2661554/2661316
http://www.ci2ma.udec.cl