Problèmes type "Feedback Set" et comportement dynamique des réseaux de régulation

Marco Montalva Medel

UNIVERSITÉ DE GRENOBLE

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ DE GRENOBLE

Spécialité : **MODELES, METHODES ET ALGORITHMES EN BIOLOGIE, SANTÉ ET ENVIRONNEMENT**

Arrêté ministériel : 6 Janvier 2005 (relatif à la cotutelle internationale de thèse) et 7 Août 2006 (relatif à la formation doctorale)

Présentée par

Marco Antonio Sebastián MONTALVA MEDEL

Thèse dirigée par **Dr. Julio ARACENA**
et codirigée par **Dr. Jacques DEMONGEOT**

préparée au sein du **Laboratoire TIMC-IMAG CNRS UMR 5525**
et de **EDISCE - École Doctorale Ingénierie pour la Santé, la Cognition et l'Environnement**

Problèmes type feedback et comportement dynamique des réseaux régulatoires

Thèse soutenue publiquement le **18 Août 2011**,
devant le jury composé de :

Martín MATAMALA
Dr., Universidad de Chile, Président
Eric GOLES
Dr., Universidad Adolfo Ibáñez, Rapporteur
Jean-Paul COMET
Dr., Université de Nice Sophia-Antipolis, Rapporteur
Eric FANCHON
Dr., Laboratoire TIMC-IMAG CNRS UMR 5525, Examinateur
Julio ARACENA
Dr., Universidad de Concepción, Directeur de thèse
Jacques DEMONGEOT
Dr., Université Joseph Fourier, Co-Directeur de thèse

Agradecimientos

En primer lugar, agradezco el apoyo que siempre me ha dado mi familia a lo largo de mis estudios; a mi padre por estar disponible ante cualquier necesidad, a mi madre por recordarme las cosas importantes, a mis hermanos por la ayuda informática que necesité, a mi Tío Hugo y a mi Tata por las entretenidas conversaciones de los fines de semana. También, a Maricell por el optimismo brindado.

Por otro lado, agradezco el apoyo, el entusiasmo y la confianza dada por mi profesor guía, Dr. Julio Aracena, quien además de ser un investigador notable y muchas veces ayudarme a entender resultados no triviales, ha sabido distribuir de manera equitativa el tiempo con sus alumnos, aprovechándolo de manera eficiente, así como también, se ha preocupado de generar nuevas oportunidades de trabajo tanto en Chile como en el extranjero, mostrándome siempre un abanico de posibilidades para mi futuro.

A mi director en Francia, el Dr. Jacques Demongeot, por darme nuevos enfoques e ideas en mi tabajo de investigación y por facilitarme las cosas en el extranjero ya sea personalmente o a través de su secretaria, la srta. Céline Fontant. A los profesores que aceptaron ser parte de la comisión, los Doctores: Eric Goles, Martín Matamala, Jean-Paul Comet y Eric Fanchon. A los colegas y amigos del Laboratorio TIMC-IMAG con quienes compartí gratos momentos durante mi estadía: Eric Fanchon, Hedi Ben-Amor, Nicolas Glade y Fabien Corblin. A los colegas y amigos de Lyon: Mathilde Noual y Sylvain Sené.

Al CI^2MA por darme el espacio físico de trabajo y a las personas que normalmente en él se encuentran: Abner, Lorena, Carolina, Edgardo, Eduardo, Angelina, Bibiana, Jessika, Luis, etc., con las que he compartido este último tiempo.

Finalmente, agradezco a CONICYT por la becas de Doctorado, Colegio Franco-Chileno y Término de Tesis Doctoral, así como también por el proyecto FONDECYT 1090549, a MECESUP por el proyecto UCO0713 y al Laboratorio TIMC-IMAG por las misiones a Lyon y a Túnez.

A mi hija Antonella.

Contents

CHAPTER 1

Résumé

Dans la nature existent de nombreux exemples des systèmes dynamiques complexes: systèmes neuronaux, communautés, écosystèmes, réseaux de régulation génétiques, etc. Notre intérêt porte sur ces derniers qui sont souvent modélisés par des réseaux Booléens.

Un réseau Booléen peut être considéré comme un digraphe, où les sommets correspondent à des gènes ou des produits de gènes, tandis que les arcs indiquent les interactions entre eux. Un niveau d'expression des gènes est modélisé par des valeurs binaires, 0 ou 1, indiquant deux états de transcription, soit activif, soit inactif, et ce niveau change dans le temps selon certaines fonctions locales d'activation qui dépendent des états d'un ensemble de noeuds (les gènes). L'effet conjoint des fonctions d'activation locale définit une fonction de transition globale: ainsi, l'autre élément nécessaire dans la description du modèle est une fonction de mise à jour, qui détermine quand chaque noeud doit être mis à jour, et donc, comment les fonctions locales se combinent dans une fonction globale. Comme un réseau Booléen avec n sommets possède 2^n états globaux, à partir d'un état de départ, et dans un nombre fini de mises à jour, le réseau atteindra un point fixe ou un cycle limite appelés attracteurs et sont souvent associés à des phénotypes distincts (états cellulaires) définis par les patrons d'activité des gènes.

Un réseau de régulation Booléen (REBN) est un réseau Booléen où chaque interaction entre les éléments du réseau correspond soit à une interaction positive soit à une interaction négative. Ainsi, le digraphe interaction associé à un REBN est un digraphe signé où un circuit est appelé positif (négatif) si le nombre de ses arcs négatifs est pair (impair). Dans ce contexte, diverses études existent sur l'importance des circuits positif et négatifs dans le comportement dynamique de différents systèmes en Biologie. En effet, le point de départ de cette thèse est basé sur un résultat disant que le nombre

maximal de points fixes d'un REBN dépend d'un ensemble de sommets de cardinalité minimale qui intersecte tous les cycles positifs (positive feedback vertex set) du digraphe signé associé.

D'autre part, un autre aspect important des circuits est leur rôle dans la robustesse des réseaux Booléens par rapport à différents types de mise à jour déterministe. Dans ce contexte, un élément clé mathématique est le digraphe update qui est un digraphe étiqueté associé au réseau dont les étiquettes sur les arcs sont définies comme suit: un arc (u, v) est dit être positif si l'état de sommet u est mis à jour en même temps ou après que celle de v, et négative sinon. Ainsi, un cycle dans le digraphe étiqueté est dite positive (négative) si tous ses arcs sont positifs (négatifs). Cela met en évidence que parler de "positif" et "négatif" a des significations différentes selon le contexte: digraphes signés ou digraphes étiquetés.

Ainsi, nous allons voir dans cette thèse, les relations entre les feedback sets et la dynamique des réseaux Booléens à travers l'étude analytique de ces deux objets mathématiques fondamentaux: le digraphe signé et le digraphe update.

Mots clés: Digraphe update, mise à jour, feedback set, digraphe signé.

CHAPTER 2

Abstract

In the nature there exist numerous examples of complex dynamical systems: neural systems, communities, ecosystems, genetic regulatory networks, etc. These latest, in particular are of our interest and are often modeled by Boolean networks.

A Boolean network can be viewed as a digraph, where the vertices correspond to genes or gene products, while the arcs denote interactions among them. A gene expression level is modeled by binary values, 0 or 1, indicating two transcriptional states, either active or inactive, respectively, and this level changes in time according to some local activation function which depends on the states of a set of nodes (genes). The joint effect of the local activation functions defines a global transition function; thus, the other element required in the description of the model is an update schedule which determines when each node has to be updated, and hence, how the local functions combine into the global one. Since a Boolean network with n vertices has 2^n global states, from a starting state, within a finite number of udpates, the network will reach a fixed point or a limit cycle called attractors and are often associated to distinct phenotypes (cellular states) defined by patterns of gene activity.

A regulatory Boolean network (REBN) is a Boolean network where each interaction between the elements of the network corresponds either to a positive or to a negative interaction. Thus, the interaction digraph associated to a REBN is a signed digraph where a circuit is called positive (negative) if the number of its negative arcs is even (odd). In this context, there are diverse studies about the importance of the positive and negative circuits in the dynamical behavior of different systems in Biology. Indeed the starting point of this Thesis is based on a result saying that the maximum number of fixed points of a REBN depends on a minimum cardinality vertex set whose elements

intersects to all the positive cycles (positive feedback vertex set) of the associated signed digraph.

On the other hand, another important aspect of circuits is their role in the robustness of Boolean networks with respect to different deterministic update schedules. In this context a key mathematical element is the update digraph which is a labeled digraph associated to the network and whose labels on the arcs are defined as follows: an arc (u, v) is said to be positive if the state of vertex u is updated at the same time or after than that of v, and negative otherwise. Hence, a cycle in the labeled digraph is called positive (negative) if all its arcs are positive (negative). This leaves in evidence that the terms "positive" and "negative" have different meanings depending on the context: signed digraphs or labeled digraphs.

Thus, we will see in this thesis relationships between feedback sets and the dynamics of Boolean networks through the analytical study of these two fundamental mathematical objects: the signed (connection) digraph and the update digraph.

Keywords: Update digraph, update schedule, feedback set, signed digraph.

CHAPTER 3

Introduction

In the nature there exist numerous examples of complex dynamical systems: neural systems, communities, ecosystems, genetic regulatory networks, etc. These latter are particulary of interest to us.

Basically, a genetic regulatory network corresponds to the interaction of a group of genes and gene products of a cell or a group of cells, that origin diverse cellular functions such as morphogenesis, metabolism, etc.

The discrete modeling of genetic regulatory networks was introduced by Kauffman more than thirty years ago (Kauffman, 1969, 1973, 1993). The central hypothesis is that the acquision of a specific cellular state (mobility, differentiation, proliferation, change of shape, metabolic adaptation, etc.) is determined by the profile of activation of a group of components that conforms a genetic regulatory network in the cell. This interaction can be mathematically modeled by a Boolean network.

A Boolean network can be viewed as a digraph, where the vertices correspond to genes or gene products, while the arcs denote interactions among them. A gene expression level is modeled by binary values, 0 or 1, indicating two transcriptional states, either active or inactive, respectively, and this level changes in time according to some local activation function which depends on the states of a set of nodes (genes). The joint effect of the local activation functions defines a global transition function. Thus, the other element required in the description of the model is an update schedule, which determines when each node has to be updated, and hence, how the local functions combine into the global one (in other words, it must describe the relative timings of the regulatory activities). Since a Boolean network with n vertices has 2^n global states, from a starting state, within a finite number of udpates, the network will reach a fixed point or a limit cycle, called attractor.

The attractors of a Boolean network are often associated to distinct phenotypes (cellular states) defined by patterns of gene activity. A regulatory Boolean network (REBN) is a Boolean network where each interaction between the elements of the network corresponds either to a positive or to a negative interaction. Thus, the interaction digraph associated to a REBN is a signed digraph where a circuit is called positive (negative) if the number of its negative arcs is even (odd).

In this context, there are diverse studies about the importance of the positive and negative circuits in the dynamical behavior of non-linear systems in Biology (Demongeot, 1998; Demongeot et al., 2000; Kauffman, 1973; Thomas and D'Ari, 1990). In fact, one has demonstrated that the positive circuits are necessary for the multi-stationarity (Plahte et al., 1995; Snoussi, 1998; Gouzé, 1998; Cinquin and Demongeot, 2002; Soulé, 2003; Richard and Comet, 2007; Richard, 2009), whose biological meaning can be differentiation and memory, and the negative circuits are a necessary condition for the existence of stable regularities what in biology represents the homeostasis (Snoussi and Thomas, 1993; Thomas et al., 1995; Demongeot et al., 2000; Aracena et al., 2003). In addition, a simple result between the disjoint positive circuits and the number of stable configurations has been established (Thomas and Richelle, 1988; Thomas and Kaufman, 2001).

Indeed the starting point of this thesis is based on a result of Aracena (2001, 2008), saying that the maximum number of fixed points of a REBN depends on a minimum cardinality vertex set whose elements intersects to all the positive cycles (also named a *positive feedback vertex set*) of the associated signed digraph.

On the other hand, in (Sontag et al., 2008) was shown that, as the number of independent negative feedback loops increases, the number of limit cycles of the REBN tends to decrease and its length tends to increase. In other words, the limit cycles in a REBN are related with the minimum cardinality of a *negative feedback vertex set*. Both decision problems of finding; a positive feedback vertex set and a negative feedback vertex set, of minimum cardinality, where introduced in (Montalva, 2006) as PFVS and NFVS respectively, where begins the study of the complexity of these problems.

Besides, PFVS and NFVS can be viewed as variants of the important classical decision problem: Feedback Vertex Set (FVS) for digraphs, which is well-known to be NP-complete (Karp, 1972) and for which there are many variants (some of them consider weights on the vertices or on the arcs), almost all of them have been proved to be NP-complete as well. Furthermore, feedback problems are fundamental in combinatorial optimization, having many applications: circuit design, certain scheduling problems and cryptography are some examples. For this reason, they have been extensively studied (see (Festa et al., 1999) for a good survey).

In consequence, as the study of complexity of PFVS and NFVS is a key feature in the understanding of REBNs as well as an interesting theoretical problem, this thesis starts deepening in these and other related problems.

On the other hand, another important aspect of circuits is their role in the robustness of Boolean networks with respect to different deterministic update schedules. In this context, some of the pioneering works were made by Robert (1986) and Goles (1986). The choice of deterministic update schedules is given by the fact that information processing performed in the living cell has to be extremely robust and therefore requires a quasi deterministic dynamics. Another reason for determinism is the need to model some periodical behaviors; when randomness is introduced, attractors become regions of the phase space, but are no longer exact dynamical cycles. Both stochastic and deterministic models are common in the biological literature, and a frequent strategy is to consider a deterministic dynamics and look at its robustness under small random perturbations.

The impact of perturbations of the update schedule on a Boolean network dynamics have been greatly studied (Chaves et al., 2005; Elena et al., 2008; Ben-Amor et al., 2008; Demongeot et al., 2008; Elena, 2009), mainly from a statistical point of view and more recently, also from an analytical point of view (Salinas, 2008; Gómez, 2009).

Some analytical works on 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 is an undirected graph and the update schedule is sequential. For this class of networks, the team of Hansson, 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 (Salinas, 2008) were defined equivalence classes of deterministic update schedules in Boolean networks according to the labeled digraph associated to the network (update digraph) and whose labels on the arcs are defined as follows: an arc (u, v) is said to be positive if the state of vertex u is updated at the same time or after than v, and negative otherwise. Hence, a cycle in the labeled digraph is called positive (negative) if all its arcs are positive (negative). This leaves in evidence that talk of "positive" and "negative" has different meanings depending on the contex: signed digraphs or labeled digraphs.

Besides, in (Salinas, 2008; Aracena et al., 2009) was proven that two update schedules in the same class yield exactly the same dynamical behavior. Motivated by this result, we study, from a mathematical point of view, the update digraphs and the number and size of these equivalence classes associated to it. All this, in order to get an idea of the possible different dynamics of networks according to the update schedule used. Such a study represents the core of this thesis. In general terms, we found out that these concepts are closely related to the feedback arc sets of the connection digraph associated to the network. These relations were reflected in combinatorial and structural aspects relating the schedule classes of update digraphs with the feedback arc sets of their connection digraphs.

In summary, we will see in this thesis relationships between feedback sets, as above mentioned, and the dynamics of Boolean networks through the analytical study of two fundamental mathematical objects: the signed (connection) digraph and the update digraph.

3.1 Thesis contents

The work of this thesis consists in the study of feedback set problems in signed digraphs as well as the study of the schedule equivalence classes associated to a network.

In chapter 2, we present the necessary terminology and notations to develop the following chapters. We mainly grouped these concepts in three parts: graphs and digraphs, feedback sets and Boolean networks.

In chapter 3, we continue with the study begun in (Montalva, 2006), where there was proven that PFVS is NP-complete, leaving the complexity of NFVS as an open problem. Thus, in this chapter is concluded the analysis of NFVS and we add the study of the analogous versions for the arcs: the PFAS and NFAS problem, all them for the general case. Next, we will study PFVS and NFVS for different families of signed digraphs with additional constraints in the structure or in the same distribution of the signs, for example, in special cases of applied interest such as Kauffman or monotone networks. Although FVS, PFVS and NFVS are NP-complete for the general case, the intuition is that there exists differences of complexity between them for certain cases. We are interested in finding particular families of digraphs where the complexity of these problems is different, and thus to understand better, how the structural properties and sign distribution can determine the complexity of these problems.

The main results of the chapter can be summarized as follows:

- PFAS and NFAS are NP-complete for the general case.
- PFVS and NFVS are NP-complete for the following families of signed digraphs: with the maximum in-degree of each vertex bounded by $k \in \mathbb{N}$ (in the trivial case $k = 1$, both are P), with the in-degree of each vertex exactly equal to $k \in \mathbb{N}$ (Kauffman's networks) and with the incoming (or outgoing) arcs of each vertex having the same sign (monotone locally REBN).

- PFVS is P and NFVS is NP-complete for the family of signed cliques.
- PFVS is NP-complete and NFVS is P for the family of signed digraphs with at most $k \in \mathbb{N}$ negative arcs.
- FVS is P (Lloyd et al., 1985) whereas PFVS and NFVS are NP-complete for the family of cyclically reducible digraphs.

Some of this results were published in:

(1) J. Aracena, A. Gajardo, M. Montalva, *On the complexity of feedback set problems in signed digraphs*, Electronic Notes in Discrete Mathematics 30 (2008) 249-254.

In chapter 4, we study combinatorial aspects of the equivalence classes of deterministic update schedules that yield the same update digraph and thus the same dynamical behavior of the associated Boolean network (see (Aracena et al., 2009) for more details).

Motivated by the above mentioned, we show a polynomial characterization of these update digraphs, which enables us to determine the corresponding schedule equivalence classes as well as a particular schedule in a class. We study the complexity of the problem of finding a label function on the arcs of a digraph with at most $k \in \mathbb{N}$ positive arcs such that the resultant digraph is update (DU problem), which enables us to find bounds relating different types of feedback arc sets with the schedule equivalence class of an update digraph.

On the other hand, we prove that the update digraphs are exactly the projections, on the respective subdigraphs, of a complete update digraph with the same number of vertices. The exact number of complete update digraphs is determined, which provides upper and lower bounds on the number of schedule equivalence classes. We see how while the number of equivalence classes increases, the number of feedback arc sets decreases as well as a necessary and sufficient condition for the equality.

Finally, we show relations between the feeedback arc sets of a given digraph and its schedule equivalence classes for particular families of digraphs such as complete digraphs, acyclic digraphs and tournaments. Thus, we can find the number and size of these sets in this families.

Let G be a digraph whose set of vertices is $V(G)$ and set of arcs is $A(G)$ and let $lab : A(G) \to \{\ominus, \oplus\}$ be a *label function* of G. The main results can be summarized as follows:

- A *labeled digraph* (G, lab) is update if and only if it has no circuit being a cycle in (G_R, lab_R) with at least a negative arc (*forbidden circuit*), where (G_R, lab_R) is the labeled digraph constructed from (G, lab) by changing the orientation of the negative arcs and keeping the original labels.

- The problems of determining whether a labeled digraph is update and finding a corresponding update schedule s are polynomial.
- If (G, lab) is an update digraph, then there exists a sequential update schedule s_q associated with (G, lab) if and only if (G, lab) has no positive cycle. In this case, the schedule class associated to (G, lab) has size strictly greater than one if and only if (G_R, lab_R) is not a negative linear digraph.
- DU problem is NP-complete. Furthermore, from the proof we deduce that the size of the set $U(G)$ of schedule classes associated to a digraph G is upper bounded by the size of the set $FAS(G)$ of feedback arc sets of G and lower bounded by the size of the set $MFAS(G)$ of minimal feedback arc sets of G. Specifically, $|MFAS(G)| < |U(G)| \leq |FAS(G)|$.
- A subdigraph of an update digraph also is an update digraph. Besides, if G is a connected digraph, then $2^{n-1} \leq |U(G)| \leq T_n$, where $T_n = \sum_{k=0}^{n-1} \binom{n}{k} T_k$ with $T_0 \equiv 1$, is the exact number of schedule classes associated to a complete digraph D of n vertices (eventually with loops), i.e., $A(D) = V \times V$.
- If G is an undirected graph and G_1 and G_2 two orientations of G such that every cycle of G_1 is also a cycle of G_2, then $|U(G_1)| \leq |U(G_2)|$ and $|FAS(G_2)| \leq |FAS(G_1)|$. Moreover, $|U(D)| = |FAS(D)|$ if and only if all circuits of a digraph D are cycles.
- If G is a complete digraph, then (G, lab) is a non-update digraph if and only if there exists a forbidden cycle of length either two or three in (G_R, lab_R).
- If G is an acyclic digraph with $|V(G)| = n$, then $|U(G)| \leq n!$ and the equality holds if and only if G is a tournament, i.e., an orientation of a complete undirected graph. In this case all its schedule classes are of size 2^k, for some $k \in \mathbb{N} \cup \{0\}$.
- If G is 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 $lab_F(u, v) = \oplus \Leftrightarrow (u, v) \in F$.

Some of these results were published in:

(2) J. Aracena, E. Fanchon, M. Montalva, M. Noual, *Combinatorics on update digraphs in Boolean networks*, Discrete Applied Mathematics 159 (6) (2011) 401-409.

(3) J. Aracena, J. Demongeot, E. Fanchon, M. Montalva, *On the number of update digraphs and its relation with the feedback arc sets and tournaments*, Discrete Applied Mathematics, Ref. No. DA1309, submitted.

In chapter 5, we are motivated by an algorithm presented in (Schwikowski and Speckenmeyer, 2002) that exploits a simple relation between minimal feedback arc sets that allows generating all minimal feedback arc sets of a digraph $G = (V, A)$ by local modifications. They further show that the underlying technique can be tailored to generate all minimal solutions for the undirected case and the directed feedback arc set problem, both with a polynomial delay of $O[|V||A|(|V| + |A|)]$, proving finally that computing the number of minimal feedback arc sets is $\sharp$P-hard.

Thus, we explore a similar idea in the context of the update digraphs, generating local transformations over an update digraph (G, lab) and studying structural properties of the multidigraph H_G associated to these transformations where the set of its nodes is the set of update digraphs associated to G and the arcs represent some local transformations between them (see definition 7.2, p. 68, for more details).

Next, we give algorithms for enumerating all the update digraphs associated with a given digraph as well as for exactly determining all the update schedules associated to it.

Finally, as an application example of the theoretical results obtained in this thesis, we analyze the possible dynamics of the real genetic regulation network of the floral morphogenesis of the plant *Arabidopsis thaliana*. For this, we consider the *reduced Mendoza and Alvarez-Buylla network* which has two non-trivial strongly connected symmetric components and whose asymptotic dynamics has the same attractors as the original network (Demongeot, Goles, Morvan, Noual, and Sené, 2010). We compare our numerical results with those obtained in the previous article.

The main results can be summarized as follows:

- The local transformations on update digraphs give us update digraphs and their composition over a feedback vertex set of G without its orientations, is an update digraph.
- If $H = (G, lab)$ is an update digraph, then $(H, H) \in A(H_G)$ if and only if $(H', H) \in A(H_G)$ where H' is another update digraph different to H. Besides, $(H, H) \notin A(H_G)$ if and only if $|\{i \in V(G) : s(i) \geq s(j),\ \forall j \in V(G)\}| > 1$, where $s : \{1, ..., |V(G)|\} \to \{1, ..., |V(G)|\}$ is an update schedule of G such that $s(\{1, ..., |V(G)|\}) = \{1, ..., m\}$, $m \leq |V(G)|$, m being maximum.
- If H is a digraph, then H_G is connected. Moreover, H_G restricted to the

reduced update digraphs is strongly connected, which means that it is possible to enumerate with polynomial delay all update digraphs in this family.

From this chapter there is an article in preparation:

(4) J. Aracena, J. Demongeot, M. Montalva, *Local transformations and enumeration of update digraphs.*

Finally, we discuss the results obtained in this thesis and propose open problems and future lines of investigation in the chapter Conclusions.

CHAPTER 4

Basic terminology and notations

In the sequel, for any integers a and b with $a < b$, we will denote:

$$[\![a, b]\!] = \{i \in \mathbb{Z} : \ a \leq i \leq b\},$$
$$[\![a, b[\![= [\![a, b-1]\!].$$

4.1 Digraphs and graphs

A **directed graph** (or just **digraph**) G consists of a non-empty finite set $V(G)$ of elements called **vertices** (or **nodes**) and a finite set $A(G)$ of ordered pairs of distinct vertices called **arcs**. We call $V(G)$ the **vertex (or node) set** and $A(G)$ the **arc set** of G. We will often write $G = (V, A)$ which means that V and A are the vertex set and arc set of G, respectively.

For an arc (u, v) the first vertex u is its **tail** and the second vertex v is its **head**. We draw G on paper by placing each vertex of V at a point and representing each arc (u, v) by an arrow from u to v. An ordered pair $(u, u) \notin A$ is called a **loop** of G. The above definition of a digraph implies that we do not allow it either loops or **multiple** arcs, that is, pairs of arcs with the same tail and the same head. When multiple arcs and loops are admissible we speak of **multidigraphs**. In the case of need of loops in a digraph, it will be said explicitly. For instance, unless otherwise specified, $G = (V, A)$ is a digraph.

A **subdigraph** of G is a digraph $G' = (V', A')$ where $V' \subseteq V$ and $A' \subseteq (V' \times V') \cap A$. We write $G' \subseteq G$. If $A' = (V' \times V') \cap A$, we say that G' is **induced** by V' (we write $G' = G[V']$) and call G' an **induced subdigraph** of G. If $V' \subsetneq V \vee A' \subsetneq A$, then we write $G' \subsetneq G$. We will often write $G - U$ whose meaning depends of the nature of U. Thus, if $U \subseteq V$, then $G - U$ is the subdigraph of G induced by $V - U$, i.e. $G - U = G[V - U]$. If $U \subseteq A$, then $G - U = (V, A - U)$.

Let G' be a subdigraph of G and B a non-empty finite set. If $f : A(G) \to B$ is a function on the arcs of G, then the function f **restricted** to G', is a function $g : A(G') \to B$ such that for each $a \in A(G')$, $g(a) = f(a)$. We write $g = f|_{G'}$.

G is a **complete** digraph if $A = \{(u, v) : \ u, v \in V \wedge u \neq v\}$. A **clique** K_r of a digraph G is a complete subdigraph of G where $r = |V(K_r)|$.

For a vertex $v \in V$, we use the following notation:

$$N_G^+(v) = \{u \in V : \ (v, u) \in A\}, \quad N_G^-(v) = \{u \in V : \ (u, v) \in A\}$$

The sets $N_G^+(v)$, $N_G^-(v)$ and $N_G(v) \equiv N_G^+(v) \cup N_G^-(v)$ are called the **out-neighbourhood**, **in-neighbourhood** and **neighbourhood** of v, respectively. We call the vertices in $N_G^+(v)$, $N_G^-(v)$ and $N_G(v)$ the **out-neighbours**, **in-neighbours** and **neighbours** of v, respectively.

A **walk** from a vertex v_1 to a vertex v_m in G is a sequence of vertices $[v_1, v_2, \ldots, v_m]$ of V such that $\forall k \in [\![1, m-1]\!]$, $(v_k, v_{k+1}) \in A(G)$ 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 **path** $[v_1, v_2, \ldots, v_m]$ of **length** $m - 1$ is a walk such that $(v_k, v_{k+1}) \in A$ for all $k \in [\![1, m-1]\!]$. A **cycle** is a closed elementary path. In particular, a loop is a cycle of length one. If G has no cycle, G is called an **acyclic** digraph. Note that a cycle is always a circuit but the converse is not always true.

G is said to be **connected** if there is a walk between every pair of vertices, and **strongly connected** if there is a path between every pair of vertices. G' is a strongly connected **component** of G if G' is a strongly connected subdigraph of G and is maximal for this property, i.e. there is no other strongly connected component G'' of G such that $G' \subsetneq G''$.

REMARK 4.1 *All the previous definitions are analougous for multidigraphs.*

An **undirected graph** (or a **graph**) $G = (V, E)$ consists of a non-empty finite set $V = V(G)$ of elements called **vertices** (or **nodes**) and a finite set $E = E(G)$ of unordered pairs of distinct vertices called **edges**. We call $V(G)$ the **vertex (or node)**

set and $E(G)$ the **edge set** of G. In other words, an edge $\{x, y\}$ is a two-element subset of V.

An **orientation** of G is a digraph $G' = (V', A)$ where $V' = V$, $|E| = |A|$ and $\forall\{x, y\} \in E$, either $(x, y) \in A$ or $(y, x) \in A$. G is a **complete** graph if $\forall x, y \in V$, $x \neq y$, then $\{x, y\} \in E$. A **tournament** is a digraph consisting of an orientation of a complete graph.

More terminology about digraphs and graphs can be found in (Bang-Jensen and Gutin, 1979; West, 1996).

4.2 Feedback sets

A vertex (An arc) set $U \subseteq V$ ($U \subseteq A$) is a **feedback vertex (arc) set** of G if $G - U$ has no cycle, i.e. $G-U$ is an acyclic digraph. U is said to be a **minimal** feedback vertex (arc) set of G if U is a feedback vertex (arc) set of G and there is no other feedback vertex (arc) set $W \subseteq V$ ($W \subseteq A$) of G such that $W \subsetneq U$. U is said to be a **minimum** feedback vertex (arc) set of G if U is a feedback vertex (arc) set of G and there is no other feedback vertex (arc) set $W \subseteq V$ ($W \subseteq A$) of G such that $|W| < |U|$. Major details can be found in (Bang-Jensen and Gutin, 1979). More generally, the problem of finding the minimum feedback vertex or arc set has many application in graph theory as well as in pure mathematics, for example in (Karatkevich, 2001; Hawick and James, 2008; Vik, 2010; Cinkir, 2011).

Very close to these concepts, there are the classic decision problems FVS and FAS that are NP-complete (Karp, 1972; Garey and Johnson, 1979) and whose definition is as follows:

> **FVS (FAS)**. Given a digraph G and given $t \in \mathbb{N}$. Does a feedback vertex (arc) set U exist such that $|U| \leq t$?

4.3 Boolean networks, update schedules and dynamical behavior

A Boolean network $N = (G, F, s)$ is defined by:

- A digraph $G = (V, A)$, with $|V(G)| = n$, called **connection** (or **interaction**) **digraph**, where each node $i \in V$ has an associated state $x \in \{0, 1\}$.

- A **global activation function** $F : \{0,1\}^n \to \{0,1\}^n$, defined by $F(x) = (f_1(x), \ldots, f_n(x))$, where $f_i : \{0,1\}^n \longrightarrow \{0,1\}$ is called **local activation function** whose value depends on the values of the in-neighbours of node i, i.e., $f_i(x) = f_i(x_j,\, j \in N_G^-(i))$.

- An update schedule s of the vertices of G.

An **update schedule** on the vertices of G is a function $s : [\![1,n]\!] \to [\![1,n]\!]$ such that $s(V) = [\![1,m]\!]$ for some $m \leq n$. A **block** of s is the set $B_i = \{v \in V : s(v) = i\}$, $1 \leq i \leq m$. The **number of blocks** of s is denoted by $nb(s) \equiv m$. If $nb(s) = 1$, then s is said to be a **parallel** update schedule. In this case, we will write $s = s_p$. If s is a permutation over the set $[\![1,n]\!]$, i.e. $nb(s) = n$, s is said to be a **sequential** update schedule. In all other cases, i.e. when $2 \leq nb(s) \leq n-1$, s is said to be a **block sequential** update schedule. Frequently, s will be denoted by $s = (j \in B_1)(j \in B_2) \cdots (j \in B_{nb(s)})$ or more compactly $s = (B_i)_{i=1}^{nb(s)}$.

The **iteration** of a Boolean network $N = (G, F, s)$ is given by:

$$x_i^{r+1} = f_i(x_1^{l_1}, \ldots, x_j^{l_j}, \ldots, x_n^{l_n}), \tag{4.1}$$

where $l_j = r$ if $s(i) \leq s(j)$ and $l_j = r+1$ if $s(i) > s(j)$. The exponent r represents the time step.

This is equivalent to applying a function $F^s : \{0,1\}^n \to \{0,1\}^n$ 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 $N_1 = (G, F_1, s_1)$ and $N_2 = (G, F_2, s_2)$ have the same dynamics if $F_1^{s_1} = F_2^{s_2}$.

Since $\{0,1\}^n$ is a finite set, we have two limit behaviors for the iteration of a network:

- **Fixed Point**. We define a fixed point as $x \in \{0,1\}^n$ such that $F^s(x) = x$.

- **Limit Cycle**. We define a limit cycle of length $p > 1$ as the sequence $[x^0, \ldots, x^{p-1}, x^0]$ such that $x^j \in \{0,1\}^n$, x^j are pairwise distinct and $F^s(x^j) = x^{j+1}$, for all $j = 0, \ldots, p-2$ and $F^s(x^{p-1}) = x^0$.

Fixed points and cycles are called *attractors* of the network.

4.3.1 Sign-definite functions

A Boolean function $f : \{0,1\}^m \longrightarrow \{0,1\}$ is increasing monotone on input i if

$$\forall x \in \{0,1\}^m,\ x_i = 0,\ f(x) \leq f(x + e_i),$$

and decreasing monotone on input i if

$$\forall x \in \{0,1\}^m,\ x_i = 0,\ f(x) \geq f(x + e_i),$$

where $e_i \in \{0,1\}^m$ denotes the binary vector with all entries equal to 0, except for entry i, which equals 1. A Boolean function $f : \{0,1\}^m \longrightarrow \{0,1\}$ is said to be a sign-definite function, also known as unate function (Anthony, 1987), if for each $i = 1, \ldots, m$, is either increasing monotone or decreasing monotone on input i. Equivalently, a Boolean function is sign-definite if it can be represented by a formula in disjunctive normal form in which all occurrences of any given literal are either negated or nonnegated (Anthony, 1987).

A well-known example of non-sign-definite Boolean function is XOR, that is, $XOR(x_1, x_2) = \overline{x}_1 x_2 \vee x_1 \overline{x}_2$.

Given a sign-definite function $f : \{0,1\}^m \longrightarrow \{0,1\}$ we denote by $I^+(f)$ and $I^-(f)$ the set of indices where f is increasing monotone and decreasing monotone respectively.

From definition, every sign-definite function $f : \{0,1\}^m \longrightarrow \{0,1\}$ satisfies the following properties:

P1: For all vectors $x, y \in \{0,1\}^m$, with $x_i \leq y_i$ for all $i \in I^+(f)$ and $x_i \geq y_i$ for all $i \in I^-(f)$, $f(x) \leq f(y)$.

P2: For all vectors $x, y \in \{0,1\}^m$, with $x_i = 0$ for every $i \in I^+(f)$ and $x_i = 1$ for every $i \in I^-(f)$, $f(x) = 0$. Analogously, if $x_i = 1$ for all $i \in I^+(f)$ and $x_i = 0$ for all $i \in I^-(f)$, then $f(x) = 1$.

4.3.2 Regulatory Boolean networks

A Boolean network where each local activation function is a sign-definite function will be called a Regulatory Boolean Network (REBN).

From now and later, we will suppose w.l.o.g. that a REBN has not constant local activation functions; that is, for every function f there are vectors x, y such that $f(x) \neq f(y)$. And each local activation function f_i really depends on the values of its incident nodes, that is to say, $j \in N_G^-(i)$ if and only if

$$\exists x \in \{0,1\}^m,\ f_i(x_1, \ldots, x_j = 0, \ldots, x_m) \neq f_i(x_1, \ldots, \overline{x}_j = 1, \ldots, x_m),$$

where $m = |N_G^-(i)|$. Notice that if f_i is not a constant function, then $|N_G^-(i)| \geq 1$. It follows that there exists at least one cycle in G (you can even have a loop).

Thus, for all $i \in V(G)$, the set $\{I^+(f_i), I^-(f_i)\}$ is a partition of the set $N_G^-(i)$. Hence, for every REBN $N = (G, F)$ we can define a weight function
$w_F : A(G) \rightarrow \{-1, 1\}$ with

$$w_F(i,j) = -1 \text{ if } i \in I^-(f_j) \text{ and } w_F(i,j) = 1 \text{ if } i \in I^+(f_j).$$

(G, w_F) will be called signed digraph of N. In Fig. 5.1, an example of the REBN is depicted.

CHAPTER 5

Positive and negative feedback set problems

The problem of finding a minimum cardinality set of vertices that meets all the cycles of a directed graph (digraph) is known as Feedback Vertex Set problem and denoted by FVS. This problem was showed to be NP-complete by Karp (1972). There are many variants of this classical problem, some of them consider weights on the vertices or on the arcs. Almost all of them have been proved to be NP-complete as well (see Festa et al. (1999) for a good survey).

We study the complexity of new variants: Positive Feedback Vertex Set problem (PFVS) and Negative Feedback Vertex Set problem (NFVS). These problems are defined on a signed digraph (G, w), that is a digraph G with signs -1 or $+1$ on the arcs according to a sign function w. A cycle of (G, w) is called positive (negative) if it has an even (odd) number of negative arcs. A positive feedback vertex set of (G, w) is a set of vertices, which contains at least a vertex from every positive cycle in (G, w). A negative feedback vertex set of (G, w) is similarly defined. PFVS (NFVS) consists in finding a minimum cardinality positive (negative) feedback vertex set in a given signed digraph.

Feedback problems arise in numerous applications: circuit design, certain scheduling problems and cryptography are some examples (Festa et al., 1999). PFVS was introduced in Aracena (2001), and is of fundamental importance in the modeling of genetic regulatory networks by regulatory Boolean networks (REBN). The Boolean

[1]work published in: J. Aracena, A. Gajardo, M. Montalva, "On the complexity of feedback set problems in signed digraphs", Electronic Notes in Discrete Mathematics 30 (2008) 249-254.

networks (BN) were first introduced by Kauffman as mathematical model to study the dynamics of gene regulatory networks (Kauffman, 1969, 1993). A BN can be viewed as a digraph, where the vertices correspond to genes or gene products, while the arcs denote interactions among them. A gene expression level is modeled by binary values, 0 or 1, indicating two transcriptional states, either active or inactive, respectively. The state of each element of the network is determined by the application of a Boolean function to its inputs. The dynamical behavior of a BN is given by the update of element values in a synchronous way. Since a BN with n vertices has 2^n global states, from a starting state, within a finite number of udpates, the network will reach a fixed point or a limit cycle, called attractors. The attractors of a BN are often associated to distinct cell states defined by patterns of gene activity. A REBN is a BN where each interaction between the elements of the network corresponds either to a positive or to a negative interaction. Thus, the interaction digraph associated to a REBN is a signed digraph. Aracena proved that maximum number of fixed points of a REBN depends on the minimum cardinality positive feedback vertex set of the signed digraph associated (Aracena, 2001, 2008). On the other hand, the authors show in (Sontag et al., 2008) that as the number of independent negative feedback loops increases, the number of limit cycles of the REBN tends to decrease and its length tends to increase. In other words, the minimum cardinality of a negative feedback vertex set is related to the limit cycles in a REBN. In this way, the study of complexity of PFVS and NFVS is a key feature in the understanding of REBNs as well as an interesting theoretical problem.

In Section 5.2, we prove that PFVS and NFVS are both NP-hard by the construction of polynomial reductions from FVS to PFVS and NFVS. To show that all of these problems are in NP, we prove that the problems of existence of positive and negative cycles are both polynomial. This is achieved by showing their equivalence to the problems Even Cycle and Odd Cycle respectively. It is easy to see that Odd Cycle is polynomial (Hemaspaandra et al., 2004), hence any search algorithm in graphs can be used to search cycles of odd length. Nevertheless, the complexity of the problem of determining whether a given signed digraph has an even length cycle remained unknown for several decades. In 1989, Vazirani and Yannakakis (Vazirani and Yannakakis, 1989) proved that Even Cycle is polynomially equivalent to the problem of testing if a given bipartite graph has a Pfaffian orientation. This last problem was proved to be polynomial only in 1999 by Robertson, Seymour and Thomas (Robertson et al., 1999). Besides, in Section 5.2 we also consider PFVS and NFVS restricted to digraphs where the number of incoming arc in each vertex is bounded by a constant k. We show that for every $k \geq 2$, PFVS and NFVS remains NP-complete like it was exhibited in FVS (Garey and Johnson, 1979).

Since PFVS and NFVS are related to the number of attractors in REBNs, we considered in Section 5.3, the complexity of these problems in two important families of digraphs arises in the modeling of gene regulatory networks: Kauffman networks and locally monotone networks. Despite the restricted structural characteristics of these digraphs, we show that both problems are NP-complete.

Allthough FVS, PFVS and NFVS are NP-complete, the intuition is that there exist differences of complexity between them. In Section 5.4, we are interested in finding particular families of digraphs where the complexity of these problems is different, and thus to understand how the structural properties and sign distribution can determine the complexity of these problems. Here, we defined the problems PFVS-Nk and NFVS-Nk as the PFVS and NFVS problems restricted to the family of digraphs with at most k negative arcs. We proved that NFVS-Nk is polynomial, while PFVS-Nk is NP-complete. Besides, we show that for cyclically reducible digraphs, where it was showed that FVS is polynomial (Lloyd et al., 1985), with a given sign function, PFVS and NFVS are NP-complete.

5.1 Definitions and notations

Let $G = (V, A)$ be a digraph. A function $w : A \longrightarrow \{-1, +1\}$ is called a **sign function** on the arcs of G. The couple (G, w) is called **signed digraph**. A **signed subdigraph** of (G, w) is a couple (G', w') where G' is a subdigraph of G and $w' = w|_{G'}$. An arc $(i, j) \in A(G)$ will be called **positive** if $w(i, j) = +1$ and **negative** otherwise. We will say that a path is positive if the number of its negative arcs is even, and negative otherwise. A cycle C is called positive (negative) if the number of negative arcs in C is even (odd) (see Fig. 5.1). A vertex (An arc) set $U \subseteq V$ ($U \subseteq A$) is said a **positive feedback vertex (arc) set** of a signed digraph (G, w) if $(G-U, w|_{G-U})$ has no positive cycle. **Negative feedback vertex (arc) set** is similarly defined. For a vertex $v \in V$, we denote by $d^-_G(v)$ the number of in-neighbours of v, i.e. $d^-_G(v) = |N^-_G(v)|$, and is called the **in-degree** of v.

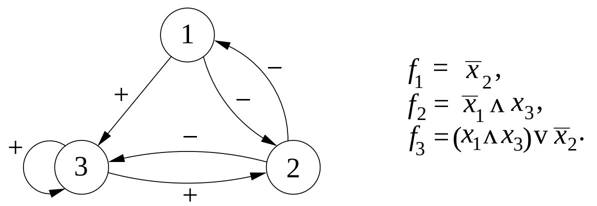


FIGURA 5.1. Example of REBN. $C_1 = [1, 3, 2, 1]$ and $C_2 = [1, 2, 1]$ are negative and positive cycles of the signed digraph of the network, respectively.

5.2 Positive and negative feedback sets

The following decision problems were introduced in Aracena (2001) and Montalva (2006):

> **PFVS (NFVS)**. Given a signed digraph (G, w) and given $t \in \mathbb{N}$, does a positive (negative) feedback vertex set U exist such that $|U| \leq t$?
>
> **PFAS (NFAS)**. Given a signed digraph (G, w) and given $t \in \mathbb{N}$, does a positive (negative) feedback arc set U exist such that $|U| \leq t$?
>
> **Positive (Negative) Cycle**. Given a signed digraph (G, w), is there a positive (negative) cycle in G?

Note that PFVS and NFVS are variant of FVS.

Given a signed digraph $(G = (V, A), w)$ and a vertex (an arc) set $U \subseteq V$ $(U \subseteq A)$ we can verify if U is a positive feedback vertex (arc) set by testing whether $(G - U, w|_{G-U})$ has or not positive cycles. Hence, if Positive Cycle is polynomial, then PFVS and PFAS are NP. Analogously, if Negative Cycle is polynomial, then NFVS and NFAS are NP.

Positive and Negative Cycle are closely related with the following known problems:

> **Even (Odd) Cycle**. Given a digraph G, is there a cycle of even (odd) length in G?

In this context, the following results were exhibited in Montalva (2006):

PROPOSITION 5.1 *Positive Cycle is polynomially equivalent to Even Cycle and Negative Cycle is polynomially equivalent to Odd Cycle.*

THEOREM 5.1 *PFVS is NP-complete and NFVS is NP.*

In order to continue this study, we have obtained the following results:

THEOREM 5.2 *NFVS is NP-complete.*

Proof. Let us define now a polynomial reduction from FVS to NFVS. Given a digraph G, we associate a signed digraph $(\widetilde{G}, w)$ obtained from G by associating a $+1$ sign to each arc of G and adding, for each arc of G, a two length path of negative sign (see Figure 5.2). In this way, the signed digraph $(\widetilde{G}, w)$ has both a positive and a negative cycle for each cycle of G, and every cycle of $(\widetilde{G}, w)$ corresponds to a unique cycle of G.

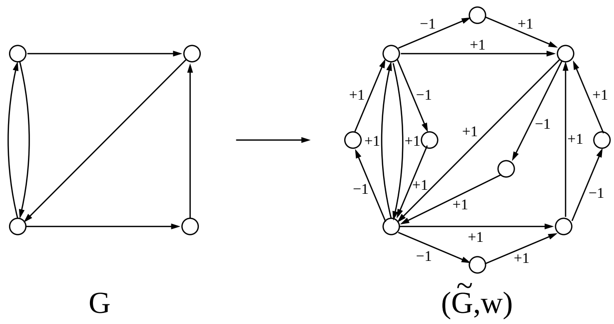


Figura 5.2. For each cycle of G, the signed digraph $(\widetilde{G}, w)$ has both a negative and a positive cycle.

Now, if we have a negative feedback vertex set for $(\widetilde{G}, w)$, it can be composed by vertices from G and some new vertices of $(\widetilde{G}, w)$. But each new vertex can be replaced by its unique incident vertex, which lies in the original vertex set, obtaining, in this way, a set with the same or smaller number of vertices, which is a feedback vertex set of G. Conversely, if U meets the cycles of G, it also meets the negative cycles of $(\widetilde{G}, w)$, which ends the proof. □

Observe that the reduction in the above proof works also to prove that FVS polynomially reduces to PFVS.

Theorem 5.3 *PFAS and NFAS are NP-complete.*

Proof. It is enough to show that PFVS and NFVS polynomially reduce to PFAS and NFAS respectively. Let us define the following reduction function: given a signed digraph $(G = (V, A), w)$, we define $\theta(G, w) = (G_{ST}, \tilde{w})$, where $(G_{ST}, \tilde{w})$ is as follows: for each vertex $v \in V$, $(G_{ST}, \tilde{w})$ has two new vertices v_s, v_t and a positive arc (v_t, v_s). For each arc $(x, y) \in A$, $(G_{ST}, \tilde{w})$ has the arc (x_s, y_t) with the same sign that the arc (x, y) (see Figure 5.3). In this way, there is a one to one relation between positive (resp. negative) cycles of (G, w) and positive (resp. negative) cycles of $(G_{ST}, \tilde{w})$.

On the one hand, if $S \subseteq V$ meets the positive (resp. negative) cycles of (G, w), then $\tilde{S} = \{(v_{i_t}, v_{i_s}) \in G_{ST} : v_i \in S\}$ meets the positive (resp. negative) cycles of $\theta(G, w)$. On the other, if we have a positive (resp. negative) feedback arc set for $\theta(G, w)$,

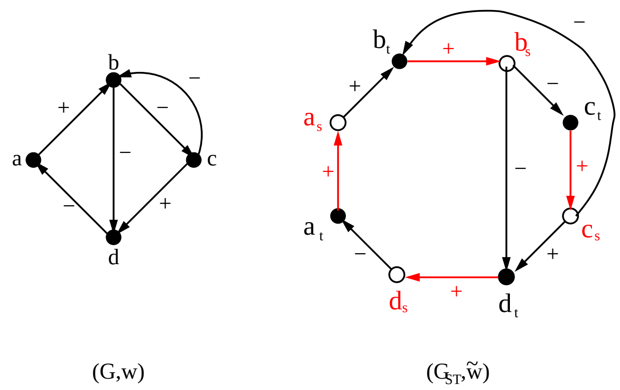


FIGURA 5.3. For each positive (resp. negative) cycle of (G, w), the signed digraph $(G_{ST}, \tilde{w})$ has a positive (resp. negative) cycle and viceversa.

it can be composed by arcs of the form (x_t, x_s) or (x_s, y_t). In the first case, they correspond to vertices in (G, w); in the second case, all the positive (resp. negative) cycles that contain it, also contain the arc (x_t, x_s). Then we can change each arc of the form (x_s, y_t) by (x_t, x_s). The vertices in (G, w) associated with these arcs constitute a positive (resp. negative) feedback vertex set for (G, w). We have simultaneously proved that the function θ is a polynomial reduction from PFVS to PFAS and from NFVS to NFAS. □

We also consider the complexity of PFVS and NFVS in signed digraphs with in-degree bounded by a constant. The following problems present constraints on the incoming arcs of each vertex, but the analysis is analogous when the constraints are on the outgoing arcs.

> **PFVS-k (NFVS-k)**. Given a signed digraph (G, w) with maximum in-degree bounded by k and given $t \in \mathbb{N}$. Does a positive (negative) feedback vertex set U exist such that $|U| \leq t$?.

Note that PFVS-1 and NFVS-1 are P, because in this case all the cycles of (G, w) are pairwise disjoints. The analogous problem, for unsigned digraphs (FVS-k) has already been studied, and it was proved that for $k \geq 2$ it is NP-complete (Garey and Johnson, 1979). In fact, it is not difficult to see that the in-degree of any network can be reduced by adding enough new vertices, as Figure 5.4 shows for the case where the digraph is signed. Here, a signed digraph (G, w) is reduced to a new signed digraph (G', w') where all its vertices have at most 2 incoming arcs. The new arcs (v_{r-2}, v) and (v_i, v_{i+1}) are all positive for $i = 1, ..., r-3$, while the arcs (w_1, v_1),(w_2, v_1),...,(w_{r-1}, v_{r-2}) have the sign of (w_1, v),(w_2, v),...,(w_{r-1}, v) respectively. In this way, PFVS (NFVS) polynomially reduces to PFVS-2 (NFVS-2).

Consequently, we have the following Theorem:

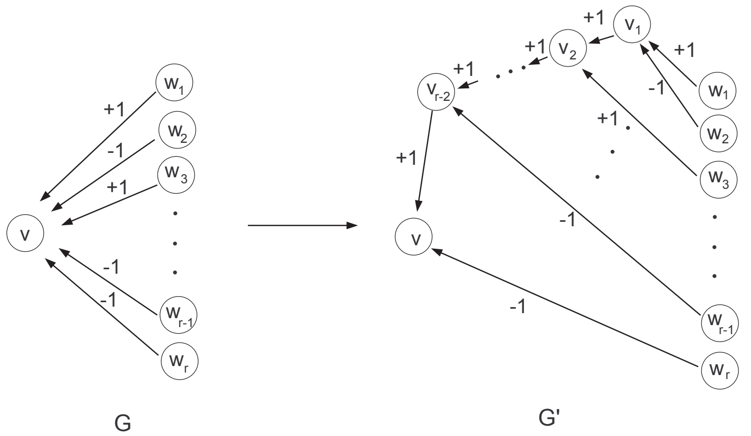


FIGURA 5.4. With this local transformation, the in-degree of each vertex v with $d_G^-(v) = r > 2$ is reduced, the signs of the cycles do not change and the minimum feedback vertex set is not modified. If a cycle of the new signed digraph is covered by some vertex v_i, it is also covered by v.

THEOREM 5.4 *PFVS-k and NFVS-k are both NP-complete for every* $k \geq 2$.

5.3 PFVS ans NFVS in families of digraphs with applied interest.

One of the most interesting applications of BNs is the study of dynamical properties of large-scale regulatory systems. The basic idea is to generate random BNs with local properties, for example networks with a fixed number k of incoming arcs in each node of the network (this kind of BNs is known as Kauffman's networks). By determining the attractors and trajectories in the state space, one can investigate the relationships between this kind of local properties and the global dynamics of the networks. Numerical simulations have shown that for low k and certain choices of local transition functions, BNs exhibited highly-ordered dynamics (Kauffman, 1993).

On the other hand, since the number of fixed points is related to the minimum cardinality positive feedback vertex set (Aracena, 2001, 2008), it is interesting to study the complexity of PFVS and NFVS in signed digraph with a fixed number k of incoming arcs in each vertex of digraph. In this way, we define the following problems:

> **PFVS=k (NFVS=k)**. Given a signed digraph (G, w) with in-degree exactly equal to k and given $t \in \mathbb{N}$, does a positive (negative) feedback vertex set U exist such that $|U| \leq t$?.

In Section 5.2 it is shown that PFVS and NFVS are NP-complete in the family of signed digraphs where the in-degree of each vertex is at most a constant $k \geq 2$. Next, we use these results to prove the NP-complexity when the in-degree of every vertex is exactly k.

THEOREM 5.5 *PFVS=k and NFVS=k are NP-complete for every $k \geq 2$.*

PROOF. If we have an arbitrary digraph G, from the last paragraph, we know how to transform it into a graph G' with in-degree less than or equal to k with the same minimum feedback vertex sets. We can eliminate the vertices with in-degree 0, since they do not participate in any cycle. Thus, we only need to modify the graph in order to augment the in-degree of the vertices with positive in-degree.

The following is a method to increase the in-degree of a vertex in one unit. Let v be a vertex with k in-neighbours: w_1, w_2,.., w_k. Let us add a new vertex v', k new arcs (w_i, v'), each with the same sign as (w_i, v), the positive arc (v', v) and the negative arc (v, v') (see Figure 5.5).

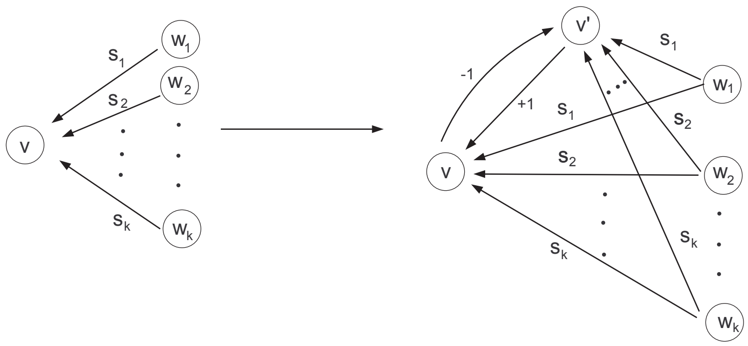


FIGURA 5.5. All the paths from w_i to v in the new digraph have sign s_i. Every positive cycle that passes by v', also passes by v.

This reduction adds a cycle: $vv'v$. But it is negative and its arcs cannot participate in any other cycle, thus the number of vertices of the minimum positive feedback vertex set is not modified. Let us remark that if we want to go from a vertex with only one in-neighbour to a vertex with k in-neighbours, we will need to add 2^{k-1} new vertices. If we want a digraph with equal negative feedback vertex set, we use the same reduction but assigning a $+1$ to the arc (v, v'). □

We remark that the same construction does not work for digraphs without signs. To our knowledge, it is an open problem to know whether the FVS problem is or not NP-complete in the family of digraphs with in-degree equal to k.

Another interesting class of REBNs, from a theoretical and biological point of view, is the family of locally monotone REBNs. In these networks, the Boolean function associated to each element is (increasing or decreasing) monotone. Hence, the interactions of each element with its input elements are either all positive or all negative. Therefore, the interaction digraph of a locally monotone REBN is a signed digraph where the incoming arcs of each vertex have the same sign. Despite of the structural constraints of these networks, NP-completeness is obtained within this class as well. In fact, any signed digraph can be transformed into a signed digraph with these characteristics, adding new vertices and arcs and without changing the number of vertices of the minimal feedback vertex sets. It is enough to separate, for each vertex v, the positive and negative arcs by connecting them to two new intermediary arcs, as Figure 5.6 shows. Then, we can establish the following Theorem:

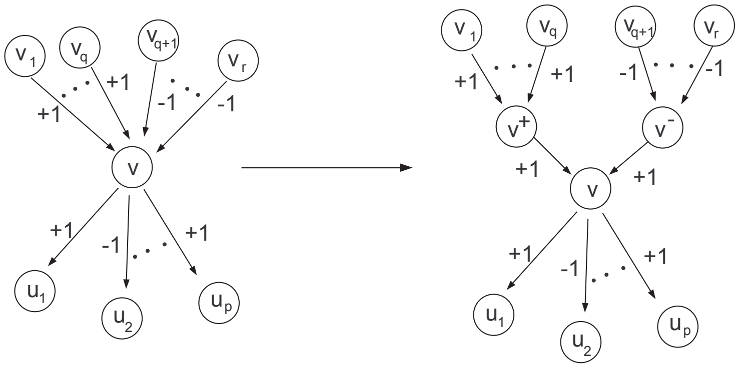


FIGURA 5.6. Two new vertices are added in order to separate the positive and negative incoming arcs of v. Every cycle that passes by v^+ or v^-, also passes by v, and the sign of the cycles is conserved.

THEOREM 5.6 *PFVS and NFVS are NP-complete in the family of sign digraphs where the incoming arcs of each vertex have the same sign.*

It is easy to see that the previous theorem also is true when the constraints are on the outgoing arcs.

5.4 Differences between FVS, PFVS and NFVS

In the most of the graph families that we have studied, FVS, PFVS and NFVS resulted to have the same complexity. Nevertheless, one could think that PFVS and NFVS may have a higher complexity because they are defined over a more complex object (a sign graph). For example, it would be not surprising that PFVS and NFVS were NP-complete in the family of signed cliques. But this is not true. Finding a digraph family

where FVS, PFVS and NFVS have different complexity was not easy.
In this section we give some examples.

REMARK 5.1 *Observe by simple inspection that all signed cliques* (K_3, w') *of a signed clique* (K_n, w) *with* $n \geq 3$ *have a positive cycle. Moreover, the only ones without negative cycles are showed in Figure 5.7 and denoted by* K_3^1 *and* K_3^2*. For this reason, if* F *is a positive feedback vertex set of a signed clique* (K_n, w)*, then:*

$$|F| = \begin{cases} n-2, & \text{if } (K_n, w) \text{ has a negative cycle of length 2,} \\ n-1, & \text{in other case} \end{cases}$$

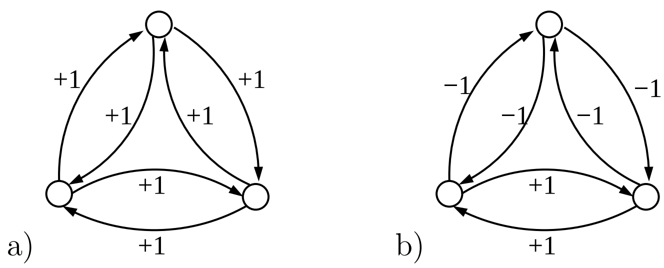


FIGURA 5.7. a) K_3^1. b) K_3^2.

On the other hand, an odd number of negative arcs in a signed clique (K_n, w) *shows the existence of negative cycles, because we can always find a cycle of length two with different signs in each one of its arcs.*

REMARK 5.2 *Let* (K_n, w) *be a signed clique with* $n \geq 3$ *without signed cliques like* K_3^1 *or* K_3^2*, and let* F *be a minimum (minimal) NFVS of* (K_n, w)*. Then,*

$$|F| = \begin{cases} n-2, & \text{if } (K_n, w) \text{ has a positive cycle of length 2,} \\ n-1, & \text{in other case} \end{cases}$$

In fact, by Remark 5.1, each signed clique (K_3, w') *of* (K_n, w) *has a negative cycle. Hence,* $|F| > n-3$ *and* $|F| = n-1$ *only when all two length cycles of* (K_n, w) *are negative.*

PROPOSITION 5.2 *Let* (K_n, w) *be a signed clique with* $n \geq 3$*. Then,*
(K_n, w) *has no negative cycle if and only if each signed clique* (K_3, w') *of* (K_n, w) *has no negative cycle.*

PROOF. $\Rightarrow$) It is straigthforward.

$\Leftarrow$) Let (K_n, w) be a signed clique with $n \geq 3$ such that each signed clique (K_3, w') of (K_n, w) has no negative cycle.

Let us prove the result by induction on the length l of a signed cycle C in (K_n, w).

Basis step, *l*=2. Every two length cycle $C = a, b, a$ is contained in some signed clique (K_3, w') and by hypothesis, (K_3, w') has no negative cycle. Therefore, C is positive.

Induction Hypothesis. Every signed cycle C in (K_n, w) such that $|V(C)| \leq l$, $l \leq n-1$, is positive.

Let $C' = x, v, y, ..., x$ be a signed cycle of (K_n, w) such that $|V(C')| = l+1$, $l \leq n-1$ (see Figure 5.8).

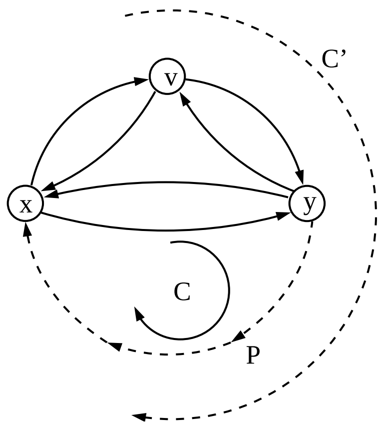


FIGURA 5.8. The cycle $C' = x, v, y, ..., x$ in G of length $l+1$ is decomposed in the paths $P' = x, w, y$ and P. Similary, C' consists of the arc (x, y) and P.

Since K_n is complete and by induction hypothesis, there exists a positive cycle $C = x, y, ..., x$ of length l in (K_n, w). That minds the path P from y to x must have the same sign as arc (x, y). Then, there are two possibilities. First, if the path P is positive, also is (x, y) and comparing the signed clique formed in x, w and y with the only signed cliques K_3 without negative cycles (i.e. cliques type K_3^1 or K_3^2, see Remark 5.1), necessarily $w(x, v) = w(v, y)$, i.e. the path $P' = x, v, y$ is positive with which the signed cycle C' formed by the positive paths P and P' also becomes positive. Second, if the path P is negative, also is (x, y) and doing the same above comparison, necessarily $w(x, v) = -w(v, y)$, i.e. the path $P' = x, v, y$ is negative. Hence, the cycle C' formed by the negative paths P and P' also becomes positive. □

In this way, to find a minimum (minimal) NFVS of a signed clique (K_n, w) is equivalent to find a maximum (maximal) signed clique of (K_n, w) without negative cycles, i.e. composed only by signed cliques of type K_3^1 or K_3^2. On the other hand, it is known that the problem of finding a maximum (maximal) clique in an undirected graph, denoted by MAX-CLIQUE, is NP-complete (Karp, 1972). Thus we have the following Theorem:

THEOREM 5.7 *PFVS is polynomial and NFVS is NP-complete for the family of signed cliques.*

PROOF. PFVS is P. In fact, due to Remark 5.1, in every signed clique with $n \geq 3$ vertices, at least $n-2$ vertices are necessary to cover all the positive cycles. Thus, by checking all sets with $n-2$ and $n-1$ vertices, we can obtain a minimum positive feedback vertex set. This last can be checked in $O(n^2)$ iterations.

NFVS is NP for the family of signed cliques. This is a direct consequence of the polynomial complexity of the Negative Cycle Problem.

MAX-CLIQUE polynomially reduces to NFVS for the family of signed cliques. In fact, let us define the reduction function as follows: given an undirected graph G, we define $\theta(G) = (G', w)$, where (G', w) is a signed clique with $|V(G)|$ vertices in which each arc $\{a, b\}$ of G is replaced by two positive arcs (a, b) and (b, a). If $\{x, y\} \notin A(G)$, then the positive arc (x, y) and the negative arc (y, x) are in $A(G')$ (see Figure 5.9). In this way, a given clique K of G is transformed into a signed clique (K', w') of (G', w) with all their arcs being positive (i.e. without negative cycles), both K and (K', w') having the same number of vertices.

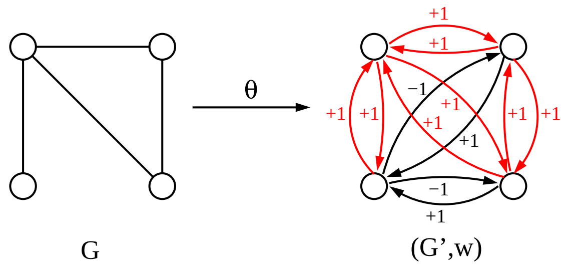


FIGURA 5.9. Each clique K of G is transformed into a signed clique (K', w') of (G', w) with all their arcs positive.

On the other hand, a signed clique (K', w') of (G', w) without negative cycles has only positive arcs due the definition of reduction θ, consequently corresponds to a unique clique K of G. $\square$

In order to have a negative cycle in a given signed digraph, negative arcs are necessary. Moreover, we can assert that the number of vertices of a minimum negative feedback vertex set is bounded by the number of negative arcs of the digraph. But this is not the case for positive cycles. The PFVS problem is NP-complete for digraphs with no negative arcs. This suggests that limiting the number of negative arcs have a different effect on the complexity of NFVS and that of PFVS. In order to precise this idea, let us consider the following decision problems:

> **PFVS-Nk (NFVS-Nk).** Given a signed digraph (G, w) with at most k negative arcs and given $t \in \mathbb{N}$, does a positive (negative) feedback vertex set U exist such that $|U| \leq t$?

PROPOSITION 5.3 *NFVS-Nk is polynomial and PFVS-Nk is NP-complete.*

PROOF. It is easy to see that PFVS-Nk is NP-complete because considering the family of signed digraphs without negative arcs (i.e. with $k = 0$), we can polynomially reduce FVS to PFVS-Nk by defining w as a constant sign function that assigns $+1$ to every arc of a given digraph G.

On the other hand, NFVS-Nk is polynomial because if $k \leq t$, the answer to the decision problem is yes. If $k > t$, it is enough to consider all the $\binom{n}{t}$ subsets $U \subseteq V(G)$ of t vertices and to verify whether some of them is a negative feedback vertex set; since $\binom{n}{t} \leq n^t \leq n^k$ this task has polynomial complexity in the input size. □

Consequently, even if PFVS and NFVS are NP-complete; there exist some families of digraphs where NFVS is easier than PFVS. In a similar way, there is a family where FVS is simpler than PFVS and NFVS.

In (Lloyd et al., 1985) the concept of *Cyclically Reducible* digraph is introduced and they proved that FVS is polynomial over this family. We prove that PFVS and NFVS are NP-complete in this case.

DEFINITION 5.1 *Given a digraph G, we say that node z is* **deadlocked** *if there is a (possibly trivial) path in G from z to some node y that lies on a cycle. The* **associated graph** *of node x with respect to G, $A(G, x)$, consists of node x and all nodes of G that are not deadlocked if x is removed from G. A* **D-sequence** *of a digraph G is a sequence of nodes (y_1,...,y_k) such that each of the graphs $A(G_{i-1}, y_i)$ have at least one cycle, where $G_0 = G$ and $G_i = G_{i-1} - A(G_{i-1}, y_i)$ for $1 \leq i \leq k$. Such a D-sequence is* **complete** *if the graph G_k (as defined above) is acyclic.*
G is **cyclically reducible** *if and only if there exists a complete D-sequence for G.*

The following result was exhibited in (Lloyd et al., 1985):

PROPOSITION 5.4 *Let G be a digraph. Then, there is a cyclically reducible digraph G' such that G is a subdigraph of G'.*

Proof. We can construct G' by doing the following for each node x in G: create a new two length directed cycle containing x and a new vertice x' (see Figure 5.10). Graph G' is cyclically reducible, since the nodes that were originally in G (in any order) form a complete D-sequence of G' for each node x originally in G, $A(G, x)$ contains both x and x' and has therefore at least one directed cycle. □

Let us consider the following problems:

> **PFVS-CR (NFVS-CR)**. Given a signed digraph (G, w), where G is cyclically reducible and, given $t \in \mathbb{N}$, does a positive feedback vertex set U exist such that $|U| \leq t$?

Theorem 5.8 *PFVS-CR and NFVS-CR are NP-complete.*

Proof. In fact, similar to the proof of Proposition 5.4, for each vertex x in a signed digraph (G, w), we can create a new vertex x', a positive arc from x to x' and a negative (positive) arc from x' to x (see Figure 5.10).

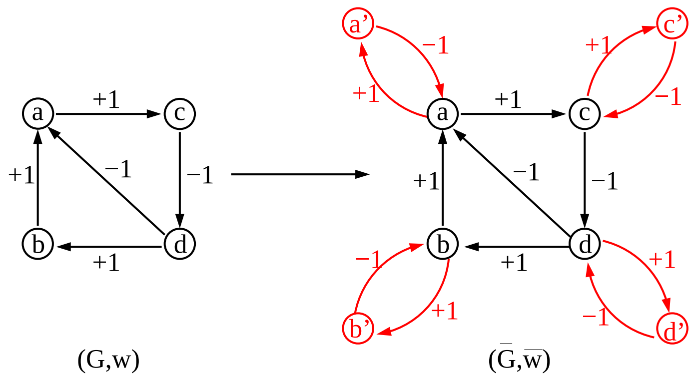


Figura 5.10. Polynomial reduction from PFVS to PFVS-CR.

The result is a signed digraph $(\overline{G}, \overline{w})$ with the same cycles of G plus $|V(G)|$ new two length negative (positive) cycles. Thus, the minimum positive (negative) feedback vertex set of $(\overline{G}, \overline{w})$ is equal to the minimum positive (negative) feedback vertex set of (G, w). Consequently, we have defined a polynomial reduction from PFVS (NFVS) to PFVS-CR (NFVS-CR). □

CHAPTER 6

Combinatorics on update digraphs in Boolean networks

Boolean networks (BNs) 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 this framework (Kauffman, 1969; Thomas, 1973; Shmulevich et al., 2003).

A BN is defined by 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 each set).

The impact of perturbations of the update schedule on a Boolean network dynamics have been greatly studied (Chaves et al., 2005; Elena et al., 2008; Ben-Amor et al., 2008; Demongeot et al., 2008; Elena, 2009), mainly from a statistical point of view and more recently, also from an analytical point of view (Salinas, 2008; Gómez, 2009).

Some analytical works on perturbations of update schedules have been made in a

[2]work published in: J. Aracena, E. Fanchon, M. Montalva, M. Noual, "Combinatorics on update digraphs in Boolean networks", Discrete Applied Mathematics 159 (6) (2011) 401-409.

[3]work submitted in: J. Aracena, J. Demongeot, E. Fanchon, M. Montalva, "On the number of update digraphs and its relation with the feedback arc sets and tournaments", Discrete Applied Mathematics, Ref. No. DA1309.

particular class of discrete dynamical networks, called sequential dynamical systems, where the connection digraph is symmetric or equivalently an undirected graph with a sequential update schedule. For this class of networks, the team of Hansson, 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 BN's according to the labeled digraph associated to the network (update digraph). It was proven that two update schedules in the same class yield exactly the same dynamical behavior.

We focus on the update digraphs and the number and size of equivalence classes of update schedules associated to a BN.

The main reason for our interest in update digraphs schedules is two-fold. On one hand, we wish to build a better understanding of the objects we are dealing with. On the other hand, we are interested in the relationships that exist between the architecture of the connection digraph of a discrete network and the robustness of its dynamics through the study of the equivalence classes of deterministic update schedules defined by its associated updated digraphs.

6.1 Definitions

Let s be an update schedule of the vertices of a digraph $G = (V, A)$ with $|V| = n$. We denote S_n the **set of update schedules** over $[\![1, n]\!]$. Besides, we denote by $P_s = \{B_1, ..., B_{nb(s)}\}$ the **partition** over $[\![1, n]\!]$ induced by s.

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

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

where $T_0 \equiv 1$.

Let G be a digraph. A function $lab : A(G) \to \{\ominus, \oplus\}$ is called a **label function** of G. An arc $a \in A(G)$ such that $lab(a) = \oplus$ is called a **positive arc** and an arc $a \in A(G)$ such that $lab(a) = \ominus$ is called a **negative arc**. A cycle C in G such that $\forall a \in A(C)$, $lab(a) = \oplus$ is called a **positive cycle** and a cycle C in G such that $\forall a \in A(C)$, $lab(a) = \ominus$ is called a **negative cycle**. Labeling every arc $a \in A(G)$ by $lab(a)$, we obtain a **labeled digraph** denoted by (G, lab).

Let s an update schedule of $V(G)$, we define the label function $lab_s : A(G) \to \{\circleddash, \oplus\}$ in the following way :

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

The labeled digraph (G, lab_s) is named **update digraph**. In this context, we say that s has the maximum number of blocks if for each update schedule s' such that $(G, lab_{s'}) = (G, lab_s)$, then $nb(s') \leq nb(s)$ (see example in Fig. 6.1).

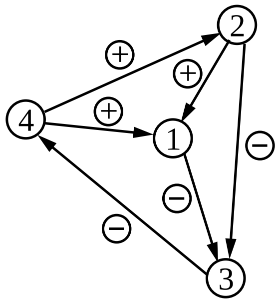


FIGURA 6.1. A digraph $G = (V, A)$ labeled by the function lab_s where $\forall i \in V = \{1, \ldots, 4\}$, $s(i) = i$. In this case s has the maximum number of blocks.

6.2 Preliminary results and motivations

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

THEOREM 6.1 *Let $N_1 = (G, F, s_1)$ and $N_2 = (G, F, s_2)$ be two Boolean 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.*

We 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

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

An equivalence class, $[s]_G$, is a set of update schedules that all yield the same labeled digraph, and consequently by Theorem 6.1, the same dynamics on networks.

In this work we study update digraphs and the equivalence classes of their update schedules. More precisely, Section 6.3 deals with the characterization of update digraphs. Sections 6.4 and 6.5 focus on the size and the number of equivalence classes of update schedules.

6.3 Characterization of update digraphs

In this section, we study the relation $\overset{G}{\sim}$ and the labelings of a given digraph G. First, we give a characterization of the label functions $lab : A(G) \to \{\ominus, \oplus\}$ that indeed correspond to label 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 to determine the number of $[\cdot]_G$ classes. First, let us give some additional definitions.

Definition 6.1 *Let $lab : A(G) \to \{\ominus, \oplus\}$ be a label function of a given digraph G. The labeled digraph (G, lab) is said to be an* **update digraph** *if there exists an update schedule s such that $lab = lab_s$, that is $\forall a \in A(G)$, $lab(a) = lab_s(a)$. We denote by $U(G) = \{lab : A(G) \to \{\ominus, \oplus\} | \ (G, lab)$ is an update digraph$\}$. An update digraph (G, lab) has a* **maximal number of negative arcs** *if there is no label function $lab^* \in U(G)$ with strictly more negative arcs that lab, and where $lab(u, v) = \ominus$ implies $lab^*(u, v) = \ominus$, for every $(u, v) \in A(G)$ (see example in Fig. 6.2).*

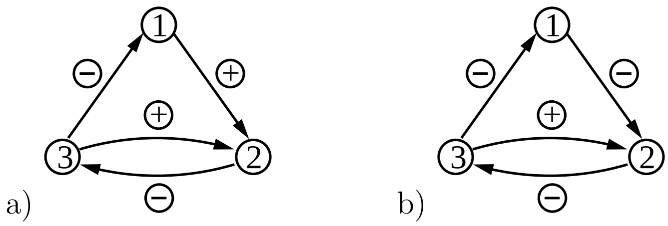


Figura 6.2. a) A labeled digraph (G, lab) which is an update digraph with maximal (but not maximum) number of negative arcs. b) A labeled digraph (G, lab') which is not an update digraph.

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

Definition 6.2 *Let (G, lab) be a labeled digraph and G' a subdigraph of G. We define the* **projection** *of (G, lab) onto G' as being the labeled digraph $(G', lab_{G'})$, where $lab_{G'}(a) = lab(a)$, $\forall a \in A(G')$.*

Definition 6.3 *Let (G, lab) be a labeled digraph. G' is said to be a* **positive strongly connected component** *of (G, lab) if G' is a strongly connected induced subdigraph of G with all its arcs positive in (G, lab) and is maximal for this property. If $V(G) = \{x\}$, then G is a positive strongly connected component of (G, lab) called* **trivial component**. *We will say that (G, lab) is* **reduced** *if it has no positive strongly connected component not trivial.*

Note that the fact that (G, lab) is an update digraph is independent of the presence or absence of positive strongly connected components, because the images $s(i)$ of the vertices i by an update schedule in a positive strongly connected component are equal. For our study, they can thus be replaced by one unique vertex.

DEFINITION 6.4 *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:*

- $V(G_R) = V(G)$.
- $A(G_R) = \begin{array}{l} \{(u,v)|\,(u,v) \in A(G) \,\wedge\, lab(u,v) = \oplus\} \\ \cup\ \{(u,v)|\,(v,u) \in A(G) \,\wedge\, lab(v,u) = \ominus\}. \end{array}$
- $\forall (u,v) \in A(G_R),\ lab_R(u,v) = \begin{cases} \ominus & \text{if } (v,u) \in A(G) \wedge lab(v,u) = \ominus, \\ \oplus & \text{otherwise.} \end{cases}$

A forbidden cycle in (G_R, lab_R) is a cycle containing a negative arc. A forbidden circuit in (G, lab) is a circuit that represents a forbidden cycle in (G_R, lab_R).

An example of labeled reoriented digraph is shown in Fig. 6.3.

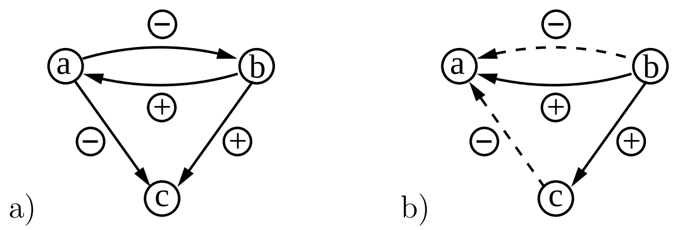


FIGURA 6.3. *a)* A labeled digraph $G = (\{a,b,c\}, A)$. *b)* (G_R, lab_R) where the arcs drawn in dotted lines are the ones that have been inverted.

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 associated to (G, lab) without its negative arcs. We also can get (G_R, lab_R) in time $\mathcal{O}(|A|)$.

DEFINITION 6.5 *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 the set $\mathcal{P}_v$ of paths in (G_R, lab_R) ending in v, and we denote $L^-(v) = \max_{P \in \mathcal{P}_v} l^-(P)$ and*

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

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

THEOREM 6.2 *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 $C : v_1, \dots, v_p = v_1$ 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 $P = [v_1, \dots, v_k]$ is a path in G_R such that $l^-(P) = L$ with $\{(v_{i_1}, v_{i_2}), (v_{i_3}, v_{i_4}), \dots, (v_{i_{2L-1}}, v_{i_{2L}})\}$ 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,

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

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

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

We observed above that $s(V(G)) = [\![1, L+1]\!]$, meaning that 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 $\forall a = (u, v) \in A(G_R),\ s(u) > s(v) \Leftrightarrow lab_G(u, v) = \ominus$. This follows from the fact that (u, v) being an arc of G_R, it necessarily holds that $L^-(v) \geq 1 + L^-(u)$ when $lab_G(u, v) = \ominus$. □

We 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, $C = v_0, v_1, \dots, v_k$, 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.

Incidentally, let us notice that, as a consequence of Theorem 6.2, 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 finds an update schedule corresponding to a given reduced labeled digraph as described in the proof of Theorem 6.2. It is adapted from the famous algorithm by 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

$$nb(s) = min\{nb(s') \mid s' \text{ is an update schedule of } G\}.$$

Algorithm 1 Update schedule associated to a labeled digraph.

Require: $(G = (V, A), lab)$ a reduced labeled digraph such that (G_R, lab_R) has no forbidden cycle.

`ValMax` $\leftarrow$ table of size $|V(G_R)|$ in which are stored the maximal possible values of $s(v),\ v \in V(G_R)$.

$n \leftarrow |V|$;

$H \leftarrow G_R$;

for $v \in V$ **do**
  `ValMax`$[v] = n$;
end for

while $\exists v \in V,\ N_H(v) = \emptyset$ **do**
  $s(v) \leftarrow$ `ValMax`$[v]$;
  for $(v, w) \in A(H)$ **do**
    if $(w, v) \in A(G)$ is a negative arc **then**
      `ValMax`$[w] \leftarrow min\{$`ValMax`$[w], s(v) - 1\}$;
    else
      `ValMax`$[w] \leftarrow min\{$`ValMax`$[w], s(v)\}$;
    end if
    delete the arc (v, w) from H;
  end for
end while

$s_{min} \leftarrow min\{s(v \mid v \in V)\}$;
for $v \in V$ **do**
  $s(v) \leftarrow s(v) - s_{min} + 1$;
end for

Figure 6.4 shows the different steps of the algorithm that returns an update schedule associated to an arbitrary possible labeled digraph (not necessarily reduced).

Corollary 6.1 *The following problems can be solved in polynomial time.*

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

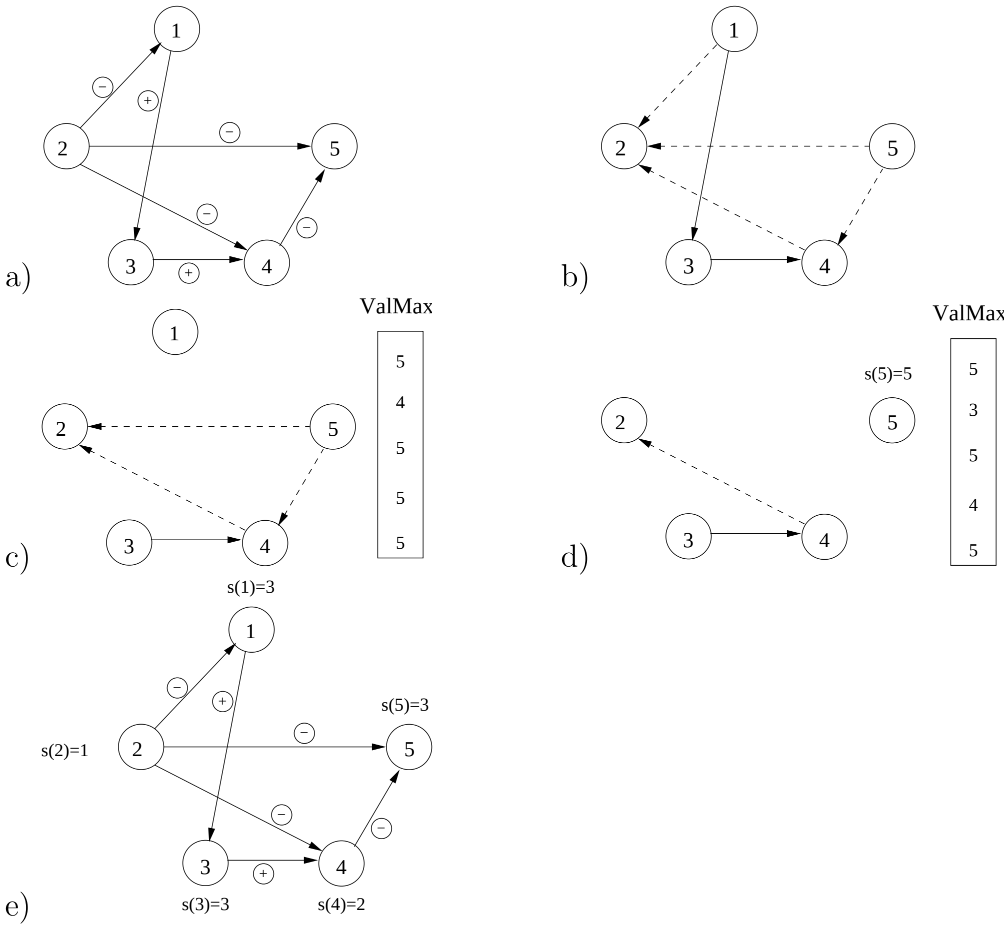


FIGURA 6.4. *a*) A labeled digraph $G = (\{1,\ldots,5\}, A)$. *b*) (G_R, lab_R). The arcs drawn in dotted lines are negative-arcs. The others are positive-arcs. *c*) and *d*) are the first two steps computed by algorithm 1 after the *while loop*. *e*) The update schedule s such that $(G, lab) = (G, lab_s)$.

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

Indeed, according to Theorem 6.2, 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 6.1 holds since the strongly connected components of a digraph can be identified in polynomial time. For the second one, an update schedule s such that $(G, lab) = (G, lab_s)$ can be constructed by using the algorithm 1 whose run time is also polynomial.

On the other hand, given a non-update digraph, it is natural to ask which modifications we can do to obtain an update digraph. For this, we introduce the concept of *update feedback arc set* as follows.

DEFINITION 6.6 *Let $(G = (V, A), lab)$ be a labeled digraph. $F \subseteq A$ is an* **update arc set** *of G if $(G - F, lab_{G-F})$ is an update digraph (see Fig. 6.5).*

DEFINITION 6.7 *We define the* **Update Arc Set** *problem as follows:*

UAS: *Let $(G = (V, A), lab)$ be a labeled digraph and $k \in \mathbb{N}$. Is there $F \subseteq A$ with $|F| \leq k$ such that F is an update arc set of G?*

PROPOSITION 6.1 *UAS is NP-complete*

PROOF. It is easy to see that FAS can be polynomially reduced to UAS, where the reduction consists to label with $\ominus$ all arcs of G. $\square$

PROPOSITION 6.2 *Let $(G = (V, A), lab)$ be a non-update digraph and $F \subseteq A$ a minimal update arc set of G. Then $(G, \overline{lab}^F)$ is an update digraph, where for each $a \in F$, $\overline{lab}^F(a) = \ominus \Longleftrightarrow lab(a) = \oplus$ and $\overline{lab}^F(a) = lab(a)$, $\forall\, a \in A - F$.*

PROOF. Let $(G = (V, A), lab)$ be a non-update digraph and $F \subseteq A$ a minimal update arc set of G. Then by definition of minimal update arc set, for each $a \in F$ there is a forbidden circuit C_a such that $A(C_a) \cap F = \emptyset$ and by definition of F, $(G - F, lab_{G-F})$ is an update digraph. Thus by projection theorem $((G - F) \cup \{a\}, \tilde{lab}_{(G-F)\cup\{a\}})$ is an update digraph (because C_a is a forbidden circuit when the label of a is $lab(a)$). $\square$

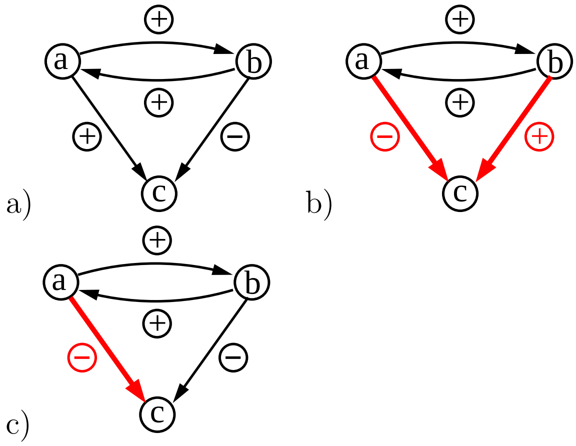


FIGURA 6.5. *a)* A non-update digraph (G, lab) where $C = [a, c, b, a]$ with $A(C) = \{(a, c), (b, c), (b, a)\}$ is a forbidden circuit. The set $U = \{(a, c), (b, c)\}$ is an update arc set of (G, lab) which is not minimal. The set $U_{min} = \{(a, c)\}$ is a minimal update arc set of (G, lab). *b)* The non-update digraph $(G, \overline{lab}^{U})$ obtained from (G, lab) by changing the labels of U. *c)* The update digraph $(G, \overline{lab}^{U_{min}})$ obtained from (G, lab) changing the labels of U_{min} as in Proposition 6.2.

6.4 Size of the equivalence classes $[\cdot]_G$

Let us now consider the following question : given a digraph G and an update schedule s, does there exist any update 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 6.2 *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 $nb(s) = m + 1$ and $(G, lab) = (G, lab_s)$.*

PROOF. We show the result by induction on m.

If $m = L$, the result was proved in Theorem 6.2.

If $L = n-1$, the proof is done. Otherwise, let $m \in]\!]L, n-1]\!]$. By induction hypothesis, there exists an update schedule $s = (B_i)_{i=1}^{nb(s)}$ such that $(G, lab) = (G, lab_s)$ and $nb(s) = m$. Since $m < n$, there exists $i^* \in [\![1, n-1]\!]$ such that $|B_{i^*}| > 1$. Notice that $\forall (u, v) \in (B_{i^*} \times B_{i^*}) \cap A(G)$, $lab_s(u, v) = \oplus$. Besides, because there are not cycles in (G_R, lab_R), there exists $w \in B_{i^*}$ such that $\{v \in B_{i^*} | \; (w, v) \in A(G_R)\} = \emptyset$. 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 6.3 *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 $(G, lab) = (G, lab_s)$.*

Proposition 6.3 *Let (G, lab) be an update digraph. The following assertions are equivalent:*

(a) There is a sequential update schedule s_q such that $(G, lab) = (G, lab_{s_q})$.

(b) (G, lab) has no positive cycle.

(c) (G, lab) is reduced.

Proof. $(a) \Rightarrow (b)$ Straightforward.

$(b) \Rightarrow (c)$ Let (G, lab) be an update digraph without positive cycles. Then (G, lab) has no positive strongly connected component, i.e. (G, lab) is a reduced update digraph.

$(c) \Rightarrow (a)$ Let (G, lab) be a reduced update digraph. By Corollary 6.2, $\forall m \in [\![L, n-1]\!]$, there exists an update schedule s such that $nb(s) = m + 1$ and $(G, lab) = (G, lab_s)$. In particular, for $m = n - 1$, there exists an update schedule s_q such that $nb(s_q) = (n-1)+1 = n$ (i.e. s_q is a sequential update schedule) and $(G, lab) = (G, lab_{s_q})$. □

Remark 6.1 *In particular, if (G, lab) is a labeled acyclic digraph, then there is always a sequential update schedule s_q such that $(G, lab) = (G, lab_{s_q})$.*

Corollary 6.4 *Let (G, lab) be a reduced update digraph with $|V(G)| = n$. Then, there exists $s_1 \neq s_2$ sequential update schedules such that $(G, lab) = (G, lab_{s_1}) = (G, lab_{s_2})$ if and only if the longest path in (G_R, lab_R) has at most $n - 2$ arcs.*

Proof. Let (G, lab) be a reduced update digraph with $|V(G)| = n$. Then, by Corollary 6.2, there exists sequential update schedule s_q, such that $(G, lab) = (G, lab_{s_q})$.

$\Rightarrow$) If the longest path in (G_R, lab_R) has $n - 1$ arcs, then s_q is the only sequential update schedule compatible with the path, because for every other sequential update schedule s there is $(u,v) \in A(G)$ such that $lab_{s_q}(u,v) \neq lab_s(u,v)$ and consequently, $(G, lab) = (G, lab_{s_q}) \neq (G, lab_s)$.

$\Leftarrow$) Let $s_1 = s_q = (v_1)\cdots(v_n)$, $v_i \in V(G)$, $\forall i \in [\![1,n]\!]$. Hence, $\forall (v_i, v_j) \in A(G)$, $i, j \in [\![1,n]\!]$,

$$i < j \Leftrightarrow lab_{s_1}(v_i, v_j) = \ominus$$

On the other hand, $A(G_R) = \{(v_j, v_i) : j > i \wedge (v_j, v_i) \in A(G) \vee (v_i, v_j) \in A(G), i, j \in [\![1,n]\!]\}$ and since the longest path in $(G_R, (lab_{s_1})_R)$ has at most $n-2$ arcs, necessarily there exists $i \in [\![1,n]\!]$, such that $(v_{i+1}, v_i) \notin A(G_R)$, this implies the existence of another sequential update schedule $s_2 \neq s_1$ which changes the block $B_i = \{v_i\}$ by the block $B_{i+1} = \{v_{i+1}\}$ of s_1 and vice versa, i.e. $s_2 = (v_1)\cdots(v_{i-1})(v_{i+1})(v_i)(v_{i+2})\cdots(v_n)$, such that $(G, lab) = (G, lab_{s_1}) = (G, lab_{s_2})$. □

Corollary 6.5 *Let $(G = (V, A), lab_s)$ be a reduced update digraph. $|[s]_G| > 1$ if and only if (G_R, lab_R) is not a negative linear digraph.*

Proof. If (G_R, lab_R) is not a negative linear digraph, i.e. it has not a directed path of length $|V|-1$ with all its arcs negative, then $L \leq |V|-2$. Thus, by Corollary 6.3, $|[s]_G| > 1$, where $(G, lab) = (G, lab_s)$.

Conversely, if G is a negative linear digraph with $p = [u_1, \ldots, u_{|V|}]$ a directed path of length $|V|-1$ with $lab_G(u_i, u_{i+1}) = \ominus$, $\forall i = 1\ldots, |V|-1$, then there only exists one update schedule s which satisfies $(G, lab) = (G, lab_s)$. □

As a consequence, $|[s_p]_G| > 1$ if and only if G is not strongly connected.

6.5 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 label 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 on their sizes).

Definition 6.8 *Let G be a digraph. We denote by $NU(G) = \{lab : A(G) \longrightarrow \{\ominus, \oplus\} : (G, lab)$ is a non update digraph$\}$, $FAS(G) = \{F \subseteq A(G) : F$ is a feedback arc set of $G\}$, $NFAS(G) = \{F \subseteq A(G) :$*

F is a non feedback arc set of $G\}$, *and* $MFAS(G) = \{F \subseteq A(G) :$ *F is a minimal feedback arc set of* $G\}$. *Note that*

$$|U(G)| = |\{[s]_G \mid s \text{ is an update schedule over } V(G)\}|$$

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 label function` $lab : A \to \{\oplus, \ominus\}$ `such that` (G, lab) `is an update digraph and its size is at most` k `?`

THEOREM 6.3 *DIGRAPH UPDATE is NP-complete.*

PROOF. We are going to prove Theorem 6.3 by reduction to the FAS problem which is remembered below and 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` $|F| \leq k$ `?`

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 label function lab such that (G, lab) is an update digraph of size at most k if and only if there exists a feedback arc set F of G such that $|F| \leq k$.

($\Rightarrow$) Let lab be a label 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 feedback arc set of size $|F| \leq k$. $G' = (V, A - F)$ 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 feedback arc set 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 $F - \{a\}$ is a feedback arc set 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 label function lab as follows:

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

Note that because there are no cycles in $G' = (V, A - F)$, there are not negative 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 6.2 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 $G' = (V, A(G) - F)$ (see Fig. 6.6 below)

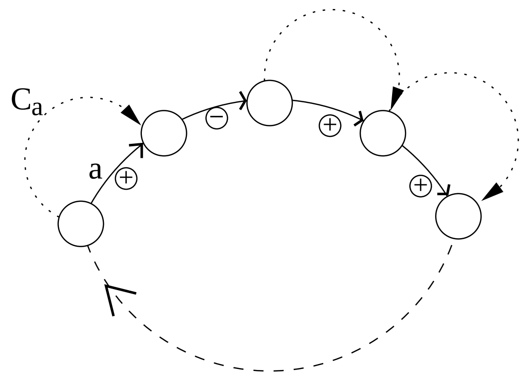


FIGURA 6.6. A forbidden cycle in (G_R, lab_R) with, surrounding it, the negative cycles C_a mentioned in the proof of Theorem 6.3. Arrows in full line represent arcs, arrows in dashed lines represent paths.

which contradicts F being a feedback arc set of G (as well as the fact that (G, lab) has no negative cycles). $\square$

COROLLARY 6.6 *Let G be a digraph. Then,*

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

PROOF. From the above proof, we can deduce that

$$|MFAS(G)| \;\leq\; |U(G)| \;\leq\; |FAS(G)|$$

i.e. the function $g : MFAS(G) \to U(G)$ such that $F \to g(F) = lab_F$ is injective. But for the label function lab_p such that $lab_p(e) = \oplus$, $\forall e \in A(G)$ there is no pre-image $F \in MFAS(G)$ such that $g(F) = lab_p$, because the only possibility would be $F = A(G)$, but a minimal feedback arc set F' has the property that each of its arcs meets a cycle that contains no other arc of F' and this is impossible with $F = A(G)$. Therefore, $|MFAS(G)| < |U(G)|$. $\square$

An example of digraph G where the number of update digraphs is distinct of the number of feedback arc sets and minimal feedback arc sets is shown in Fig. 6.7. For this digraph, $|MFAS(G)| = 3$, $|FAS(G)| = 11$ and the number of associated update digraphs is six.

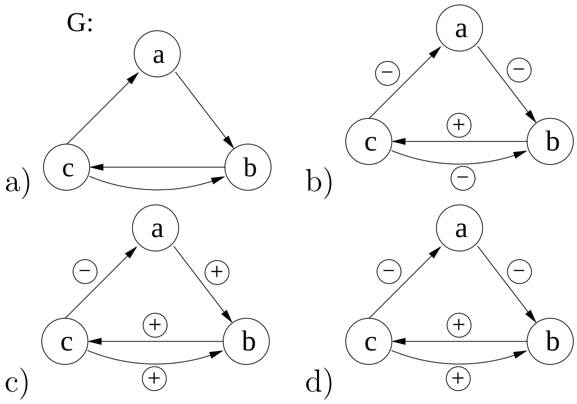


FIGURA 6.7. *a*) A digraph G. *b*) An update digraph, where $\{(b, c)\}$ is a minimal feedback arc set. *c*) An update digraph, where $\{(a, b), (b, c), (c, b)\}$ is a feedback arc set but not minimal. *d*) A non update digraph, but $\{(b, c), (c, b)\}$ is a feedback arc set.

6.5.1 Extensions and projections of update digraphs

We see how the property of update digraph can be inherited into local structures such as subdigraphs. Thus, add or remove labeled arcs of a given update digraph leads to keep the property of being an update digraph.

THEOREM 6.4 *Let G be a digraph and G' a subdigraph of G. If (G', lab') is an update digraph, then there exists a label function of $A(G)$ lab such that (G, lab) is an update digraph and $lab_{G'} = lab'$.*

PROOF. If (G', lab') is an update digraph we will show that for all $a = (u, v) \in A(G) - A(G')$, 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 $a = (u, v) \in A(G) - A(G')$ such that neither $(G' + a, lab_a^+)$ nor $(G' + a, lab_a^-)$ are update digraphs. Then there exists 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, the 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. 6.8), that is a forbidden cycle. Thus (G', lab') is not an update digraph, which is a contradiction.

Therefore, if $A(G) - A(G') = \{a_1, \ldots, a_r\}$, then by induction we can prove that for all k in $\{1, \ldots, r\}$ there exists a label 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 label function lab in G such that (G, lab) is an update digraph and $lab' = lab_{G'}$.
□

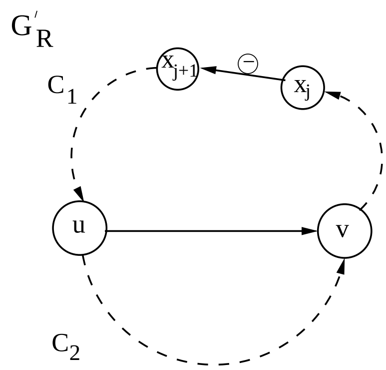


FIGURA 6.8. Scheme of the forbidden cycle in (G'_R, lab'_R) mentioned in the proof of Theorem 6.4.

Note that if (G, lab) is an update digraph and $lab' = lab_{G'}$, then (G', lab') is also an update digraph by Theorem 6.2. Therefore, the update subdigraphs are the projections of the update digraphs.

Now, we see how the number and size of the equivalence classes are bounded by a function that depends on the number of update schedules T_n for a given digraph G such that $|V(G)| = n$.

COROLLARY 6.7 *Let G be a connected digraph of $n > 1$ vertices. Then,*

$$2^{n-1} \ \leq \ |U(G)| \ \leq \ T_n$$

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

PROOF. From Theorem 6.4 for all digraphs G and subdigraphs G' of G,

$$|U(G')| \ \leq \ |U(G)|$$

On other hand, the connected digraph of n vertices with the least number of arcs, i.e. $n-1$, 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 great number of arcs, equal to n^2 (including the arcs (u,u)), is a complete digraph. In this case, each label function on a complete digraph defines 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. □

REMARK 6.2 *If G is a complete digraph with $|V(G)| = n$, then $|U(G)| = |S_n| = T_n$ and consequently $|[s]_G| = 1$ for every $s \in S_n$ because for every pair of update schedules s_1, s_2 with $s_1 \neq s_2$ there exist i, j such that $s_1(i) < s_1(j)$ and $s_2(i) \geq s_2(j)$, i.e. $lab_{s_1}(i,j) \neq lab_{s_2}(i,j)$. Besides, if G is a reduced digraph, the above mentioned result and Corollary 6.5 imply that the reoriented update digraph $(G_R, (lab_s)_R)$ is a negative linear digraph.*

DEFINITION 6.9 *Let G be a digraph. We define $P_G = \{[s]_G : s \in S_n\}$ and denote $P_G \precsim P_{G'} \Leftrightarrow \forall [s]_{G'} \in P_{G'}, \exists [\tilde{s}]_G \in P_G, [s]_{G'} \subseteq [\tilde{s}]_G$. We denote $P_G \precnsim P_{G'} \Leftrightarrow P_G \neq P_{G'} \wedge P_G \precsim P_{G'}$.*

PROPOSITION 6.4 *If $G \subsetneq G'$, then $P_G \precnsim P_{G'}$, where $V(G) = V(G') = n$.*

PROOF.

$$\begin{aligned} P_G \precnsim P_{G'} &\Leftrightarrow P_G \neq P_{G'} \wedge P_G \precsim P_{G'} \\ &\Leftrightarrow P_G \neq P_{G'} \wedge \forall [s]_{G'} \in P_{G'}, \exists [\tilde{s}]_G \in P_G, [s]_{G'} \subseteq [\tilde{s}]_G. \end{aligned}$$

It is easy to see that for all $s \in S_n$, $[s]_{G'} \subseteq [s]_G$, i.e $P_G \precsim P_{G'}$.

On the other hand, $G \subsetneq G' \Rightarrow \exists (a,b) \in A(G') - A(G)$.
Let $s, s' \in S_n$ be such that $s(a) = s(b) = s'(a) = 1$, $s'(b) = 2$, $s(c) = 2$ and $s'(c) = 3$, $\forall c \in V(G) - \{a,b\}$. Then $[s]_{G'} \neq [s']_{G'}$ but $[s]_G = [s']_G$, i.e $P_G \neq P_{G'}$. □

COROLLARY 6.8 *Let G be a digraph and $G' \subsetneq G$ where $V(G) = V(G')$. Then $|U(G')| < |U(G)|$ and $|FAS(G')| < |FAS(G)|$.*

PROOF. By Proposition 6.4, $P_G \precnsim P_{G'}$, i.e $|P_G| < |P_{G'}|$ but $|P_G| = |U(G)|$. Hence, $|U(G')| < |U(G)|$.

On the other hand, the function $g : FAS(G') \to FAS(G)$, $F' \to g(F') = F' \cup (A(G) - A(G'))$, is evidently injective and well-defined. Thus, $|FAS(G')| \leq |FAS(G)|$. But for a given $(a, b) \in A(G) - A(G')$, $F = A(G) - \{(a, b)\} \in FAS(G)$, however, there is no $F' \in FAS(G')$ such that $g(F') = F$. Hence, g is not surjective and in this way. $|FAS(G')| < |FAS(G)|$. □

6.5.2 Feedback arc sets and update digraphs

As we saw at the beginning of this section, there are links between feedback arc sets and update digraphs. Next, we exhibit some additional results showing increasing and decreasing monotone properties for $|U(G)|$ and $|FAS(G)|$ respectively, where G is a given digraph.

THEOREM 6.5 *Let G be an undirected graph and G_1 and G_2 two orientations of G such that every cycle of G_1 is also a cycle of G_2. Then $|U(G_1)| \leq |U(G_2)|$ and $|FAS(G_2)| \leq |FAS(G_1)|$.*

PROOF. Let G, G_1 and G_2 as in the hypothesis of the Theorem. In particular, note that:

$$\forall (u, v) \in A(G_1),\ (u, v) \in A(G_2) \vee (v, u) \in A(G_2)$$

We define the function $f : NU(G_2) \longrightarrow NU(G_1)$, $lab \rightarrow f(lab) = lab'$ by:

$$lab'(u, v) = \begin{cases} lab(u, v), & \text{if } (u, v) \in A(G_1) \cap A(G_2), \\ \overline{lab}(u, v), & \text{in other case.} \end{cases} \qquad , \forall (u, v) \in A(G_1)$$

where $\overline{lab} : A(G_1) \longrightarrow \{\ominus, \oplus\}$ is defined by:

$$\overline{lab}(u, v) = \oplus \Leftrightarrow lab(v, u) = \ominus$$

f is well defined. First, note from definition of lab' that $(G_2)_R = (G_1)_R$. In this way, if C is a cycle in $((G_2)_R, lab_R)$ with some negative arc, then C is also a cycle in $((G_1)_R, lab'_R)$ with some negative arc (see Fig. 6.9).

In fact, suppose on the contrary that C is a cycle in $((G_1)_R, lab'_R)$ with all its arcs positive. This is possible if and only if C is a cycle in (G_1, lab') with all its arcs positive and by hypothesis of the Theorem, C is also a cycle in (G_2, lab) with all its arcs positive, which is a contradiction. Therefore, $lab \in NU(G_2) \Rightarrow f(lab) = lab' \in NU(G_1)$.

On the other hand, it is easy to see from the definition of lab' that f is injective. Therefore, $|NU(G_2)| \leq |NU(G_1)|$ and consequently $|U(G_1)| \leq |U(G_2)|$.

Finally, if $F \subseteq A(G_1)$ is a non feedback arc set of G_1, then $F' = \{(u, v) \in G_2 : (u, v) \in F \vee (v, u) \in F\}$ is a non feedback arc set of G_2. Therefore $|NFAS(G_1)| \leq |NFAS(G_2)|$ and consequently $|FAS(G_2)| \leq |FAS(G_1)|$. □

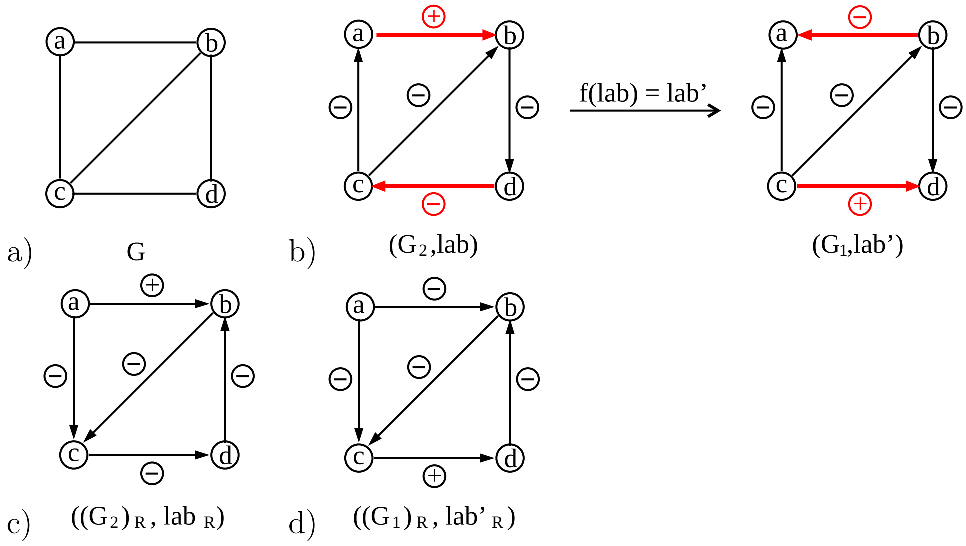


Figura 6.9. Proof idea for Theorem 6.5. *a)* An undirected graph G. *b)* Two orientations G_1 and G_2 of G and the injection f. Observe that $|U(G_1)| = 18 < 20 = |U(G_2)|$ and $|FAS(G_2)| = 27 < 32 = |FAS(G_1)|$. *c)* The labeled reoriented digraph $((G_2)_R, lab_R)$ associated to (G_2, lab). *d)* The labeled reoriented digraph $((G_1)_R, lab'_R)$ associated to (G_1, lab').

Proposition 6.5 $|U(G)| = |FAS(G)|$ *if and only if all circuits of G are cycles.*

Proof. $\Rightarrow$) By counter-reciprocal, let us suppose that G has a circuit C which is not a cycle. We will prove that $|U(G)| < |FAS(G)|$. We know from proof of Theorem 6.3 that given (G, lab) an update digraph, $\{a \in A(G) : lab(a) = \oplus\}$ is a feedback arc set of G induced by the function lab, and thus $|U(G)| \leq |FAS(G)|$. We will show that there is another feedback arc set $F' \in FAS(G)$, but F' is not induced by any element of $U(G)$. Indeed, there exists a label function lab' such that the circuit C is a forbidden circuit of (G, lab') with at least a positive arc and a negative arc in C. Let $F' = A(G) - \{a \in A(C) : lab'(a) = \ominus\}$. It is easy to check that $F' \in FAS(G)$ but F' is not induced by any element in $U(G)$.

$\Leftarrow$) Let us suppose that all circuits of G are cycles. We want to prove that $|FAS(G)| \leq |U(G)|$.

We define the following function $T_G : FAS(G) \longrightarrow U(G)$ such that for each feedback arc set $F \in FAS(G)$, $T_G(F)$ is the label function:

$$T_G(F)(a) = \begin{cases} \oplus, & \text{if } a \in F, \\ \ominus, & \text{if } a \notin F \end{cases}$$

Then, $T_G(F) \in U(G)$. Indeed, if there exists a forbidden circuit C in $(G, T_G(F))$, there is at least an arc $a \in A(C)$ such that $T_G(F)(a) = \oplus$. Since, if for all $a \in A(C)$, $T_G(F)(a) = \ominus$, then $A(C) \cap F = \emptyset$, which contradicts the fact $F \in FAS(G)$. Besides, there exists $a' \in A(C)$, $T_G(F)(a') = \ominus$. Hence, C is a circuit in G which is not a cycle, which is a contradiction. Therefore $T_G(F)$ is well-defined and obviously injective. □

6.6 Complete digraphs and Tournaments on update digraphs

In the previous sections we showed bounds on the number and size of equivalence classes of update schedules associated with a given update digraph G and relationships between the sets $U(G)$ and $FAS(G)$. In this section we will restrict to classical families such as acyclic digraphs, complete digraphs and tournaments, in order to know accurately the number and size of its schedule equivalence classes.

Also we will show how the number of negative arcs of an update digraph (G, lab) is related to feedback arc sets in these families.

PROPOSITION 6.6 *Let (G, lab) be an update complete digraph with at least one negative arc. Then, $\forall x \in V(G)$, $\exists y \in V(G)$ such that $lab(x, y) = \ominus$ or $lab(y, x) = \ominus$.*

PROOF. Suppose on the contrary that $\exists x \in V(G)$, $\forall y \in V(G)$, $lab(x, y) = \oplus$ and $lab(y, x) = \oplus$. By hypothesis, G is a complete digraph and it has a negative arc $(r, s) \in A(G)$, then in the clique x, r and s there is a forbidden cycle which contradicts the fact that G is an update digraph. □

PROPOSITION 6.7 *Let G be a complete digraph. Then, (G, lab) is a non update digraph if and only if there exists a forbidden cycle of length either two or three in (G_R, lab_R).*

PROOF. $\Leftarrow$) Is straightforward.

$\Rightarrow$) Let (G, lab) be a complete non update digraph. Then, there exists a forbidden cycle C_R with a negative arc and of smallest length in (G_R, lab_R). Let suppose that the length of C_R is strictly greater than three.

Let C be the forbidden circuit associated to C_R in G. If C has a path of the form (a, b), (b, c) $\oplus$-labeled, then necessarily (a, c) also is positive in G because otherwise, if (a, c) is negative in G, then there would exist a forbidden cycle C' of length three smaller than C_R in G_R. Analogously, if C has a path of the form (a, b), (b, c) $\ominus$-labeled, then necessarily (a, c) is negative (see Fig. 6.10, a) and b)).

Therefore, we can suppose that C does not have two consecutive arcs with the same label. But when $lab(b, a) = \ominus$ and $lab(b, c) = \oplus$, necessarily the arc (c, a) is negative in G (i.e (a, c) is negative in G_R), because on the contrary, there would exist a forbidden cycle C' of length three smaller than C_R in G_R (see Fig. 6.10, c) and d)).

In this way, it is always possible to reduce the length of the forbidden cycle C_R up to obtain a length of three. □

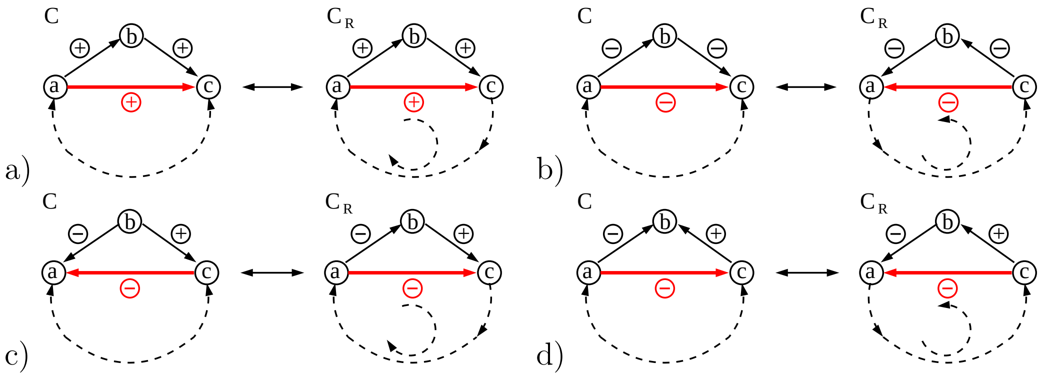


FIGURA 6.10. Proof idea of Proposition 6.7. *a)* When the arcs (a, b) and (b, c) are positive in C, then necessarily (a, c) also is positive in C. *b)* When the arcs (a, b) and (b, c) are negative in C, then necessarily (a, c) also is negative in C. *c)* If $lab(b, a) = \ominus$ and $lab(b, c) = \oplus$ in C, then necessarily $lab(c, a) = \ominus$. *d)* If $lab(a, b) = \ominus$ and $lab(c, b) = \oplus$ in C, then necessarily $lab(a, c) = \ominus$.

THEOREM 6.6 *Let G be an acyclic digraph with $|V(G)| = n$. Then,*

(a) $|U(G)| \leq n!$.

(b) $|U(G)| = n! \Leftrightarrow G$ *is a tournament.*

(c) If G is a tournament, then for each update schedule s of G, $|[s]_G| = 2^k$, for some $k \in \mathbb{N} \cup \{0\}$.

PROOF. *(a)* Let G be an acyclic digraph with $|V(G)| = n$ and let $lab \in U(G)$. Then, by Proposition 6.3, there is a sequential update schedule s_q such that $(G, lab) = (G, lab_{s_q})$. Therefore, since the number of different sequential update schedules is $n!$, we have that:

$$|U(G)| \leq n!$$

$(b) \Leftarrow$) Let G be a tournament with $|V(G)| = n$ and let s_1, s_2 be two different sequential update schedules, then by tournament definition there is $(i, j) \in A(G)$ such that $lab_{s_1}(i, j) \neq lab_{s_2}(i, j)$. Since every sequential update schedule s_q defines a different update digraph (G, lab_{s_q}) and the number of different sequential update schedules is $n!$, we have that:

$$|U(G)| \geq n!$$

and due to (a), we conclude that $|U(G)| = n!$.

$\Rightarrow$) Let G be an acyclic digraph with $|V(G)| = n$ and $|U(G)| = n!$. Suppose on the contrary that G is not a tournament, then necessarily there exist u, v in $V(G)$ without arcs between them (two arcs between them are not possible because G is acyclic) which implies that there are two different sequential update schedules with the same update digraph. This contradicts the fact that $|U(G)| = n!$. Therefore, G also is a tournament.

(c) Let T be an acyclic tournament. By induction on $m = |V(T)|$.

Basis Step, m=2. There are two posibilities: the equivalence class of the parallel update schedule s_p, where it is easy to see that $|[s_p]_T| = 2^1 = 2$, and of the sequential update schedule $s_q \notin [s_p]_T$, where clearly $|[s_q]_T| = 2^0 = 1$.

Induction Hypothesis. T acyclic tournament with $m \leq n-1$. Hence, for each update schedule s over $V(T)$, $|[s]_T| = 2^k$, for some $k \in \mathbb{N} \cup \{0\}$.

Let G be an acyclic tournament with $|V(G)| = n$. First, note that the proof of (b) implies the existence of a bijection $f : U(G) \rightarrow \{s \in S_n : s \text{ sequential update schedule on } V(G)\}$. Let (G, lab_s) be an update digraph with sequential update schedule $s = (i_1)(i_2)\cdots(i_j)(n)(i_{j+1})\cdots(i_{n-1})$, where $\{i_1, ..., i_{n-1}\} = \{1, \ldots, n-1\}$.

Let $G' = G - \{n\}$. Clearly, G' is an acyclic tournament with $|V(G')| = n-1$ and $s' = (i_1)\cdots(i_j)(i_{j+1})\cdots(i_{n-1})$ is the sequential update schedule such that $(G', lab_{s'})$ is the labeled subdigraph of (G, lab) induced by $\{i_1, \ldots, i_{n-1}\}$. Defining $S_{dif} = \{s^* \in [s']_{G'} : s^*(i_j) = s^*(i_{j+1}) - 1\}$ and $S_{eq} = \{s^* \in [s']_{G'} : s^*(i_j) = s^*(i_{j+1})\}$, we can see that if $S_{eq} \neq \emptyset$, then $|S_{dif}| = |S_{eq}| = 2^{k-1}$ for some $k \in \mathbb{N} \cup \{0\}$ (note that always $S_{dif} \neq \emptyset$).

There are the following 6 cases:

Case 1. $lab_s(i_j, n) = \ominus$ and $lab_s(i_{j+1}, n) = lab_s(i_{j+1}, i_j) = \oplus$ (see Figure 6.11 a)).
To obtain all the elements of $[s]_G$, for every $s^* \in S_{dif}$ it is enough to generate s_1 and s_2 in $[s]_G$ where $s_1(p) = s_2(p) = s^*(p)$, $\forall p \in V(G')$ such that $s^*(p) \leq s^*(i_j)$; $s_1(n) = s_2(n) = s^*(i_j) + 1$ and $s_2(q) = s_1(q) - 1 = s^*(q)$, $\forall q \in V(G')$ satisfying

$s^*(q) \geq s^*(i_{j+1})$. In this way, $|[s]_G| = 2 \cdot |S_{dif}| = 2^k$ for some $k \in \mathbb{N} \cup \{0\}$.

Case 2. $lab_s(n, i_j) = lab_s(i_{j+1}, n) = lab_s(i_{j+1}, i_j) = \oplus$ (see Figure 6.11 b)).
To obtain all the elements of $[s]_G$, for every $s_1^* \in S_{dif}$ we generate s_1, s_2 and s_3 in $[s]_G$ where $s_1(p) = s_2(p) = s_3(p) = s_1^*(p)$, $\forall p \in V(G')$ such that $s_1^*(p) \leq s_1^*(i_j)$; $s_1(n) = s_3(n) = s_2(n) + 1 = s_1^*(i_j) + 1$ and $s_2(q) = s_3(q) = s_1(q) - 1 = s_1^*(q)$, $\forall q \in V(G')$ satisfying $s_1^*(q) \geq s_1^*(i_{j+1})$. And for every $s_2^* \in S_{eq}$ we generate $s_4 \in [s]_G$ defined by $s_4(p) = s_2^*(p)$, $\forall p \in V(G')$, $p \neq n$ and $s_4(n) = s_2^*(i_j)$. Therefore, $|[s]_G| = 3 \cdot |S_{dif}| + |S_{eq}| = 3 \cdot 2^{k-1} + 2^{k-1} = 2^{k+1}$ for some $k \in \mathbb{N} \cup \{0\}$.

Case 3. $lab_s(i_j, n) = lab_s(i_j, i_{j+1}) = lab_s(n, i_{j+1}) = \ominus$ (see Figure 6.11 c)).
Here, for each $s_1^* \in S_{dif}$ we generate $s_1 \in [s]_G$ where $s_1(p) = s^*(p)$, $\forall p \in V(G')$ such that $s^*(p) \leq s^*(i_j)$; $s_1(n) = s^*(i_j) + 1$ and $s_1(q) = s^*(q) + 1$, $\forall q \in V(G')$ satisfying $s^*(q) \geq s^*(i_{j+1})$. Therefore, $|[s]_G| = |S_{dif}| = 2^{k-1}$ for some $k \in \mathbb{N} \cup \{0\}$.

Case 4. $lab_s(i_j, n) = lab_s(i_j, i_{j+1}) = \ominus$ and $lab_s(i_{j+1}, n) = \oplus$.

Case 5. $lab_s(n, i_j) = lab_s(i_{j+1}, i_j) = \oplus$ and $lab_s(n, i_{j+1}) = \ominus$.

Case 6. $lab_s(n, i_{j+1}) = lab_s(i_j, i_{j+1}) = \ominus$ and $lab_s(n, i_j) = \oplus$.

Cases 4, 5 and 6 are similar to case 1. $\square$

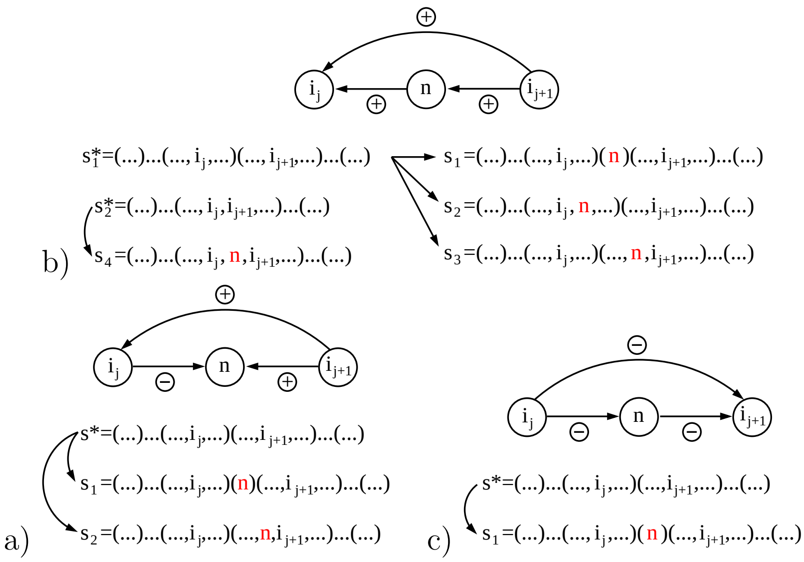


FIGURA 6.11. Proof idea of Theorem 6.6. a) Case 1. b) Case 2. c) Case 3.

Observe that the assertion (c) of Theorem 6.6 is not true for tournaments in general (see Fig. 6.12 as an counterexample).

PROPOSITION 6.8 *Let G be an acyclic tournament. Then, all arcs of an update digraph*

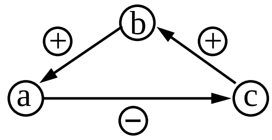


FIGURA 6.12. An update digraph (G, lab_s) where $|[s]_G| = |\{(a)(b)(c), (a)(b,c), (a,b)(c)\}| = 3$.

(G, lab) are negative if and only if there exists sequential update schedule s_q such that $(G, lab) = (G, lab_{s_q})$ and $|[s_q]_G| = 1$.

PROOF. Let G be an acyclic tournament. First, observe that every acyclic digraph (G, lab) with all its arcs negative is an update digraph.

$\Rightarrow$) Let (G, lab) with all its $|A(G)| = \binom{n}{2}$ arcs negative. Hence, by Proposition 6.3 there exists a sequential update schedule s_q such that $(G, lab) = (G, lab_{s_q})$. Now, suppose that there exists another update schedule $s \neq s_q$ such that $s \in [s_q]_G$. If s is a sequential update schedule, then since (G, lab) is a tournament and $s \neq s_q$, necessarily there are u, v in $V(G)$ such that $s(u) < s(v)$ and $s_q(u) > s_q(v)$. Thus, $(G, lab_s) \neq (G, lab_{s_q})$ which contradicts the assumption $s \in [s_q]_G$. If s is an update schedule but not sequential, then there is a block B_i, $1 \leq i \leq nb(s) < |V(G)|$ such that $|B_i| > 1$, i.e., there exist u, v in B_i such that the arc between them is positive in (G, lab_s), but the same arc is negative in (G, lab_{s_q}). Again a contradiction. Therefore, $|[s_q]_G| = 1$.

$\Leftarrow$) Let $s_q = (v_1)(v_2)\cdots(v_n)$ be a sequential update schedule such that $(G, lab) = (G, lab_{s_q})$, $|[s_q]_G| = 1$ and $V(G) = \{v_1, \ldots, v_n\}$. Since (G, lab) is an update tournament, for every pair of vertices v_i, v_j in $V(G)$, there is only one arc between them, either the negative arc (v_i, v_j) or the positive arc (v_j, v_i) with i, $j \in \{1, ..., n\}$. Observe that $\forall i = 1, \ldots, n-1$, the negative arc (v_i, v_{i+1}) is in $A(G)$, since otherwise there would be another update schedule $s \neq s_q$, $s = (v_1)(v_2)\cdots(v_{i-1})(v_i, v_{i+1})(v_{i+2})\cdots(v_n)$ such that $(G, lab) = (G, lab_{s_q}) = (G, lab_s)$, which contradicts that $|[s_q]_G| = 1$ (see Fig. 6.13 a)). Besides, for every $i \neq j \in \{1, \ldots, n\}$ such that $j > i+1$, the negative arc (v_i, v_j) is in $A(G)$ because if there exists the positive arc (v_j, v_i), by the above mentioned, we would have the existence of the negative arcs (v_i, v_{i+1}), (v_{i+1}, v_{i+2}),...,(v_{j-1}, v_j) which implies the existence of a cycle (see Fig. 6.13 b)). A contradiction because G is acyclic. Therefore, all arcs of (G, lab) must be negative. □

REMARK 6.3 *Let G be a complete digraph with $|V(G)| = n$ and F a minimal feedback arc set of G. Then, $|F| = \binom{n}{2}$ and talk of minimum feedback arc set is equivalent to talk of minimal feedback arc set. In fact, suppose that $|F| > \binom{n}{2}$, then necessarily there exist $u, v \in V(G)$ such that $\{(u,v),(v,u)\} \subseteq F$. Since F is minimal, there exists path $P_1 = u_1, ..., u_k$ with $u_1 = v$, $u_k = u$ which does not contain any arc of F (in particular*

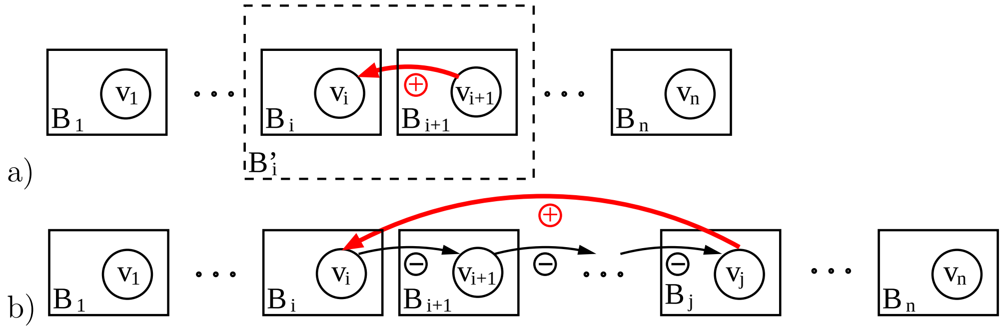


FIGURA 6.13. Proof idea of Proposition 6.8. a) Case $j = i + 1$. If there is a positive arc from the B_{i+1} to B_i, it is possible to construct a new update schedule $s \neq s_q$ with block $B'_i = B_i \cup B_{i+1}$. b) Case $j > i + 1$. There is necessarily a cycle (v_i, v_{i+1}), (v_{i+1}, v_{i+2}),...,(v_{j-1}, v_j), (v_j, v_i).

the arc (v, u)), for some $k \in [\![3, n]\!]$ and $u_i \in V(G)$, $\forall i \in [\![1, k]\!]$. Analogously, there exists path $P_2 = v_1, ..., v_j$ with $v_1 = u$, $v_j = v$ which does not contain any arc of F (in particular the arc (u, v)), for some $j \in [\![3, n]\!]$ and $v_i \in V(G)$, $\forall i \in [\![1, j]\!]$. Thus, joining the paths P_1 and P_2, we deduce the existence of a cycle which does not contain any arc of F, a contradiction, because F is a feedback arc set. Hence $|F| \leq \binom{n}{2}$. On the other hand, $|F| \geq \binom{n}{2}$ because G has $\binom{n}{2}$ cycles of length two.

LEMMA 6.1 *Let G be a complete digraph with $|V(G)| = n$. Then, F is a minimal feedback arc set of G if and only if $G' = G - F$ is an acyclic tournament.*

PROOF. $\Rightarrow$) Let F be a minimal feedback arc set of G, then $G' = G - F$ represents an acyclic digraph. Besides, G' is a tournament. In fact, suppose on the contrary that G' is not a tournament, i.e. there is a pair of vertices a, b in $V(G')$ such that (a, b), $(b, a) \notin A(G')$. That minds $\{(a, b), (b, a)\} \subseteq F$ and since F must have at least one arc for each pair of vertices of $V(G)$, we have that $|F| \geq \binom{n}{2} + 1$, which is a contradiction because by Remark 6.3, $|F| = \binom{n}{2}$.

$\Leftarrow$) Let G' be an acyclic tournament, then $F = A(G) - A(G')$ is a minimal feedback arc set of G, because $|F| = |A(G) - A(G')| = \binom{n}{2}$. □

THEOREM 6.7 *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 $lab_F(u, v) = \oplus \Leftrightarrow (u, v) \in F$.*

PROOF. $\Rightarrow$) Let G be a digraph with $|V(G)| = n$ and F a minimal feedback arc set of G. Then $G - F$ is an acyclic digraph and (G, lab_F) is an update digraph (see proof of Theorem 6.3 for more details).

Moreover, suppose on the contrary that there exists a label function lab with strictly more negative arcs than lab_F, where $lab_F(u,v) = \ominus \Rightarrow lab(u,v) = \ominus$, $(u,v) \in A(G)$ and such that (G, lab) is an update digraph. Then $F' = \{(u,v) \in A(G) : lab(u,v) = \oplus\}$ is a feedback arc set of G and verifies that $F' \subsetneq F$, which contradicts the minimality of F. Hence, (G, lab_F) is an update digraph with maximal number of negative arcs.

$\Leftarrow$) Let lab_F be a label function for which (G, lab_F) is an update digraph with maximal number of negative arcs, $F \subseteq A(G)$. We know that F is a feedback arc set of G because (G, lab_F) is an update digraph, which implies that $G - F$ is acyclic.
Now, suppose on the contrary that F is not minimal, i.e, there exists a minimal feedback arc set $F' \subsetneq F$ of G and consequently a label function $lab_{F'}$ with which $(G, lab_{F'})$ also is an update digraph but having the same negative arcs that (G, lab_F) and other more. This contradicts the maximality of (G, lab_F). Therefore, F is a minimal feedback arc set of G. □

Proposition 6.9 *An update digraph (G, lab) with maximum number of negative arcs has at least $\frac{|A(G)|}{2}$ negative arcs.*

Proof. Let (G, lab) be an update digraph and $na(G, lab) = |\{(u,v) \in A(G) : lab(u,v) = \ominus\}|$ the number of negative arcs of (G, lab).

If G is an acyclic digraph, then (G, lab) with all its arcs negative is an update digraph, i.e. $na(G, lab) = |A(G)|$.

If G has a cycle, then let $s_{q_1} = (v_1)(v_2)\cdots(v_n)$ and $s_{q_2} = (v_n)(v_{n-1})\cdots(v_1)$ two sequential update schedules with $v_i \in V(G)$, $\forall i \in [\![1,n]\!]$. Then, $\forall i,j \in [\![1,n]\!]$ such that $(v_i, v_j) \in A(G)$,

$$lab_{s_{q_1}}(v_i, v_j) = \ominus \Leftrightarrow lab_{s_{q_2}}(v_i, v_j) = \oplus$$

Thus, since (G, lab) has the maximum number of negative arcs, it must satisfy $na(G, lab) \geq max\{na(G, lab_{s_{q_1}}), na(G, lab_{s_{q_2}})\} \geq \frac{|A(G)|}{2}$. □

Corollary 6.9 *Let (G, lab) be a complete labeled digraph with $|V(G)| = n$. Then, (G, lab) is an update digraph with a maximal number of negative arcs if and only if the graph induced by the negative arcs is an acyclic tournament.*

Proof. By Theorem 6.7 we have that (G, lab) is an update digraph with a maximal number of negative arcs if and only if F is a minimal feedback arc set of G, where $F = \{(u,v) \in A(G) : lab(u,v) = \oplus\}$. By Lemma 6.1, F is a minimal feedback arc set of G if and only if $G' = G - F$ is an acyclic tournament, i.e., the graph induced by the negative arcs of (G, lab) is an acyclic tournament. □

Corollary 6.10 *Let (G, lab) be an update complete digraph. Then, (G, lab) has the maximal number of negative arcs if and only if the update schedule associated to (G, lab) is sequential.*

Proof. Let (G, lab) be an update complete digraph.

$\Rightarrow$) By Corollary 6.9, the digraph induced by the negative arcs is an acyclic tournament T which by Proposition 6.8 has a sequential update schedule s_q such that $(T, lab|_T) = (T, lab_{s_q}|_T)$ and $|[s_q]_T| = 1$. Hence, it is easy to see that $(G, lab) = (G, lab_{s_q})$.

$\Leftarrow$) Let $s_q = (v_1)(v_2)\cdots(v_n)$ a sequential update schedule such that $(G, lab) = (G, lab_{s_q})$. Hence, all the negative arcs of (G, lab) are of the form (v_i, v_j) with $j > i$ and i, j in $\{1, ..., n\}$, i.e. the digraph induced by the negative arcs of (G, lab) is an acyclic tournament which implies again by Corollary 6.9 that (G, lab) has the maximal number of negative arcs. □

Corollary 6.11 *Let G be a complete digraph with $|V(G)| = n$. Then, $|MFAS(G)| = |\{lab : (G, lab)$ is an update digraph with the maximal number of negative arcs$\}| = n!$.*

Proof. It is easy to see that the number of acyclic tournaments of n vertices is $n!$. Then, the Corollary follows from Lemma 6.1 and Theorem 6.7. □

Proposition 6.10 *Let (G, lab) be an update digraph with a maximal number of negative arcs. Then, there is no positive cycle.*

Proof. Let (G, lab) be an update digraph with maximal number of negative arcs, i.e, there is no positive arc that can be changed by a negative arc holding the property of update digraph. Let s be an update schedule such that $(G, lab) = (G, lab_s)$ and suppose on the contrary that there is a positive cycle C, then there exists $(i, j) \in A(C)$ such that $s(i) = s(j)$. Defining s' by:

$$s'(k) = s(k),\ \forall k,\ s(k) < s(i).$$

$$s'(i) = s(i)$$

$$s'(k) = s(k) + 1,\ \forall k \neq i,\ s(k) \geq s(i).$$

we have that $lab_s(u, v) = \ominus \Rightarrow lab_{s'}(u, v) = \ominus$, but $lab_s(i, j) = \oplus \neq lab_{s'}(i, j)$ which contradicts the maximality of (G, lab). □

Remark 6.4 *In general, the necessary condition of Proposition 6.10 is not sufficient. As a counterexample, we can consider the update digraph H_2 of Fig. 7.3 and observe that H_2 satisfies the hypotheses of the Proposition and in addition, it does not have positive cycles. Nevertheless it is not any of three update digraphs with maximal number of negative arcs: H_5, H_6 or H_7. Also H_2 is a counterexample for the families of update strongly connected digraphs and update tournaments with cycles. Even in acyclic digraphs (such as acyclic tournaments), the sufficient condition does not hold. We can consider as a counterexample a simple labeled digraph (G, lab) composed by only one positive arc (a, b) which is obviously an update digraph, without positive cycles, but not maximal. However, the following Proposition shows that in update complete digraphs, the necessary condition of Proposition 6.10 is also sufficient one.*

Proposition 6.11 *Let (G, lab) be an update complete digraph with $|V(G)| = n$. Then, (G, lab) has a maximal number of negative arcs if and only if (G, lab) has no positive cycle.*

Proof. $\Rightarrow$) Direct from Proposition 6.10.

$\Leftarrow$) Let (G, lab) be an update complete digraph without positive cycles with $|V(G)| = n$, i.e. all its cycles of length two have only one negative arc. Since a complete digraph has $\binom{n}{2}$ cycles of length two, (G, lab) has at least $\binom{n}{2}$ negative arcs, but this is the maximum number of negative arcs that an update complete digraph can have. $\square$

CHAPTER 7

Enumeration of update digraphs and equivalence classes

In the previous chapter, we have studied bounds for the cardinality of update digraphs associated with a given digraph as well as bounds for the size of its associated equivalence classes.

In this chapter, we will see how through local transformations on update digraphs we can obtain other update digraphs, all in a polynomial time, being this one, one of the principal motivation to study the above mentioned. Also we will study properties associated with these transformations and the associated multidigraphs generated from them, where speaking in simple terms, the vertices of these multidigraphs are update digraphs and its arcs are defined depending on the existence or not of some transformation that allows us to transform a vertex to another.

We finish this chapter by giving an exact algorithm for enumerating all the update digraphs associated to a given digraph as well as an exact algorithm for determining all the update schedules associated to a given update digraph. Both use a special property that allows us to find all the elements of interest without repetition.

[4]work in preparation: J. Aracena, J. Demongeot, M. Montalva, "Local transformations and enumeration of update digraphs".

7.1 Some transformations on update digraphs

In the proof of Theorem 6.3, we see that if $F \subseteq A(G)$ is a minimal feedback arc set of a digraph G, then (G, lab_F) is an update digraph where $\forall e \in A(G)$, $lab_F(e) = \oplus \Leftrightarrow e \in F$. For this reason, study the minimal feedback arc sets of a digraph can help us to understand better the update digraphs. In this context, Schwikowski and Speckenmeyer (2002) present an algorithm that exploits a simple relation between minimal feedback arc sets that allows for generating all minimal feedback arc sets of a directed graph G by local modification. They further show that the underlying technique can be tailored to generate all minimal solutions for the undirected case and the directed feedback arc set problem, both with a polynomial delay of $O(|V||E|(|V|+|E|))$, proving finally that computing the number of minimal feedback arc sets is ♯P-hard. This is used as motivation for the results that follow. We begin with a simple proposition that allows us later, to define local transformations on update digraphs.

PROPOSITION 7.1 *Let (G, lab) be an update digraph, then $\exists x \in V(G)$, $[\forall w \in N^-(x),\ lab(w,x) = \oplus] \vee [\forall y \in N^+(x),\ lab(x,y) = \oplus]$.*

PROOF. Suppose on the contrary that $\forall x \in V(G)$, $\exists w \in N^-(x)$, $\exists y \in N^+(x)$, $lab(w,x) = lab(x,y) = \ominus$, in this way, there exists a succession of vertices $\{v_i\}_{i\in\mathbb{N}}$ of $V(G)$ such that $lab(v_i, v_{i+1}) = \ominus$. Then there would be a cycle with only negative arcs, i.e. a forbidden circuit in (G, lab), which is a contradiction. □

Proposition 7.1 was the motivation for the next definition

DEFINITION 7.1 *Let G be a digraph and $U_G = \{G\} \times U(G)$. For all $x \in V(G)$, we define the functions $T^i_x : U_G \to U_G$ such that $(G, lab) \to T^i_x(G, lab) = (G, lab^i_x)$, $i = 1, 2$ where $lab^i_x : A(G) \to \{\ominus, \oplus\}$ is defined as follows (see Figure 7.1):*

$$lab^1_x(u,v) = \begin{cases} \ominus, & \text{if} \quad v = x, \\ \oplus, & \text{if} \quad u = x, \\ lab(u,v), & \text{in other case} \end{cases}$$

$$lab^2_x(u,v) = \begin{cases} \oplus, & \text{if} \quad v = x, \\ \ominus, & \text{if} \quad u = x, \\ lab(u,v), & \text{in other case} \end{cases}$$

Observe that T^1_x and T^2_x are polynomial time computable functions for all $x \in V(G)$.

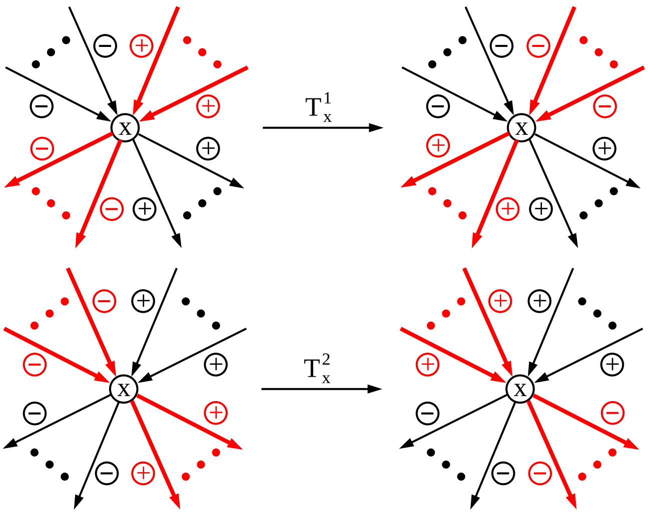


FIGURA 7.1. Local effect of T_x^1 and T_x^2 over a vertex x of a given labeled digraph yields the properties P_1 and P_2 respectively.

PROPOSITION 7.2 *Let (G, lab) be an update digraph. Then $T_x^1(G, lab)$ and $T_x^2(G, lab)$ are update digraphs for each $x \in V(G)$.*

PROOF. Let (G, lab) be an update digraph and $x \in V(G)$. Then, there is no forbidden circuit in (G, lab). If $T_x^i(G, lab)$ is not an update digraph, then there exists necessarily a forbidden circuit containing x. But this is impossible due the definition of $T_x^i(G, lab)$. □

DEFINITION 7.2 *Let G be a digraph without loops and $I = \{1, 2\}$. We denote H_G the multidigraph associated to G, defined by:*

$V(H_G) = \{(G, lab) : (G, lab)$ *is an update digraph*$\}$ *and,*

$A(H_G) = \{(H, H') \in V(H_G) \times V(H_G) : H = (G, lab) \wedge H' = T_x^1(G, lab),\ x \in V(G)\}$.

Let be (G, lab) an update digraph. A vertex $H \in V(H_G)$ such that $N_{H_G}^-(H) = \emptyset$ is said to be a root vertex of H_G (see Fig. 7.3).
$x \in V(G)$ has the property P_1 (resp. P_2) if $lab(v, x) = \ominus$ (resp. $lab(v, x) = \oplus$), $\forall v \in N_G^-(x)$ and $lab(x, y) = \oplus$ (resp. $lab(x, y) = \ominus$), $\forall y \in N_G^+(x)$ (see Fig. 7.1).

Observe that:

1.- If (G, lab) is an update digraph with loops, then each loop is necessarily positive. Thus, we can define H_G for update digraphs with loops considering that T_x^1, $x \in V(G)$, is applied on (G, lab) without its loops and later put positive arcs on its loops.

2.- If G is a complete digraph with $n = |V(G)|$, then $|V(H_G)| = |S_n| = T_n = \sum_{k=0}^{n-1} \binom{n}{k} T_k$, i.e. for each (G, lab) update, $\exists!\ s \in S_n$ such that $(G, lab) = (G, lab_s)$.

3.- $T_x^i \circ T_x^j = T_x^i$, $\forall (i,j) \in I \times I$.

4.- Similary we can define H_G in terms of T_x^2, $x \in V(G)$ without changing the analysis that follows.

COROLLARY 7.1 *Let (G, lab) be a labeled digraph and $F = \{v_1, ..., v_k\}$, $k \in \{1, ..., |V(G)| - 1\}$ a feedback vertex set of the undirected multigraph associated to G, i.e. G without its orientations. Then $(G, lab_F) = T_{v_k}^{i_k} \circ \cdots \circ T_{v_2}^{i_2} \circ T_{v_1}^{i_1}(G, lab)$ is an update digraph, where $i_j \in \{1, 2\}$, $\forall j \in \{1, ..., k\}$.*

PROOF. Suppose on the contrary that there exists a forbidden circuit C in (G, lab_F). Since F is a feedback vertex set of the undirected multigraph associated to G, there exists $p = \max_{i \in \{1, ...k\}} \{i : v_i \in V(C)\}$, then by $T_{v_p}^{i_p}$ definition, there is no cycle in (G_R, lab_R) that contains v_p, i.e. C is not a forbidden circuit, which is a contradiction. □

REMARK 7.1 *In particular, Corollary 7.1 says that applying T_x^i, $i \in I$ to all vertices x of a given labeled digraph (G, lab) gives us a new labeled digraph (G, lab') which is update.*

PROPOSITION 7.3 *Let $H \equiv (G, lab)$ be an update digraph. The following are equivalent:*

(a) H is a non root vertex of H_G.

(b) There exists $H^* \in V(H_G)$, $H^* \neq H$ such that $(H^*, H) \in A(H_G)$.

(c) There exists $x \in V(G)$ having the property P_1.

(d) $(H, H) \in A(H_G)$.

PROOF. $(a) \Rightarrow (b)$. H non root vertex of H_G implies that $N_{H_G}^-(H) \neq \emptyset$, i.e. there exists $H_1 \in N_{H_G}^-(H)$ such that $(H_1, H) \in A(H_G)$. If $H_1 = H$, then $(H, H) \in A(H_G)$

and $T_x^1(H) = H$ for some $x \in V(G)$. Let $H^* = T_x^2(H)$, then x has the property P_1 in H and the property P_2 in H^* which implies that $H^* \neq H$. Finally, $T_x^1(H^*) = T_x^1(T_x^2(H)) = T_x^1(H) = H$, i.e. $(H^*, H) \in A(H_G)$.

$(b) \Rightarrow (c)$. $(H^*, H) \in A(H_G) \Rightarrow T_x^1(H^*) = H$, for some $x \in V(G)$, i.e. $x \in V(G)$ has the property P_1.

$(c) \Rightarrow (d)$. $x \in V(G)$ with property P_1 implies that $H = T_x^1(H)$ (by definition of T_x^1), i.e. $(H, H) \in A(H_G)$.

$(d) \Rightarrow (a)$. $(H, H) \in A(H_G) \Rightarrow N^-_{H_G}(H) \neq \emptyset$, i.e. H is a non root vertex of H_G. □

PROPOSITION 7.4 *Let (G, lab) be an update digraph with $|V(G)| = n$. Then (G, lab) is a root vertex of H_G if and only if for all update schedule s such that $(G, lab) = (G, lab_s)$ and with maximum number of blocks, the partition $P_s = \{B_1, ..., B_k\}$ associated to s for some $k \leq n$, satisfies $|B_k| > 1$.*

PROOF. $\Rightarrow$) Let (G, lab) be a root vertex of H_G, then (G, lab) is an update digraph and consequently, there exists update schedule s such that $(G, lab) = (G, lab_s)$. We assume w.l.o.g. that s has maximum number of blocks and $P_s = \{B_1, ..., B_k\}$. Suppose on the contrary that $B_k = \{x\}$, i.e. $|B_k| = 1$ for some $x \in V(G)$. By definition of B_k, $\forall y \in N^-(x)$, $lab_s(y, x) = \ominus$ and $\forall z \in N^+(x)$, $lab_s(x, z) = \oplus$. This implies that x has the property P_1 in (G, lab_s). Then, by Proposition 7.3, $(G, lab_s) = (G, lab)$ is a non root vertex of H_G. Contradiction. Therefore, for all update schedule s such that $(G, lab) = (G, lab_s)$, with maximum number of blocks and associated partition $P_s = \{B_1, ..., B_k\}$, $|B_k| > 1$.

$\Leftarrow$) Let (G, lab) be a labeled digraph and suppose that for all update schedule s with maximum number of blocks, the partition $P_s = \{B_1, ..., B_k\}$ associated to s for some $k \leq n$, is such that $(G, lab) = (G, lab_s)$ and $|B_k| > 1$.

If $k = 1$, then (G, lab) has all its arcs positive and $\forall x \in V(G)$, x belongs to a positive cycle (in other case there would exist another update schedule $s' \neq s$ such that $(G, lab) = (G, lab_{s'})$ and such that $nb(s') > nb(s)$, contradiction). Then, $\forall x \in V(G)$, x has not the property P_1. Thus, by Proposition 7.3, (G, lab) is a root vertex of H_G.

If $k > 1$, then suppose on the contrary that another update digraph (G, lab') is such that $T_x^1(G, lab') = (G, lab)$ (eventually $(G, lab') = (G, lab)$), for some $x \in V(G)$. By definition of T_x^1, x does not belong to any positive cycle of (G, lab). Since $\forall y \in B_k$, y belongs to a positive cycle of (G, lab) (reasoning similarly to case $k = 1$), the above mentioned necessarily implies that $x \notin B_k$ and consequently x has the property P_1 in (G, lab). But this would implies that there exists another update schedule $s' \neq s$ with associated partition $P_{s'} = \{Z_1, ..., Z_r\}$, for some $r \in [\![k, n]\!]$ and such that $Z_r = \{x\}$, i.e. $|Z_r| = 1$ which is a contradiction. Therefore, every update digraph (G, lab') is such that $T_x^1(G, lab') \neq (G, lab)$, $\forall x \in V(G)$, i.e. (G, lab) is a root vertex of H_G. □

COROLLARY 7.2 *Let G be a digraph. If $H \in H_G$ is a root vertex, then H is not a reduced digraph.*

PROOF. Direct from Propositions 6.3 and 7.4. □

REMARK 7.2 *The necessary condition of Corollary 7.2 is not sufficient (see Fig.7.2 as an counterexample).*

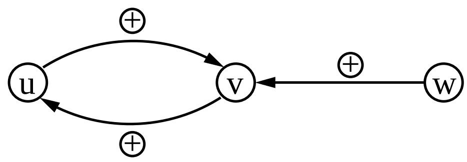


FIGURA 7.2. An update digraph (G, lab) which is not reduced nor a root vertex of H_G because $T_w^1(G, lab) = (G, lab)$.

PROPOSITION 7.5 *Let G be a digraph. Then H_G is connected.*

PROOF. In fact, for each $(G, lab) \in H_G$ there exists an update schedule s with a maximum number of blocks such that $(G, lab) = (G, lab_s)$ and an associated partition $P_s = \{B_1, ..., B_k\}$, $k \leq |V(G)|$. Hence, there exists an update schedule s' with associated partition $P_{s'} = \{B'_1, ..., B'_k, B'_{k+1}\}$ and maximum number of blocks such that $T_x^1(G, lab_s) = (G, lab_{s'})$, for some $x \in B_i$ such that $|B_i| > 1$. This process can be repeated up to obtaining a partition $P_{s''}$ induced by a sequential update schedule s''. On the other hand, for $H \in H_G$ with parallel update schedule associated, there exists a path that finish in each update digraph of H_G that has a sequential update schedule in its equivalence class. □

REMARK 7.3 *In general, given a digraph G, then H_G is not a strongly connected digraph (see Fig. 7.3). However, as we will see, this property is achieved when H_G is restricted to the reduced update digraphs.*

THEOREM 7.1 *Let G be a digraph. Then, H_G restricted to the reduced update digraphs is strongly connected.*

PROOF. Let G be a digraph and $H_1 = (G, lab_1)$ a reduced update digraph. Then by Proposition 6.3, H_1 has no positive cycle and there exists a sequential update

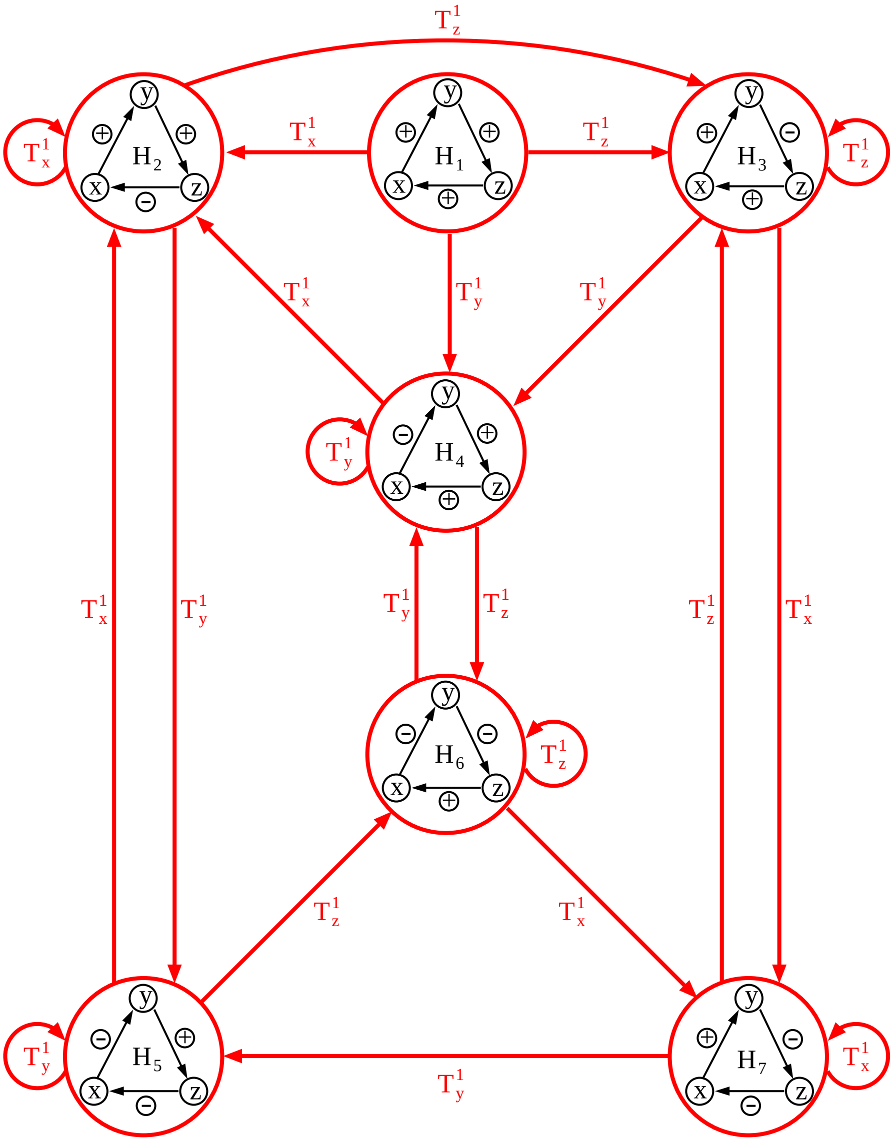


FIGURA 7.3. The digraph H_G associated to the digraph G which is a cycle. For G, there are seven different update digraphs named H_1,...,H_7 respectively which are the vertices of H_G and only H_1 is a root vertex. The arcs of H_G are obtained from direct aplication of T_k^1 over H_j, where $k \in \{x, y, z\}$ and $j \in \{1, ..., 7\}$.

schedule s_1 such that $H_1 = (G, lab_{s_1})$. Similarly, if $H_2 = (G, lab_2)$ is another reduced update digraph, then there will exist a sequential update schedule $s_2 \neq s_1$ such that $H_2 = (G, lab_{s_2})$.

On the other hand, let $s = (v_1)\cdots(v_n)$ be a sequential update schedule with $v_i \in V(G)$, $\forall i \in [\![1, n]\!]$. Observe from the definition of T^1_x, $x \in V(G)$ that, $\forall v_i \in V(G)$, $i \in [\![1, n]\!]$, $T^1_{v_i}(G, lab_s)$ has an associated sequential update schedule,

$$s^*_i = \begin{cases} (u_1)\cdots(u_{k-1})(u_{k+1})\cdots(u_{n-1})(v_i) & \text{if } u_k = v_i \wedge k \in [\![1, n-1]\!]. \\ s_1 & \text{if } u_n = v_i. \end{cases}$$

such that $T^1_{v_i}(G, lab_s) = (G, lab_{s^*_i})$ and again by Proposition 6.3, $(G, lab_{s^*_i})$ is reduced. Thus, if $s = (v_1)\cdots(v_n)$ with $v_i \in V(G)$, $\forall i \in [\![1, n]\!]$, then there exists a succession of vertices of H_G such that $H_1 = (G, lab_{s_1})$, $H^*_2 \equiv T^1_{v_1}(H_1)$, $H^*_3 \equiv T^1_{v_2}(H^*_2)$,..., and $H^*_{n+1} \equiv T^1_{v_n}(H^*_n) = H_2$, i.e. there exists a path in H_G from H_1 to H_2. Reasoning analogously to the above, there exists a path in H_G from H_2 to H_1. Therefore, H_G restricted to the reduced update digraphs is strongly connected. □

REMARK 7.4 *From the above proof we can deduce that H_G restricted to reduced update digraphs is equivalent to H_G restricted to update digraphs with a sequential update schedule in its equivalence classes. Therefore, we can enumerate with a polynomial-delay the equivalence classes containing a sequential update schedule. In particular, this could be useful for enumerating, for example, the different sequential dynamical systems with fixed local functions on their vertices (Mortveit and Reidys, 2001).*

REMARK 7.5 *If G is acyclic, then H_G is strongly connected.*

COROLLARY 7.3 *Let G be a complete digraph and let H'_G be the subgraph of H_G induced by $V(H'_G) = \{H \in H_G :$ H has a maximum number of negative arcs$\}$. Then H'_G is strongly connected.*

PROOF. Let $H \in H'_G$. By corollary 6.9, the graph induced by the negative arcs of H is an acyclic tournament which has associated an unique sequential update schedule s_q. Hence, by definition of $T^i_x(H)$ for some $x \in V(G)$, $i \in I$, we obtain a new sequential update schedule s'_q that differs from s_q only in the update of vertex x ($s'_q(x) = n$ in $T^1_x(H)$ and $s'q(x) = 1$ in $T^2_x(H)$). Consequently, s'_q has an associated new update complete digraph H' with a maximum number of negative arcs. In this way, $\forall H_1, H_2 \in H'_G$, there is a path from H_1 to H_2, i.e. H'_G is strongly connected. □

PROPOSITION 7.6 *Let G be a complete digraph with $n = |V(G)|$. Then, the number of root vertices in H_G is:*

$$I_n = \sum_{k=0}^{n-2} \binom{n}{k} (T_k - kT_{k-1}), \quad n \geq 2 \text{ and } T_{-1} \equiv 0.$$

PROOF. First, note that by Remark 6.2, $|V(H_G)| = |U(G)| = T_n$ because G is a complete digraph.

Let s be an update schedule for G, $A = \{i \in V(G) : s(i) = 1\}$, $B = V(G) - A$ and suppose w.l.g that $|A| = n - k$ and $|B| = k$, $k \in \{0, \dots, n-1\}$.

If $k = 0$, then (G, lab_s) is a complete digraph with all its arcs positive, i.e. there is no vertex with property P_1 or P_2 and consequently by Proposition 7.3, (G, lab_s) is a root vertex of H_G.

If $k \in \{1, \dots, n-2\}$, then $s(u) < s(v)$, for all $(u, v) \in A \times B$. Hence, again by Proposition 7.3, (G, lab_s) is a non root vertex of H_G if and only if there exists $w \in V(G)$ such that $s(v) < s(w)$, $\forall v \in N_G^-(w) \cap B$, because in this way, w will have the property P_1. There are $\binom{n}{k} k T_{k-1}$ different ways of obtaining the above mentioned for each $k \in \{1, \dots, n-2\}$ (see a) of Figure 7.4).

If $k = n - 1$, we have that $|A| = 1$, then $[v \in A \Rightarrow s(v) < s(w), \forall w \in N_G^+(v) = V(G) - \{v\}]$, i.e. the vertex $v \in A$ has the property P_2 and again by Proposition 7.3, (G, lab_s) is a non root vertex of H_G. In this case, there are $\binom{n}{n-1} T_{n-1}$ different combinations to obtain an update digraph (see b) of Figure 7.4).

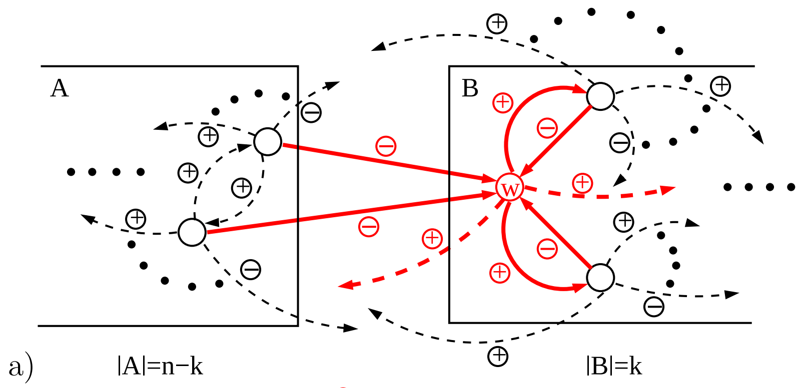
Therefore:

$$
\begin{aligned}
I_n &= |V(H_G)| - |\{v \in V(H_G) : v \text{ is a non root vertex of } H_G\}| \\
&= T_n - \left\{ \sum_{k=1}^{n-2} \binom{n}{k} k T_{k-1} + \binom{n}{n-1} T_{n-1} \right\} \\
&= \left\{ T_n - \binom{n}{n-1} T_{n-1} \right\} - \sum_{k=1}^{n-2} \binom{n}{k} k T_{k-1} \\
&= \sum_{k=0}^{n-2} \binom{n}{k} T_k - \sum_{k=1}^{n-2} \binom{n}{k} k T_{k-1} \\
&= \sum_{k=0}^{n-2} \binom{n}{k} (T_k - k T_{k-1}), \text{ with } T_{-1} \equiv 0.
\end{aligned}
$$

□

PROPOSITION 7.7 *Let be G a complete digraph. Then, for all $H \in H_G$ there exists a path from a root vertex of H_G to H of length at most 1.*

PROOF. Let $H \in H_G$. If H is a root vertex, then it is straightforward. If H is not a root vertex, let be $P = \{B_1, ..., B_k\}$ a partition of $V(G)$ associated with the only update schedule s of H. Then by Proposition 7.4, $|B_k| = 1$, i.e. $B_k = \{x\}$ for some



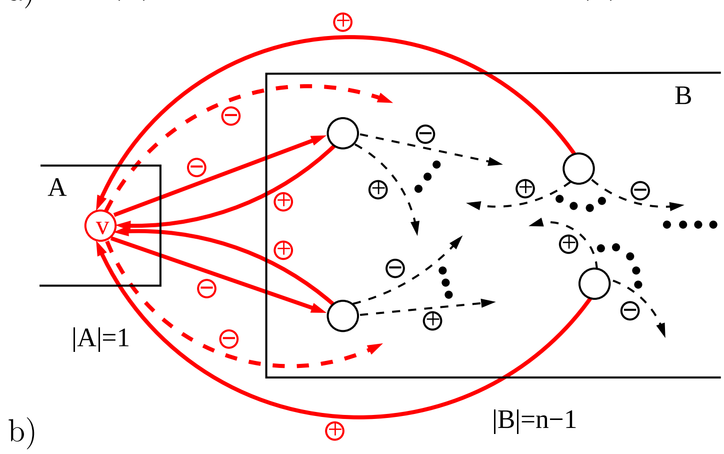


FIGURA 7.4. Proof idea from Proposition 7.6. a) Case $k \in \{1, \ldots, n-2\}$. There are $\binom{n}{k}$ different ways to obtain A and B, for each one of them there are $\binom{k}{1} = k$ different ways to obtain w and for each one of them there are T_{k-1} different update digraphs. This give us a total number of $\sum_{k=1}^{n-2} \binom{n}{k} k T_{k-1}$ update digraphs non root vertices of H_G.
b) Case $k = n-1$. Since A has only the vertex v, there are $\binom{n}{1} = \binom{n}{n-1}$ different ways to choose such vertex and for each one of them there are T_{n-1} different update digraphs. This gives us a total number of $\binom{n}{n-1} T_{n-1}$ update digraphs non root vertices of H_G.

$x \in V(G)$ (see Fig. 7.5), then the partition $P' = \{B'_1, ..., B'_{k-1}\}$ with $B'_i = B_i$ for all $i = 1, .., k-2$ and $B'_{k-1} = B_{k-1} \cup B_k$ associated to an update schedule s', verifies that $H' = (G, lab_{s'}) \neq H$ (because G is complete), H' is a root vertex of H_G (because $|B'_{k-1}| > 1$ and by Remark 6.2 and Proposition 7.4) and $T_x^1(H') = H$. $\square$

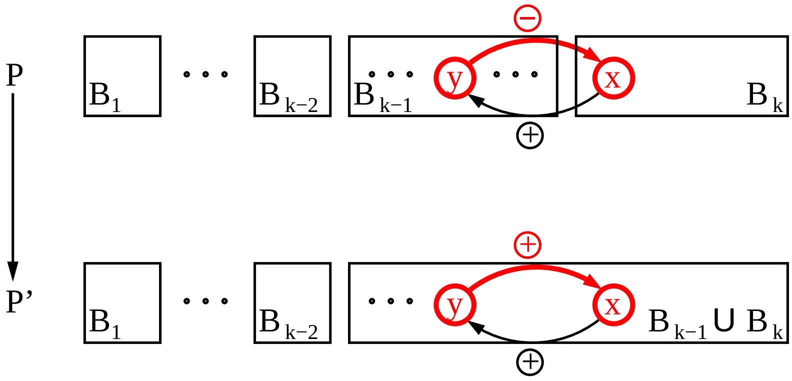


FIGURA 7.5. Proof idea from Proposition 7.7. All negative arcs of the form (y, x) in $H = (G, lab_s)$ with the associated partition P become positive in $H' = (G, lab_{s'})$ with the associated partition P'.

REMARK 7.6 *In general, Proposition 7.7 is not true for digraphs. A counterexample is the subdigraph G' of G in Fig. 7.2 formed only by the arc (w, v). In this case, $H_{G'}$ would have only two vertices, being none of them a root vertex of him.*

7.2 EUD algorithm (enumerating update digraph)

The following is an algorithm for determinating every update digraph from a given digraph.

Algorithm 2 EUD

Require: $G = (V, E)$ a digraph.
  $UD \leftarrow$ DigraphUD($\emptyset$,$\emptyset$,V);

DEFINITION 7.3 *Let $G = (V, A)$ be a digraph and $C, D \subseteq V$. We define the subgraph of G asociated to C and D by $G_{(C,D)} = G[C \cup D]$. Also we define $lab_{(C,D)} : A(G_{(C,D)}) \longrightarrow \{\oplus, \ominus\}$ by:*

$$lab_{(C,D)}(u, v) = \begin{cases} \ominus, & u \in C \ \wedge \ v \in D, \\ \oplus, & otherwise \end{cases}$$

Algorithm 3 `DigraphUD`

Require: U, A, B subsets of vertices of a digraph G.
  $UD \leftarrow \emptyset$;
  if $U = A = \emptyset$ **then**
    $UD = UD \cup B$;
    for $A_0 \subset B$ such that $A_0 \neq \emptyset$ with decreasing size **do**
      $B_0 = B - A_0$;
      $UD = UD \cup DigraphUD(\emptyset, A_0, B_0)$;
    end for
  end if

  if $mover(U, A) = 0$ **then**
    if $mover(A, B) = 0$ **then**
      $UD = UD \cup (U(*)A)(*)B$;
    end if
    if $|B| > 1$ **then**
      for $A_1 \subset B$ such that $A_1 \neq \emptyset$ with decreasing size **do**
        $B_1 = B - A_1$;
        $UD = UD \cup DigraphUD(U(*)A, A_1, B_1)$;
      end for
    end if
    return(UD);
  end if

Algorithm 4 `mover`

Require: U, A subsets of vertices of a digraph G.
  if $U = \emptyset$ **then**
    return(0);
  else
    if $\exists H \subseteq A$ such that $(G_{(U,A)}, lab_{(U,A)}) = (G_{(U \cup H, A-H)}, lab_{(U \cup H, A-H)})$ **then**
      return(1);
    else
      return(0);
    end if
  end if

Proposition 7.8 *Let G be a digraph with $|V(G)| = n$ and let $P = \{Z_1, ..., Z_k\}$, $k \in [\![1, n]\!]$, be a partition obtained of EUD algorithm such that (G, lab_P) is its associated update digraph. Then, (G, lab_P) is repeated if and only if it is possible to move vertices from Z_{k-1} to Z_{k-2} or from Z_k to Z_{k-1}.*

Proof. $\Rightarrow$) Let $P = \{Z_1, ..., Z_k\}$ be a partition obtained from EUD algorithm, $k \in [\![1, n]\!]$ and let (G, lab_P) be its associated update digraph such that it is repeated, i.e. (G, lab_P) has already been obtained from another partition $P' = \{W_1, ..., W_t\}$, $t \in [\![1, n]\!]$. Suppose on the contrary that it is not possible to move vertices from Z_{k-1} to Z_{k-2} and from Z_k to Z_{k-1}. This implies that it is not possible to move vertices from Z_{k-2} to Z_{k-3}, because in contrary case, the algorithm EUD would have stopped before, without P as output, which contradict what has been said in the beginning. Continuing recursively with this argument, we deduce in general that it is not possible to move vertices from Z_i to Z_{i-1}, $\forall i \in [\![2, k]\!]$.

Let $w \in W_i$ be the first element such that $w \notin Z_i$, by comparing the elements of W_i with the elements of Z_i, for $i \in [\![1, min\{k, t\}]\!]$. This implies that $W_j = Z_j$, $\forall j < i$, because in contrary case, that would imply that P was obtained from EUD algorithm before P', which would be a contradiction. Hence, in P, necessarily $w \in Z_p$, for some $p > i$. Since it is not possible to move vertices from Z_p to Z_{p-1}, then necessarily $\exists y \in Z_{p-1}$ for which the arc (y, v) is negative for some $v \in Z_p$ such that there exists a path from w to v (eventually $v = w$) because in contrary case, we might move the set $\{w\} \cup \{v \in Z_p :$ there is a path from w to $v\} \subseteq Z_p$ to Z_{p-1} which would be a contradiction. But if the arc (y, v) is negative for some $v \in Z_p$ such that there exists a path from w to v, necessarily $v \in W_i$ and this would imply that necessarily $y \in W_r$, for some $r < i$, which would contradict that $W_r = Z_r$.

$\Leftarrow$) Is straightforward. □

7.3 EC algorithm (equivalence class)

The following is an algorithm for determinating all the elements of an equivalence class $[s]_G$ associated to a reduced update digraph (G, lab_s).

EC algorithm works recursively, through the subroutine PART, checking all possible partitions of $V(G)$ that generate the same update digraph (G, lab_s).

Algorithm 5 `EC`

Require: A digraph G and a update schedule s such that (G, lab_s) is a reduced update digraph.
  if $(G_R, (lab_s)_R)$ is a negative linear digraph **then**
    $S \leftarrow s$.
  else
    $S \leftarrow \mathrm{PART}(V)$
  end if

Algorithm 6 `PART`

Require: $V' \subseteq V$ subset of vertices of a digraph $G = (V, E)$ and an update schedule s such that (G, lab_s) is a reduced update digraph.
  $S \leftarrow \emptyset$;
  if $V' = \emptyset$ **then**
    return($\emptyset$);
  else
    $A \leftarrow \{v \in V' : \exists u \in V'$ such that $(u, v) \in E$ and $s(u) < s(v)\}$;
    for $W \subseteq V' - A$ such that $W \neq \emptyset$ **do**
      if *compatibility-test*$(W, V' - W) = 1$ **then**
        $S = S \cup W(*)\mathrm{PART}(V' - W)$;
      end if
    end for
  end if
  return(S)

7.4 Experimental analysis

In this section, we will apply some theoretical results and algorithms developed on the equivalence classes of the different deterministic update schedules associated to a given update digraph. Specifically, we will analyze the possible dynamics of a real genetic regulation network of the floral morphogenesis in the plant *Arabidopsis thaliana*. We will consider the *reduced Mendoza and Alvarez-Buylla network* which has two non-trivial strongly connected symmetric components and whose asymptotic dynamics has the same attractors as the original network (see (Elena, 2009; Demongeot et al., 2010) for more details). Thus, we will focus on work with the subdigraphs G and F depicted in Fig. 7.6, where the states of the network at time t, $x_i(t) \in \{0, 1\}$, $i = 1, ..., 7$ are defined as follows:

Algorithm 7 `compatibility-test`

Require: $A, B \subseteq V$ subsets of vertices of a reduced update digraph $(G = (V, E), lab)$.

```
if B = ∅ then
  if A has no negative arc in G then
    return(1);
  else
    return(0);
  end if
else
  if exists positive arc (u, v) ∈ G such that u ∈ A and v ∈ B then
    return(0);
  else
    if exists negative arc (u, v) ∈ A ∪ B such that v ∈ A then
      return(0);
    else
      return(1);
    end if
  end if
end if
```

$x_1(t) = H(-2x_3(t-1) - 2x_2(t-1) - 1),$
$x_2(t) = H(-2x_4(t-1) - 2x_1(t-1) - 2),$
$x_3(t) = x_4(t-1),$
$x_4(t) = x_4(t-1),$

$x_5(t) = x_7(t-1),$
$x_6(t) = x_7(t-1),$
$x_7(t) = H(x_5(t-1) + x_6(t-1) - 1),$
$H(x(t)) = 1$ if $x(t) > 0$ and
$H(x(t)) = 0$ if $x(t) \leq 0.$

The principal idea is to observe all their possible dynamics when we use the $T_4 = 75$ and $T_3 = 13$ different update schedules for each one of them, respectively. But this will be done reducing the computational work thanks to the Theorem 6.1 that allows us to consider only one update schedule for each equivalence class (which represents a different update digraph).

We remark that all the following calculations are realized using the algorithms given in this chapter.

We begin the analysis with the smaller component in Fig. 7.6: F. There are $T_3 = 13$ different update schedules that can be grouped into nine different equivalence classes as it is showed in Table 7.1, each one of them yielding the same update digraph according to Theorem 6.1. These update digraphs are showed in Fig. 7.7.

The attractors of this networks are the fixed points: 000 and 111. Besides, only for the parallel update schedule (associated to F_1) there is a limit cycle $[001, 110]$. This is coherent with the results showed in (Elena, 2009; Demongeot et al., 2010).

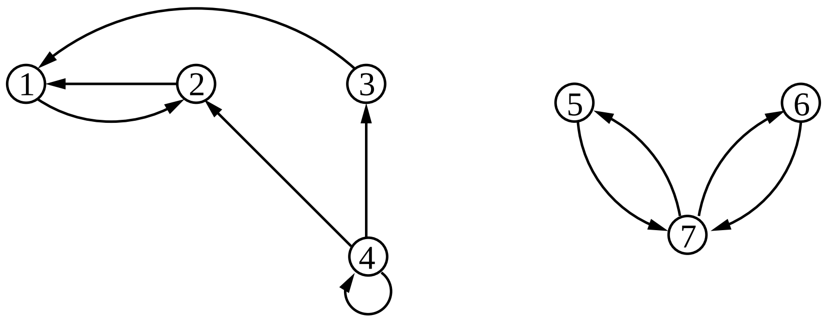


FIGURA 7.6. The subdigraph of the reduced Mendoza and Alvarez-Buylla network composed by two connected components: G (left side) and F (rigth side). The vertices 1,...,7 represent the following genes of the plant Arabidopsis thaliana involved in its floral morphogenesis: *AGAMOUS (AG)*, *APETALATA 1 (AP1)*, *TERMINAL FLOWER 1 (TF1)*, *EMBRYONIC FLOWER 1 (EMF1)*, *APETALATA 3 (AP3)*, *PISTILLATA (PI)* and *BURST FORMING UNIT (BFU)*, respectively.

$[s_1]_F$	$[s_2]_F$	$[s_3]_F$	$[s_4]_F$	$[s_5]_F$	$[s_6]_F$	$[s_7]_F$	$[s_8]_F$	$[s_9]_F$
(5,6,7)	(6)(5,7)	(5,7)(6)	(6,7)(5)	(6)(7)(5)	(7)(5,6) (7)(5)(6) (7)(6)(5)	(5)(6,7)	(5,6)(7) (5)(6)(7) (6)(5)(7)	(5)(7)(6)

TABLE 7.1. The different equivalence classes associated to F.

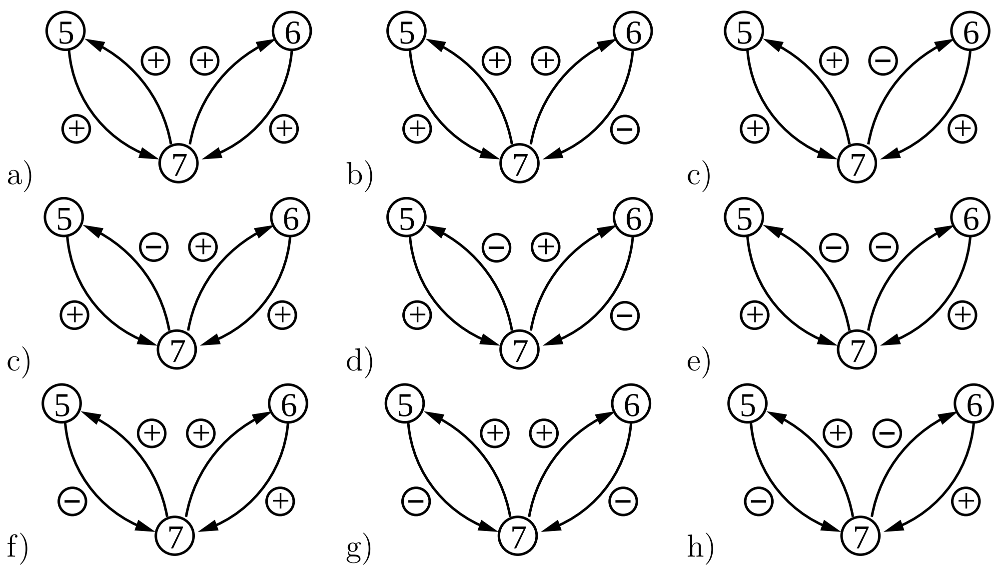


FIGURA 7.7. The update digraphs F_1,...,F_9 associated to the equivalence classes of s_1,...,s_9 are showed in the sub-figures a),...,h) respectively.

On the other hand, we can see that the equivalence classes associated to sequential update schedules have update digraphs without positive cycles according to Proposition

6.3. Furthermore, due to Theorem 7.1 and the results related in Remark 7.4, the subdigraph of H_F induced by the update digraphs above mentioned is strongly connected (see Fig. 7.8).

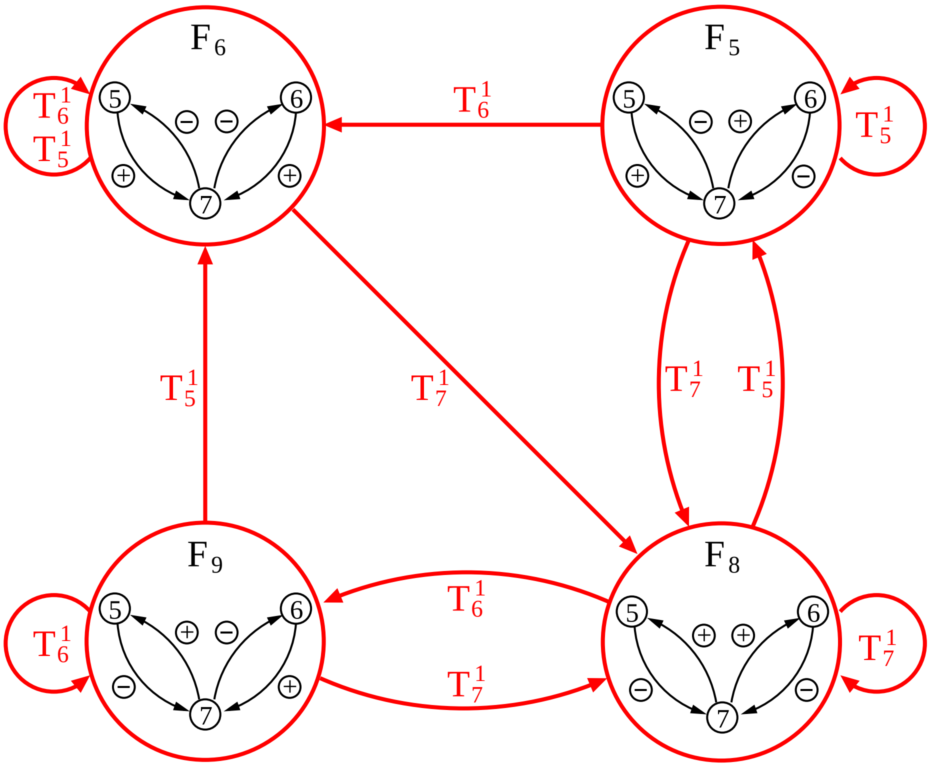


FIGURA 7.8. The subdigraph of H_F induced by the update digraphs with a sequential equivalence class associated, F_5, F_6, F_8, F_9, is strongly connected due to Theorem 7.1.

Furthermore, without knowing the dynamical behavior produced for each update schedule, we could know which dynamics is more robust in terms of the size of these equivalence classes. In this context, for example, the dynamics associated with the parallel update schedule in F_1 would be less robust than the dynamics associated with the update schedules in F_6 or F_8 (see Table 7.1).

The different dynamics of F are showed in Tables 7.2 and 7.3.

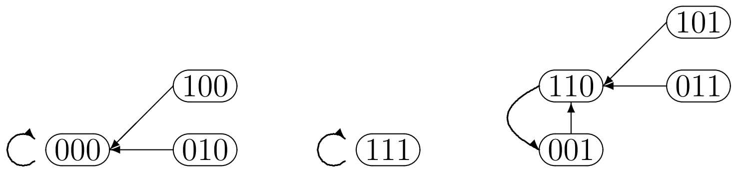
State	Sched. 1 $s_1(5)=1$ $s_1(6)=1$ $s_1(7)=1$	Sched. 2 $s_2(5)=2$ $s_2(6)=1$ $s_2(7)=2$	Sched. 3 $s_3(5)=1$ $s_3(6)=2$ $s_3(7)=1$	Sched. 4 $s_4(5)=2$ $s_4(6)=1$ $s_4(7)=1$	Sched. 5 $s_5(5)=3$ $s_5(6)=1$ $s_5(7)=2$
000	000	000	000	000	000
001	110	110	100	010	010
010	000	000	000	000	000
011	110	110	100	010	010
100	000	000	000	000	000
101	110	111	100	010	111
110	001	000	011	101	000
111	111	111	111	111	111

TABLE 7.2. Dynamics associated to F_1,...,F_5.

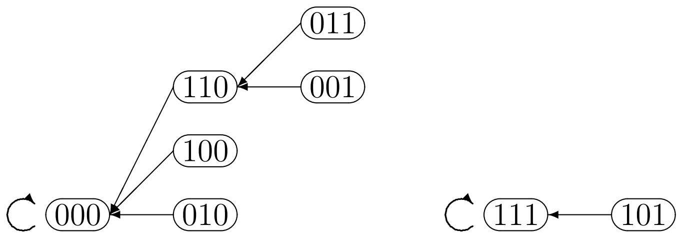
State	Sched. 6 $s_6(5)=2$ $s_6(6)=2$ $s_6(7)=1$	Sched. 7 $s_7(5)=1$ $s_7(6)=2$ $s_7(7)=2$	Sched. 8 $s_8(5)=1$ $s_8(6)=1$ $s_8(7)=2$	Sched. 9 $s_9(5)=1$ $s_9(6)=3$ $s_9(7)=2$
000	000	000	000	000
001	000	110	111	100
010	000	000	000	000
011	000	111	111	111
100	000	000	000	000
101	000	110	111	100
110	111	000	000	000
111	111	111	111	111

TABLE 7.3. Dynamics associated to F_6,...,F_9.

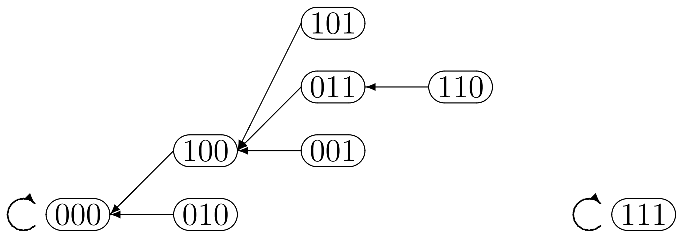
N_1 Dynamics



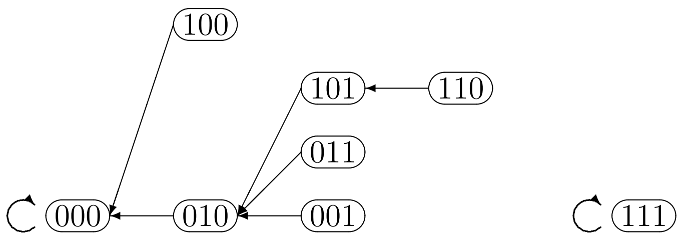
N_2 Dynamics



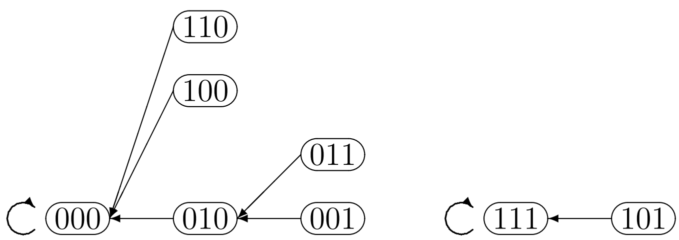
N_3 Dynamics



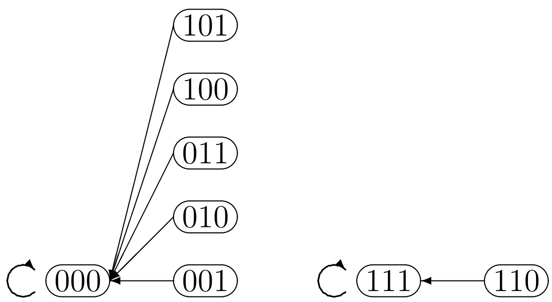
N_4 Dynamics



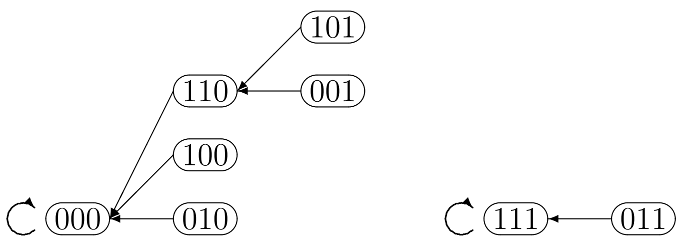
N_5 Dynamics



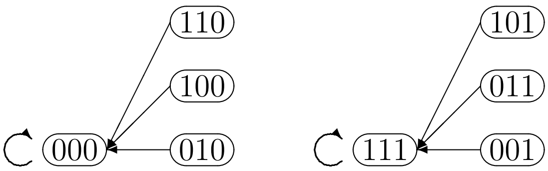
N_6 Dynamics



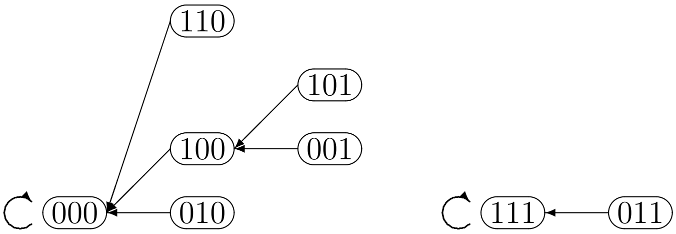
N_7 Dynamics



N_8 Dynamics



N_9 Dynamics



Analogously to what was done for F, the computational experience showed that there are twenty equivalence classes (i.e. twenty update digraphs) associated to G, but only six different dynamics where found, each one of them with three fixed points: 0011, 0100 and 1000. Also appears a limit cycle $[0000, 1100]$, but only in two of these six different dynamics (one of them is the parallel equivalence class), which is coherent with (Elena, 2009; Demongeot et al., 2010).

On the other hand, there are fourteen sequential equivalence classes which represent a strongly connected component in H_G. The smaller and bigger equivalence class of G have one and twelve elements each one, respectively.

Finally, we can combine the attractors of G and F to determine the attractor of the reduced Mendoza and Alvarez-Buylla network as in (Elena, 2009; Demongeot et al., 2010).

CHAPTER 8

Conclusions

Along this thesis we saw relationships between the feedback sets and the dynamics of Boolean networks from two different points of view: the signed digraphs and the update digraphs. Next, we will expose some conclusions of this work grouped in the following sections.

8.1 Feedback sets and signed digraphs

We have made progress studying the complexity of feedback set problems, work begun in (Montalva, 2006). We saw for the general case, that PFVS and NFVS problems are NP-complete (also their versions with arcs, PFAS and NFAS). Unfortunately, this complexity continues being kept when we add additional constraints in the structure or in the same distribution of the signs, for example in special cases of applied interest such as Kauffman or monotone networks. Nevertheless, under certain restrictions, we could have separated the complexities of FVS, PFVS and NFVS, for example in the families of complete digraphs, signed digraphs with at most k negative arcs and cyclically reducible digraphs, observing that structural properties and sign distribution can determine the complexity of these problems.

In particular, since the maximum number of fixed points of a REBN depends on a minimum positive feedback vertex set (Aracena, 2001, 2008), we can think that if we have such structural property or sign distribution that makes PFVS (or NFVS), for example polynomial, then in a way, the dynamical properties of the network should be simpler of study.

8.2 Feedback sets and update digraphs

Here, we have developed the central part of this thesis. One of the important aspects to remark is how the structural properties of the connection digraph of a network is related to the feedback sets and the update digraphs associated to it.

More precisely, we have observed monotone increase and decrease properties of the number of update digraphs and of the feedback arc sets associated to the connection digraph when its number of cycles increase (Corollary 6.8, Theorem 6.5 and Proposition 6.5). That allows us to have a general idea about all the different dynamics, bounded by the size of the set of feedback arc sets (Corollary 6.6 and Proposition 6.5).

On the other hand, it is easy to see that there is also a decreasing monotone property in the average size of the equivalence classes associated with the connection digraph, specifically, while more equivalence classes has the connection digraph, smaller is the average size of these equivalence classes. In this case, we have given necessary and sufficient conditions to ensure that the size of an equivalence class should have more than one element as well as we have proved different results related with the itself structure of the update schedules.

Furthermore, similarly to the precedent section, we have introduced new decision problems related to the update digraphs such as DU and UAS problems. Beyond of knowing their complexity, we have seen in a natural way that they are related with problems type feedback, specifically with the FAS problem.

Was observed that the property of being update digraph is invariant under removal of its arcs (Theorem 6.4) and allows us to understand better the update digraphs from that with biggest number of arcs: the complete digraph. Moreover, we saw that the maximum number of negative arcs in an update complete digraph is represented by an acyclic tournament with all its arcs negative. Therefore, we have paid special attention in these digraphs and how they are related to the update digraphs obtaining different results of combinatorial type, using graph theory and giving algorithms for determining the update digraphs as well as the update schedules associated to them.

Finally, we have studied relationships within the same update digraphs, through of simply local transformations, in order to better understand the structure of the multidigraph generated by these transformations. Thus, from the knowledge obtained, different enumeration algorithms were developed for the update digraphs and their associated update schedules.

8.3 Open problems

In this section, we list some open problems which are basically of three different kinds: complexity and algorithms, combinatorics and dynamics of REBNs.

1. To find more families of networks where the principal decision problems; PFVS, NFVS, PFAS and NFAS could be polynomial.

2. To know for which values of k, PFVS-k would be NP-complete, since for the extreme cases $k = 1$ and $k = n$ was proven to be polynomial.

3. To prove the following conjecture:

   Conjecture: *Let G be a connected digraph with $n > 1$ vertices. Then, $|[s]_G| \leq 2 \cdot T_{n-1}$ for every update schedule s.*

   This conjecture has been established because we have seen that the update digraph with the bigger equivalence class seems to be one with all its arcs positive and with only one vertex pointing to all others $n-1$ vertices (or having a single vertex that is pointed by all others). Such digraph has $2 \cdot T_{n-1}$ update schedules associated to it (see Fig. 8.1).

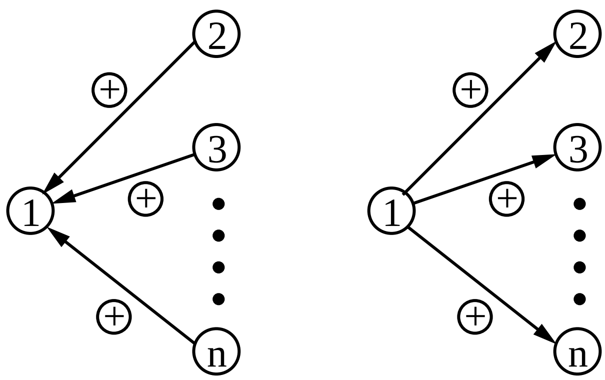


FIGURA 8.1. Two update connected digraphs having equivalence classes associated of size $2 \cdot T_{n-1}$.

4. Giving a digraph G, to study the structure of H_G restricted to non-reduced digraphs as well as the families of digraphs G having H_G strongly connected.

5. Since we know the relationship between the positive feedback vertex set and the number of fixed points of a network, it is interesting to study the influence of the negative feedback vertex set on the dynamical properties of the network.

6. To develop efficient approximation algorithms for each of the recently studied decision problems: PFVS, NFVS or their versions with arcs.

7. To study the number of update digraphs associated to digraphs which do not contain tournaments as well as to characterize the size of its equivalence classes.

8. To study the relationships between update digraphs and signed digraphs. In particular, to see how is related the positive and negative vertex set of a signed digraph to the update digraphs.

9. To study decomposition of feedback sets for digraphs in terms of the positive and negative feedback vertex sets of a signed digraph.

10. To analyze how the changes in the sign of a signed digraph are related to the size of the positive and negative feedback arc sets.

References

S. Kauffman, Metabolic stability and epigenesis in randomly connected nets, Journal of Theoretical Biology 22 (1969) 437–467.

S. Kauffman, Control circuits for determination and transdetermination, Science 181 (1973) 310–318.

S. Kauffman, The Origins of Order: Self-Organization and Selection in Evolution, Oxford University Press, New York, 1993.

J. Demongeot, Multi-stationarity and cell differentiation, J. Biol. Syst. 6 (1998) 1.

J. Demongeot, M. Kauffman, R. Thomas, Positive regulatory circuits and memory, C. R. Acad. Sci. 323 (2000) 69–80.

R. Thomas, R. D'Ari, Biological feedback, CRC Press, 1990.

E. Plahte, T. Mestl, W. Omholt, Feedback loops, stability and multistationarity in dynamical systems, J. Biol. Syst. 3 (1995) 409–413.

E. Snoussi, Necessary conditions for multistationarity and stable periodicity, J. Biol. Syst. 6 (1998) 3–9.

J. Gouzé, Positive and negative circuits in dynamical systems, J. Biol. Syst. 6 (1998) 11–15.

O. Cinquin, J. Demongeot, Positive and negative feedback: striking a balance between necessary antagonists, Journal of Theoretical Biology 216 (2) (2002) 229–241.

C. Soulé, Graphic requirements for multistationarity, ComPlexUs 1 (3) (2003) 123–133.

A. Richard, J.-P. Comet, Necessary conditions for multistationarity in discrete dynamical systems, Discrete Applied Mathematics 155 (2007) 24032413.

A. Richard, Positive circuits and maximal number of fixed points in discrete dynamical systems, Discrete Applied Mathematics 157 (2009) 32813288.

E. Snoussi, R. Thomas, Logical identification of all steady states: The concept of feedback loop characteristic states, Bull. Math. Biol. 55 (1993) 973–991.

R. Thomas, D. Thieffry, M. Kaufman, Dynamical behavior of biological regulatory networks. I. Biological role and logical analysis of feedback loops, Bull. Math. Biol. 57 (1995) 328–339.

J. Aracena, S. B. Lamine, M. Mermet, O. Cohen, J. Demongeot, Mathematical modelling in genetic networks: relationships between the genetic expression and both chromosomic breakage and positive circuits, IEEE Trans. Systems Man Cyber 33 (2003) 825–834.

R. Thomas, J. Richelle, Positive feedback loops and multistationarity, Discrete Applied Mathematics 19 (1-3) (1988) 381–396.

R. Thomas, M. Kaufman, Multistationarity, the basis of cell differentiation and memory. I. Structural conditions of multistationarity and other non trivial behavior, Chaos 11 (1) (2001) 170–179.

J. Aracena, Modelos matemáticos discretos asociados a los sistemas biológicos. Aplicación a las redes de regulación génica, PhD thesis, U. Chile and U. J. Fourier, Santiago, Chile, 2001.

J. Aracena, On the number of fixed points in regulatory boolean networks, Bulletin of Mathematical Biology 70 (5) (2008) 1398–1409.

E. Sontag, A. Veliz-Cuba, R. Laubenbacher, A. Jarrah, The effect of negative feedback loops on the dynamics of boolean networks, Biophysical Journal 95 (2) (2008) 518–526.

M. Montalva, Conjunto de vértices recubridor de los ciclos positivos en un digrafo. Engineering Thesis, Universidad de Concepción, Chile, 2006.

R. Karp, Reducibility among Combinatorial Problems, in: Complexity of Computer Computations, Plenum Press, New York, 1972.

P. Festa, P. Pardalos, M. Rsende, Feedback set problems, in: Handbook of Combinatorial Optimization, vol. A, Kluwer Academic Publishers, 1999.

F. Robert, Discrete Iterations: A Metric Study (Biomathematics), Springer-Verlag, Michigan University, 1986.

E. Goles, Comportement oscillatoire d'une famille d'automates cellulaires non uniformes. Thèse docteur ingénieur, IMAG, Université de Grenoble, Grenoble, France, 1986.

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

A. Elena, H. Ben-Amor, N. Glade, J. Demongeot, Motifs in regulatory networks and their structural robustness, IEEE BIBE' 08, IEEE Proceedings, Piscataway (2008) 234–242.

H. Ben-Amor, J. Demongeot, A. Elena, S. Sen, Structural Sensitivity of Neural and Genetic Networks, Lecture Notes in Computer Science 5317 (2008) 973–986.

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

A. Elena, Robustesse des réseaux d'automates booléens à seuil aux modes d'itération. Application a la modélisation des réseaux de régulation génétique, PhD thesis, Université Joseph Fourier (Grenoble I), Grenoble, France, 2009.

L. Salinas, Estudio de modelos discretos: estructura y dinámica, PhD thesis, Universidad de Chile, Santiago, Chile, 2008.

L. Gómez, Robustez de Ciclos Dinmicos en Redes Booleanas, Engineering thesis, Universidad de Concepción, Concepción, Chile, 2009.

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

E. L. Lloyd, M. L. Soffa, C.-C. Wang, Feedback vertex sets and cyclically reducible graphs, Journal of the ACM 32 (1985) 296–313.

B. Schwikowski, E. Speckenmeyer, On enumerating all minimal solutions of feedback problems, Discrete Applied Mathematics 117 (1-3) (2002) 253–265.

J. Demongeot, E. Goles, M. Morvan, M. Noual, S. Sené, Attraction Basins as Gauges of Robustness against Boundary Conditions in Biological Complex Systems, PLoS ONE 5 (8) (2010) doi:10.1371/journal.pone.0011793.

J. Bang-Jensen, G. Gutin, Digraphs: Theory, Algorithms and Applications, Springer-Verlag, 1979.

D. B. West, Introduction to Graph Theory, Prentice Hall, 1996.

A. Karatkevich, On algorithms for decyclisation of oriented graphs, in: Proc. Int. Workshop on Discrete-Event System Design, DESDes'01, TU of Zielona Góra, Computer Engineering and Electronics Institute, 35–40, 2001.

K. A. Hawick, H. A. James, Enumerating Circuits and Loops in Graphs with Self-Arcs and Multiple-Arcs, in: Proc. Int. Conference on Foundations of Computer Science FCS'08, CSREA Press, 14–20, 2008.

J. O. Vik, Johnson's algorithm implemented in Python for finding all elementary circuits (cycles, feedback loops,...) in a directed graph, https://networkx.lanl.gov/trac/ticket/394, 2010.

Z. Cinkir, Zhang's Conjecture and the Effective Bogomolov Conjecture over function fields, Invent. Math. 183 (2011) 517–562.

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

M. Anthony, Discrete Mathematics of Neural Networks: Selected Topics (Monographs on Discrete Mathematics and Applications), Society for Industrial and Applied Mathematics .

E. Hemaspaandra, H. Spakowski, M. Thakur, Complexity of cycle length modularity problems in graphs, in: Proc. 6th Latin American Symp. on Theoret. Informatics (LATIN '04), Springer-Verlag, LNCS 2976, 509–518, 2004.

V. Vazirani, M. Yannakakis, Pfaffian orientations, 0-1 permanents, and even cycles in directed graphs, Discrete Applied Mathematics 25 (1989) 179–190.

N. Robertson, P. Seymour, R. Thomas, Permanents, pfaffian orientations, and even directed circuits, Annals of Mathematics 150 (1999) 929–975.

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

I. Shmulevich, H. Lähdesmäki, E. R. 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.

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