Structural Aspects of Switching Classes

Structural Aspects of Switching Classes

Proefschrift

ter verkrijging van
de graad van Doctor aan de Universiteit Leiden,
op gezag van de Rector Magnificus Dr. D.D. Breimer,
hoogleraar in de faculteit der Wiskunde en
Natuurwetenschappen en die der Geneeskunde,
volgens besluit van het College voor Promoties
te verdedigen op woensdag 26 september 2001
te klokke 14.15 uur

Jurriaan Hage

geboren te Alphen aan den Rijn
in 1969

Promotiecommissie

Promotor:	Prof. dr. G. Rozenberg
Co-promotor:	Dr. T. Harju (Turun Yliopisto, Finland)
Referent:	Prof. dr. H.-J. Kreowski (Universität Bremen, Duitsland)
Overige leden:	Prof. dr. H. Ehrig (Technische Universität Berlin, Duitsland)
	Prof. dr. E. Welzl (ETH Zürich, Zwitserland)
	Dr. J. Engelfriet
	Prof. dr. J. Kok
	Prof. dr. H.A.G. Wijshoff

The work in this thesis was carried within the context of the IPA graduate school.

And then there was light...

Contents

Preface

This thesis studies the subject of switching classes of undirected graphs and switching classes of skew gain graphs. The field of research lies in graph theory with a group theoretical flavour.

A switching class is an equivalence class of graphs under the switching operation, that leaves the set of vertices unharmed, but may destroy or create new edges. In the field of switching classes of skew gain graphs the edges have a label from a group.

The subjects that are studied here are often of a combinatorial nature and investigate diverse properties of switching classes. There is a rather strong focus on algorithms, which are based on the theory presented here. The thesis also contains a number of complexity results.

The initial motivation for the subject came, in our case, from a model of networks of processors of a specific kind: processors that by an action influence all the connections they might have to other processors. Results in this field are part of the second part of the thesis. The results focus on the study of the evolutionary behaviours of such networks of processors and algorithms to make certain problems decidable and, once decidable, feasible.

During the research we found that the model we used was already defined in a simpler form in mathematics. It turned out however that many of our results were new which shows the diversity in motivation between those fields.

The first part of the thesis is of a more graph theoretic nature and concerns itself with the occurence of special kinds of graphs such as acyclic graphs, hamiltonian graphs and the like in switching classes.

Another motivation for our research has to do with problems that are difficult for graphs, e.g., NP-complete problems. In our research we found that the hamiltonian cycle problem is rather simple for switching classes, while we know it is hard for graphs. Instead of answering a difficult question for a specific graph, we might first approximate the answer by answering the question for the switching class in which that graph resides.

Of a more theoretical nature are the results that link graph theory with the theory of switching classes. For instance, it is proved in this thesis that switching classes contain at most one tree up to isomorphism. In such a way we also learn more about the nature of these types of graphs.

Much of the material in this thesis comes from articles that appeared in various places. Examples have been added to clarify the material from the papers. In some cases generalizations replace the results of the paper and new results have been added as well; in all cases we indicate where such is the case.

For completeness we give a general overview of what ended up where: the results about complexity by Ehrenfeucht, Hage, Harju and Rozenberg [12] can be found in Section 3.4. A related result about pancyclic graphs in switching classes is by Ehrenfeucht, Hage, Harju and Rozenberg [13] and appears in Section 4.1. The two papers [24] and [25] by Hage and Harju about trees and acyclic graphs in switching classes can be found in the same chapter, Sections 4.2 and 4.3.

For the skew gain graphs, Chapter 6 is based on [26] by Hage and Harju and Chapter 7 contains the material of Hage [21].

The results of Appendix A can be of interest to people programming switching classes.

Many people at LIACS helped me during my work, or indulged in my desire to propound my newest results and latest optimizations.

In between writing papers and program, I have greatly enjoyed the long discussions with Frans Birrer, Joost Engelfriet, Jano van Hemert, Hendrik Jan Hoogeboom, but also the shorter ones with Walter Kosters, Marloes van der Nat, Rudy van Vliet, Pier Frisco and Henk Goeman.

I owe a great debt to Maurice ter Beek, Sebastian Maneth and most of all to Tjalling Gelsema for carrying the burden of sharing a room with me.

Financially the work was supported (in part) by Arto Salomaa, Ralph Back (and all at TUCS) and the GETGRATS project. I am grateful to LIACS for the extra time granted me in finishing the thesis and also gaining some experience in lecturing.

I am grateful to Piet van Oostrum and Tycho Strijk for help with LaTeX and CorelDraw.

On the personal front I thank my parents and my brother Arjan for their encouragement. I am grateful to Roberto Lambooy for his driving support.

This is also a good place to mention Jaap van der Sar, Joop Minderman, Frits Feddes, Jan van de Sant, Piet van Tienhoven and Leo van de Heuvel from the energy company (G.E.B./E.W.R./Nuon) in Alphen aan den Rijn for, as they still constantly remind me, "teaching me everything I know".

Finally, I want to thank Nikè, for her constant encouragement, inspiration and doing the household chores during the busy periods. I hope to return the favour in the near future.

Jurriaan Hage
Alphen aan den Rijn, May 2001.

Chapter 1

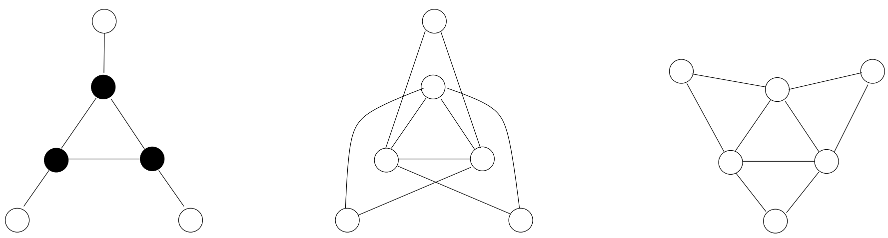
Introduction

This thesis covers a number of problems in the area of switching classes of undirected graphs and that of switching classes of directed graphs with skew gains. It is a self contained excursion into these areas. An understanding of group theory, graph theory and the theory of algorithmic complexity is presumed, but in the first two cases, we do establish terminology and notation. Knowledge of group theory is only necessary for the second part of the thesis which treats switching classes of graphs with skew gains.

We introduce the switching classes of graphs of the first part of the thesis: for a finite undirected graph $G = (V, E)$ and a subset $\sigma \subseteq V$ (called a selector), the vertex-switching of G by σ is defined as the graph $G^\sigma = (V, E')$, which is obtained from G by removing all edges between σ and its complement $V - \sigma$ and adding as edges all nonedges between σ and $V - \sigma$. The switching class $[G]$ determined by G consists of all switchings G^σ for subsets $\sigma \subseteq V$. An example of switching can be found in Figure 1.1(a) and (b). Here σ is the set of black vertices. For clarity we have redrawn Figure 1.1(b) as Figure 1.1(c).

The initiators of the theory of switching classes (and equivalently two-graphs) were Van Lint and Seidel [37]. Somewhat earlier, signed graphs, a variant of the model described above, were used in psychology by Abelson and Rosenberg [1]. For a survey of switching classes, and especially its many connections to other parts of mathematics, we refer to Seidel [43], Seidel and Taylor [44], and Cameron [7]. Because of the importance of Seidel in this field, switching in the form of the first part of the thesis is often called Seidel-switching.

The results in the first part of the thesis build on earlier results in the theory of switching classes. First, we have some complexity results for problems on graphs when lifted to switching classes. For instance, we show in Section 4.1 that we can



(a) (b) (c)

Figure 1.1: A graph and one of its switches

determine efficiently whether a switching class contains a hamiltonian graph.

In Chapter 3 we generalize to switching classes a general result in complexity theory by Yannakakis [47] and use it to prove that the embedding problem for switching classes is NP-complete.

In Section 4.2 and Section 4.3 we consider a number of problems of combinatorial nature that concern cycles in graphs. We prove that every switching class contains at most one tree up to isomorphism. A similar result holds for acyclic graphs, but then the bound is three. We close the chapter by giving a characterization of the switching classes that contain an acyclic graph. We do this by means of a set of forbidden subgraphs.

The model in the second part of the thesis is that of the switching classes of graphs with skew gains, a generalization of the model in the first part.

The vertices of a directed graph can be interpreted as processors in a network and the edges can be interpreted as the channels/connections between them. One can extend the model of the first part of the thesis by labelling each edge by an element of some (structured) set, call it Δ. The dynamics of such a network lies in the ability to change the labellings of the graph which is done by operations performed by the processors. A major aspect of the model here presented is that if a processor performs an action, it influences the labellings of all incoming edges in the same way; the same holds for the outgoing edges. In other words, we have no separate control over each edge, only over each processor. On the other hand, actions done by different processors should not interfere with each other, making this model an asynchronous one.

Ehrenfeucht and Rozenberg set forth in [16] (see also [15]) a number of axioms they thought should hold for such a network of processors. Each processor i was to have a set of *output actions* Ω_i and a set of *input actions* Σ_i. If a processor i executes one of its output actions, this entails the modification of the labels on the edges going out from i, and similarly for the input actions.

The following properties were assumed to hold for a network of processors:

A1 Any two input (output) actions can be combined into one single input (output) action.

A2 For any pair of elements $a, b \in \Delta$, there is an input action that changes a into b; the same holds for output actions.

A3 For any channel (i, j), the order of applying an input action to i and an output action to j is irrevelant.

A fourth axiom stated that every network/graph should have at least three processors. This was to make sure that no exceptions arose when deriving the most general model that upholds the axioms above; in the paper of Ehrenfeucht and Rozenberg it turned out that the model of a network of two processors is more flexible than that of more than two processors. We do not state the axiom, but shall use the model derived for more than two processors also for networks of two processors.

In the book by Ehrenfeucht, Harju and Rozenberg [14] a condition was added: each vertex can choose to stay inactive. In other words, there is a trivial action. There is however no real need for that additional axiom, because it follows from the others.

In [16] (see also [14]) it was derived that under these axioms the input (output) actions of every vertex are the same and form a group. Also, the sets of input and output actions coincide, but an action will act differently on incoming and outgoing edges, as evidenced by the asymmetry in (5.2) in Chapter 5. The difference is made explicit by an anti-involution δ, which is an anti-automorphism of order at most two on the group of actions. The notion of anti-involution generalizes that of group inversion. The result of this will be that if a channel between processors i and j is labelled with a, then the channel from j to i will be labelled with $\delta(a)$. This generalizes the gain graphs of [50] and the voltage graphs of [20].

Note that Axiom A1 refers to the fact that groups are closed under the group operation, Axiom A2 refers to the fact that $a^{-1}b$ and ba^{-1} are elements of the group (and can hence be chosen as the actions in a processor), and Axiom A3 corresponds to the associativity of the group operator.

As we shall see later the graphs labelled with elements from a fixed group Δ (and under some fixed anti-involution of that group), called skew gain graphs in the following, are partitioned into equivalence classes. These equivalence classes capture the possible outcomes of performing actions in the vertices and as such capture the potential behaviours and resulting states of the system from a certain initial state. The actions generalize the selectors of the first part of the thesis and shall be modelled by selecting a value from the group Δ in each vertex; we can always select the identity of the group in a vertex for the trivial action. Although the equivalence classes themselves are usually considered static objects, it is not hard to see that there is also a notion of change or dynamics: applying a selector to a skew gain graph yields a new skew gain graph on the same underlying network of processors, but possibly with different labels. For this reason the equivalence classes were called dynamic labelled 2-structures in [16].

When we are given a graph and two different labellings of this graph with elements of a group, a question that arises is: is there a selector, i.e., a sequence of actions, mapping the first into the second? This problem, called the membership or equivalence problem is treated in Chapter 7. The inspiration for this chapter came from an investigation of the sizes of the equivalence classes, here treated in Chapter 6. The results of the membership problem can also be used more generally: suppose we have been given a skew gain graph, where a gain, say a, on the edge from i to j has some interpretation, such as "processor j is waiting for processor i". Then it is possible to determine whether there is a possible state of the system, i.e., a skew gain graph, that contains some interesting or forbidden configuration of labels. For instance, we might detect that one of the possible states of the system includes a directed cycle between processors with all the channels labelled with a; the system is deadlocked. Sometimes these questions can even be answered when the number of labels is infinite, in which case the switching classes have infinite size. An illustration is given in Figure 1.2. Here we are interested to know whether from the configuration of the left network, we can arrive by some sequence of actions in the deadlocked configuration on the right.

We close the thesis with an overview of directions for future research in both areas.

Many of the results in the thesis were obtained by first examining the problem area by means of a computer program. For instance in the case of characterizing the acyclic switching classes by forbidden subgraphs (Section 4.3) we used software written in `C` and `Scheme` extensively. Because of the combinatorial explosion we were forced to spend some time on optimizing our algorithms. The resulting techniques and reflections on experiences in this respect are decribed in Appendix A and B respectively.

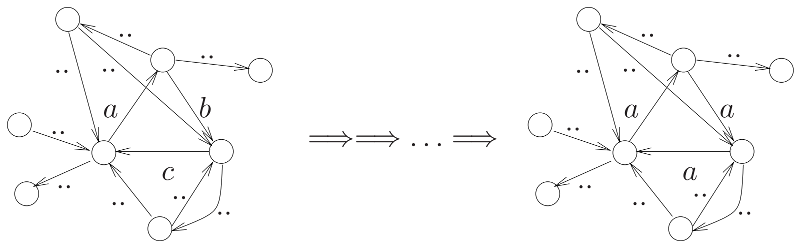


Figure 1.2: A part of a network and a possible behaviour

The references at the end of the thesis are a good starting point for articles and books related to the material treated in this thesis. For a significantly larger annotated bibliography we refer to the dynamic survey of Zaslavsky from 1999 [51] that includes also a number of related areas of interest.

Chapter 2

Preliminaries

We assume that the reader is familiar with the elements of group theory and graph theory, but parts of these elements are formulated here in order to establish our notation. Also to fix notation, we start with a short section on sets, functions and (binary) relations.

2.1 Sets, functions and relations

For a finite set V, $|V|$ denotes its *cardinality* or *size*. Recall that a set of one element is a *singleton*, one of two elements a *doubleton* and so forth.

We denote the union, intersection and difference of two sets V and W by $V \cup W$, $V \cap W$ and $V - W$ respectively. The *cartesian product* of V and W is the set $V \times W = \{(v, w) \mid v \in V, w \in W\}$.

A set of nonempty subsets of V is a *partition* of V if these subsets are pairwise disjoint and their union equals V.

We shall often identify a set $V' \subseteq V$ with its *characteristic function* $V' : V \to \{0, 1\}$, where we use the convention that for $v \in V$, $V'(v) = 1$ if and only if $v \in V'$. The *symmetric difference* of two sets V and W, denoted by $V \ominus W$, is defined to be equal to $(V - W) \cup (W - V)$. It can also be formulated by its characteristic function:

$$(V \ominus W)(v) = V(v) + W(v), \text{ for all } v \in V \cup W \tag{2.1}$$

where $+$ is addition modulo 2.

Some special sets are $\mathbf{N}$, $\mathbf{Z}$, $\mathbf{R}$ and $\mathbf{R}^+$ which denote the sets of *positive integers*, *integers*, *reals* and *positive reals*, respectively. If we want to add zero explicitly to $\mathbf{N}$ or $\mathbf{R}^+$ we write $\mathbf{N}_0$ or $\mathbf{R}_0^+$ respectively.

As usual, a function f, from its *domain* V to its *range* W, is denoted by $f : V \to W$. In this thesis *all functions are total*, i.e., a value $f(v) \in W$ exists for all $v \in V$. The *composition* of two functions $f : V \to W$ and $g : U \to V$ is $f \cdot g$ and is such that $(f \cdot g)(u) = f(g(u))$ for all $u \in U$. Sometimes we may omit the function composition operator $\cdot$ writing fg for $f \cdot g$.

The *identity function* on a set V is denoted by id_V or simply id. The set of *fixed points* of a function f is

$$\mathrm{Fix}(f) = \{v \mid f(v) = v\} .$$

For a function $f : V \to W$ and element $w \in W$, we define $f^{-1}(w) = \{v \mid f(v) = w\} \subseteq V$. A function is *injective* if for all $w \in W$, $|f^{-1}(w)| \leq 1$ and it is *surjective* if for all $w \in W$, $|f^{-1}(w)| \geq 1$. A function is *bijective* if it is both injective and surjective, i.e., all the sets $f^{-1}(w)$ are singletons. In the literature injective is often referred to as one-to-one, surjective as onto and bijective as one-on-one. A function $f : V \to V$ is a *permutation* on V if it is a bijection.

The *restriction* of a function $f : V \to W$ to a set $V' \subseteq V$ is the function denoted by $f|_{V'} : V' \to W$ such that $f|_{V'}(v) = f(v)$ for all $v \in V'$.

For a set X, $\pi = (x_1, \ldots, x_n)$ with $x_i \in X$ for $i = 1, \ldots, n$ is a *sequence* over X. The length of π is n; the sequence of length 0 is denoted by λ. The sequence π is called *closed* if $x_1 = x_n$. We use $a:\pi$ to denote the sequence $(a, x_1, \ldots, x_n)$.

A set of pairs $R \subseteq V \times V$ is a *(binary) relation* on V, which is the *underlying set* of R. The relation R is *reflexive* if $(v, v) \in R$ for all $v \in V$. It is *symmetric* if for each $(v, w) \in R$ also $(w, v) \in R$ and it is *transitive* if $(v, w), (w, z) \in R$ imply $(v, z) \in R$. A relation is *anti-symmetric* if $(v, w), (w, v) \in R$ imply $v = w$.

A relation that is reflexive, symmetric and transitive is called an *equivalence relation*. An equivalence relation R partitions the underlying set into equivalence classes where such a class is defined as follows: $[v]_R = \{w \mid (v, w) \in R\}$. An elementary result is that if $w \in [v]_R$, then $[v]_R = [w]_R$. Hence it does not matter which of the elements of an equivalence class is used as a representative for that class. Sometimes, v and w are called *equivalent up to* R.

Given a possibly infinite number of mutually disjoint subsets V_i of V, a *transversal* $\mathcal{T}$ of the V_i is a set that satisfies $|\mathcal{T} \cap V_i| = 1$ for all i. Because the V_i are disjoint, there is a bijection between the V_i and the elements of $\mathcal{T}$. Each of the elements of the transversal should be interpreted as uniquely representing one of the sets in the partition. In the literature the demand of the disjointness of the V_i is not standard. However, it is the only circumstance in which we need it.

For two equivalence relations R_1 and R_2 on V, R_1 is a *refinement* of R_2, if $R_1 \subseteq R_2$. The concept of refinement is illustrated in Figure 2.1. Here, the classes of R_2 are given by the solid lines, and the classes of R_1 are defined by the solid and dashed lines combined. For instance, $[v]_{R_1}$ equals $[v]_{R_2}$. It is also true that $[w_1]_{R_2} = [w_2]_{R_2}$, but $[w_1]_{R_1} \neq [w_2]_{R_1}$. By refining we may split up equivalence classes into smaller ones.

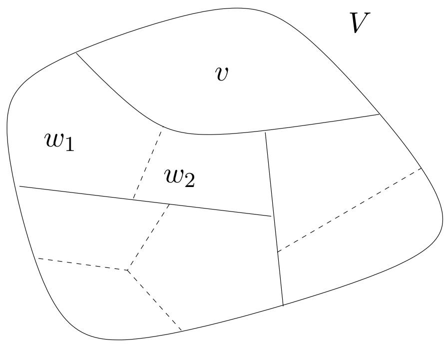


Figure 2.1: A refinement

2.2 Basic group theory

This section contains an introduction into group theory that should suffice to understand all group theory used in this thesis. Much of the contents of this section comes from the highly readable and precise book of Rotman [41].

A *group* is a pair $\Gamma = (C, \circ)$, where the set C is the *carrier* of Γ and

- $\circ$ is a binary associative operation on C, i.e., it is a total binary function $\circ : C \times C \to C$ such that $(a \circ b) \circ c = a \circ (b \circ c)$ for all $a, b, c \in C$,
- there exists a unique *identity element* $1_\Gamma \in C$ such that for all $a \in C$, $a \circ 1_\Gamma = 1_\Gamma \circ a = a$,
- for each element $a \in C$, there exists a unique element a^{-1}, the *inverse* of a, such that $a \circ a^{-1} = a^{-1} \circ a = 1_\Gamma$.

The *trivial group* containing only the identity element will be denoted by $\{1\}$. A group is *finite* if the carrier is finite. The *order* of a finite group equals the cardinality of the carrier. The function $^{-1}$ mapping each element of a group Γ to its inverse is called the *group inversion*.

We want to be able to tell when two groups are "the same". Let $\Gamma_1 = (C_1, \circ)$ and $\Gamma_2 = (C, \Box)$ be groups. A function $h : C_1 \to C_2$ is a *homomorphism* from Γ_1 to Γ_2 if

$$h(a \circ b) = h(a) \Box h(b), \text{ for all } a, b \in C_1 .$$

The groups Γ_1 and Γ_2 are called *isomorphic*, denoted by $\Gamma_1 \cong \Gamma_2$, if there exists a bijective homomorphism $h : \Gamma_1 \to \Gamma_2$. An isomorphism from a group to itself is called an *automorphism*. The set of automorphisms on a group is denoted by $\mathrm{AUT}(\Gamma)$.

Following the usage in group theory, we will in the future often omit the operation from expressions, e.g., writing ab instead of $a \circ b$. Also we will identify Γ with C and will thus write simply $a \in \Gamma$ instead of $a \in C$ for a group $\Gamma = (C, \circ)$. The other way around we sometimes give a set and treat it as a group; it shall be obvious from the context what the operation should be.

A group Γ is called *abelian* if for all $a, b \in \Gamma$, $ab = ba$. In words: all pairs of elements $a, b \in \Gamma$ *commute*. The operation of a group is usually referred to as multiplication; in the context of abelian groups we sometimes use "+" for the operation and use the name addition.

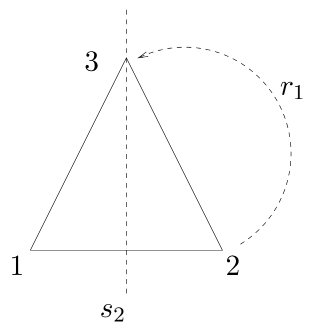


Figure 2.2: The group S_3 schematically

Example 2.1
(1) $(\mathbf{Z}, +)$ is the group of integers under addition. The inverse of $a \in \mathbf{Z}$ is $-a$ and 0 is the identity.
(2) $(\mathbf{R}^+, *)$ is the group of positive reals under multiplication, where 1 is the identity and the inverse of a is its reciprocal $1/a$.
(3) $\mathbf{Z}_n = (\{0, \ldots, n-1\}, +)$ is the group of integers modulo n. As with $\mathbf{Z}$ the identity is 0, but the inverse of $a \in \{0, \ldots, n-1\}$ is $n - a \mod n$.

(4) The *symmetric group* S_n is the group of permutations on $\{1, \ldots, n\}$; the order of this group is $n!$. The symmetric group S_3 consists of all the rotations and reflections of the equilateral triangle as illustrated in Figure 2.2. The three rotations are denoted by r_0, r_1 and r_2 (r_i stands for rotation by angle $i{*}120°$ anticlockwise). For $i = 0, 1, 2$, the reflection s_i is a reflection about the axis going through the vertex $i + 1$. The operation of this group is the composition of functions: $s_2 r_1$ means first rotating by 120 degrees and then reflecting about the s_2 axis. This transformation maps $1, 2$ and 3 to $1, 3$ and 2, respectively and thus is equal to s_0. Figure 2.3 contains a table of values ab for $a, b \in S_3$. $\diamond$

b a ab	r_0	r_1	r_2	s_0	s_1	s_2
r_0	r_0	r_1	r_2	s_0	s_1	s_2
r_1	r_1	r_2	r_0	s_2	s_0	s_1
r_2	r_2	r_0	r_1	s_1	s_2	s_0
s_0	s_0	s_1	s_2	r_0	r_1	r_2
s_1	s_1	s_2	s_0	r_2	r_0	r_1
s_2	s_2	s_0	s_1	r_1	r_2	r_0

Figure 2.3: The multiplication table of S_3

In Example 2.1 all groups are abelian except the group S_3: as can be seen from the table $s_0 s_1 = r_1 \neq r_2 = s_1 s_0$. The group S_3 is especially interesting, because it is the smallest nonabelian group. This makes it a useful group for illustrating nonabelian aspects of our theory.

We continue now with the definition of a subgroup. A nonempty subset Δ of Γ is a *subgroup* of Γ if it is a group under the operation of Γ. The fact that Δ is a group means that it contains the identity element of Γ, is closed under the operation of Γ and for every $a \in \Delta$, also $a^{-1} \in \Delta$.

To verify whether a subset of a group is a subgroup the following result can be used, Theorem 2.2 in Rotman [41].

Lemma 2.2
A nonempty subset Δ of a group Γ is a subgroup of Γ if and only if for all $a, b \in \Delta$ also $ab^{-1} \in \Delta$.

An important subgroup is the *centre* of a group Γ. It is

$$\mathrm{Z}(\Gamma) = \{x \in \Gamma \mid xy = yx \ \text{ for all } \ y \in \Gamma\} \ .$$

The centre of a group contains exactly those elements in the group that commute with all elements.

We can generalize the definition of centre into that of the *centralizer* of a set $A \subseteq \Gamma$:

$$C(A) = \{x \in \Gamma \mid ax = xa \ \text{ for all } \ a \in A\} \ .$$

Obviously $\mathrm{Z}(\Gamma)$ equals $C(\Gamma)$ and $C(\{1_\Gamma\}) = \Gamma$.

Example 2.3
(1) The centre of an abelian group is, of course, the group itself and all centralizers

coincide.
(2) The centre of S_3 is $\{r_0\} \cong \{1\}$. Furthermore,, $C(\{r_0\}) = S_3$, $C(\{r_1\}) = C(\{r_2\}) = \{r_0, r_1, r_2\}$ and $C(\{s_0\}) = \{r_0, s_0\}$, but $C(\{r_1, s_0\} = C(\{r_1\}) \cap C(\{s_0\}) \cong \{1\}$. All these identities can be read from the previous multiplication table for S_3.◇

For $n > 0$, the nth power of $a \in \Gamma$ is $a^n = \overbrace{aa\ldots a}^{n}$. We define $a^0 = 1_\Gamma$. For any element a of a group Γ, we can construct the cyclic subgroup generated by a, denoted by $\langle a \rangle$, that contains all powers $a^0 = 1_\Gamma, a, a^2, \ldots$ of a. The element a is called a *generator* of $\langle a \rangle$. We define the *order* of a, denoted by $\#a$, to be the order of the cyclic group $\langle a \rangle$ (if it is finite); in other words it is the smallest number $k > 0$ such that $a^k = 1_\Gamma$. If a group Γ equals $\langle a \rangle$ for some $a \in \Gamma$ it is called *cyclic*. Note that two cyclic groups are isomorphic if and only if they are of the same order or both are infinite.

More generally, for any $X \subseteq \Gamma$ there exists always the subgroup $\langle X \rangle$ of Γ generated by X; it consists of all finite products of elements of X and their inverses. Again if $\langle X \rangle = \Gamma$, then X is said to generate Γ. If there exists a finite set X which generates Γ, then Γ is said to be *finitely generated*.

Example 2.4
(1) For the cyclic group $\mathbf{Z}_n$, $\langle 2 \rangle = \mathbf{Z}_n$ if $n > 2$ is odd, but $\langle 2 \rangle \cong \mathbf{Z}_{n/2}$ if $n > 2$ is even.
(2) For the group S_3, $\langle r_1 \rangle = \{r_0, r_1, r_2\}$ and $\langle s_0 \rangle = \{r_0, s_0\}$. For $X = \{r_1, r_2\}$ the subgroup of S_3 generated by X equals $\langle r_1 \rangle$. On the other hand, taking $X = \{r_1, s_0\}$, $\langle X \rangle = S_3$. One can check that no set of less than two elements will generate S_3, so this set of generators is minimal, but note that it is not unique.
(3) There is no finite set X such that $\langle X \rangle$ equals the group $(\mathbf{R}^+, *)$, because $\langle X \rangle$ is countable. ◇

We introduce some notation for the *product* of two subsets Δ_1 and Δ_2 of Γ:

$$\Delta_1 \Delta_2 = \{xy \mid x \in \Delta_1, y \in \Delta_2\} .$$

If $\Delta_2 = \{x\}$, then we write $\Delta_1 x$, and analogously if Δ_1 is a singleton.

A subset S of Γ is a *right coset* of a subgroup Δ of Γ if $S = \Delta x = \{yx \mid y \in \Delta\}$ for some $x \in \Gamma$. If $x \in \Delta$ then obviously Δx equals Δ, since Δ is closed under the operation of Γ. Analogously, a *left coset* of a subgroup Δ of Γ is a subset $S = x\Delta$ for some $x \in \Gamma$.

The following result, Lemma 2.5 in Rotman [41], can be used to verify whether two cosets are the same.

Lemma 2.5
Let Δ be a subgroup of Γ. It holds that $\Delta x = \Delta y$ (respectively $x\Delta = y\Delta$) if and only if $xy^{-1} \in \Delta$ (respectively $x^{-1}y \in \Delta$).

If Γ is nonabelian it is possible that the left and right cosets of a subgroup of Γ are not the same, although one can prove that there are equally many of them, as the following example taken from Rotman [41] illustrates.

Example 2.6
In a previous example we found that $S = \langle s_0 \rangle = \{r_0, s_0\}$. The left cosets are $S = r_0 S = s_0 S$, $r_1 S = s_2 S = \{r_1, s_2\}$ and $r_2 S = s_1 S = \{r_2, s_1\}$. The right cosets however are $S = S r_0 = S s_0$, $S r_1 = S s_1 = \{r_1, s_1\}$, and finally $S r_2 = S s_2 = \{r_2, s_2\}$.

From Lemma 2.5 it follows rather easily that the left (right) cosets of a subgroup are either disjoint or exactly the same. Consequently, given a subgroup Δ of Γ, and by computing the left (right) cosets of Δ we get a partition of Γ. This is a very useful fact, since it will enable us to define the quotient of a group with respect to a subgroup: the quotient has cosets as its elements, and the subset product as its operator. It turns out however that the notion of quotient is only well-defined as a group for the subset product of normal subgroups, which we turn to next.

The *conjugate* of $x \in \Gamma$ by $a \in \Gamma$ is the element axa^{-1} (in the literature, the latter is usually denoted by x^a). Now, a subgroup Δ of Γ is a *normal subgroup* of Γ if it is closed under conjugation with every element of Γ. Clearly all subgroups of an abelian group are normal.

It can be proved that for normal subgroups the left and right cosets coincide. In fact, this characterizes a normal subgroup. Hence, in the context of normal subgroups we can omit the qualifications “left” and “right” for cosets.

Let Δ be a normal subgroup of Γ. The set of cosets of Δ forms a group under the operation of subset product. It is the *quotient group* and shall be denoted by Γ/Δ.

Example 2.7
(1) Let n be any integer, then $\mathbf{Z}n$ is the set of (positive and negative) multiples of n. Adding two multiples of n yields another multiple of n, so it is not surprising that $\mathbf{Z}n$ is a subgroup of $\mathbf{Z}$ and since $\mathbf{Z}$ is abelian, it is a normal subgroup. The n cosets of this subgroup are $\{in + j \mid i = 0, \pm 1, \pm 2, \ldots\}$ for $j = \{0, \ldots, n-1\}$. Each of these cosets is an element of the quotient group $\mathbf{Z}/\mathbf{Z}n$, which is isomorphic to the cyclic group of order n, $\mathbf{Z}_n$.
(2) The group $\Delta = \{r_0, s_0\}$ is a subgroup of S_3. This subgroup is not normal, since $r_1 \Delta {r_1}^{-1} = \{r_0, s_1\} \neq \Delta$.

Now we have shown how to collapse a group into a smaller group in a well-defined way, we shall take the opposite approach, combining groups into larger ones. The most obvious way is by means of a construction called the direct product of two groups, similar to the cartesian product of sets.

For two groups Γ_1 and Γ_2 the *outer direct product* is the group $\Gamma = \Gamma_1 \times \Gamma_2$ which has as elements the ordered pairs (x, y) where $x \in \Gamma_1$ and $y \in \Gamma_2$. The operation of Γ is defined elementwise: $(x_1, y_1)(x_2, y_2) = (x_1x_2, y_1y_2)$ for $(x_1, y_1), (x_2, y_2) \in \Gamma$. Obviously, Γ contains as subgroups $\Gamma_1 \times \{1_{\Gamma_2}\}$ and $\{1_{\Gamma_1}\} \times \Gamma_2$, isomorphic to Γ_1 and Γ_2 respectively. It can be easily verified that $\times$ is commutative, i.e., $\Gamma_1 \times \Gamma_2 \cong \Gamma_2 \times \Gamma_1$ and associative, i.e., $(\Gamma_1 \times \Gamma_2) \times \Gamma_3 \cong \Gamma_1 \times (\Gamma_2 \times \Gamma_3)$.

The inner direct product approaches a group from the other direction: how can we factor a group? The basic result is that if we have two normal subgroups Γ_1 and Γ_2 of Γ such that $\Gamma_1 \cap \Gamma_2 = \{1_\Gamma\}$ and $\Gamma_1\Gamma_2 = \Gamma$, then $\Gamma \cong \Gamma_1 \times \Gamma_2$. The group Γ is called the *inner direct product* of Γ_1 and Γ_2. Note that the difference with the outer direct product is that in the inner case, Γ_1 and Γ_2 are *themselves* subgroups of Γ. Since both ways yield isomorphic groups we shall not distinguish between them.

Note that in general a group Γ can allow more than one decomposition. It is possible that $\Gamma = \Gamma_1 \times \Gamma_2 = \Gamma_1 \times \Delta$ (inner direct products) where $\Gamma_2 \neq \Delta$. Of course, Γ_2 and Δ are then isomorphic. On the other hand Γ may allow decompositions that are not even determined up to isomorphism.

An alternative condition for normal subgroups Γ_1 and Γ_2 of Γ to have $\Gamma_1\Gamma_2 = \Gamma$ (by inner direct product) is for each $x \in \Gamma$ to have a unique expression $x = y_1y_2$, where $y_i \in \Gamma_i$ for $i = 1, 2$.

Example 2.8
Consider the group $\mathbf{Z}_6$. This group is the inner direct product of its normal subgroups $\{0, 3\}$ and $\{0, 2, 4\}$. The latter two are isomorphic to $\mathbf{Z}_2$ and $\mathbf{Z}_3$ respectively. Obviously, $\mathbf{Z}_6 \neq \mathbf{Z}_2 \times \mathbf{Z}_3$, but $\mathbf{Z}_6 \cong \mathbf{Z}_2 \times \mathbf{Z}_3$. $\diamond$

Let $\Gamma = \Gamma_1\Gamma_2$ be a direct product, and let $\alpha : \Gamma \to \Gamma$ be any function. We define the *projections* $\alpha^{(i)} : \Gamma \to \Gamma_i$ for $i = 1, 2$ by: for each $x \in \Gamma$ let $\alpha(x) = \alpha^{(1)}(x)\alpha^{(2)}(x)$, where $\alpha^{(1)}(x) \in \Gamma_1$ and $\alpha^{(2)}(x) \in \Gamma_2$. Because every $x \in \Gamma$ can be uniquely written as x_1x_2 for $x_i \in \Gamma_i$, these functions are well defined.

In the context of abelian groups, the direct product is often called the *direct sum* and is denoted by $\oplus$.

In the following we will give the main theorem of finitely generated abelian groups, which yields a way of decomposing any such group into a direct sum of groups of a special kind.

A group is a *p-group* if every element has order a power of a fixed prime q. In the context of abelian groups a p-group is often called a *primary group*.

Example 2.9
The group $\mathbf{Z}_6$ is not a p-group (3 has order 2 and 2 has order 3), but $\mathbf{Z}_4$ is (every element has either order 1, 2 or 4, all powers of 2). $\diamond$

Now we can formulate the following result.

Theorem 2.10 (Fundamental Theorem On Finitely Generated Abelian Groups)
For every finitely generated abelian group Γ, it holds that

$$\Gamma \cong \mathbf{Z}_{p_1^{m_1}} \oplus \mathbf{Z}_{p_2^{m_2}} \oplus \cdots \oplus \mathbf{Z}_{p_r^{m_r}} \oplus \overbrace{\mathbf{Z} \oplus \mathbf{Z} \cdots \oplus \mathbf{Z}}^{m_{r+1}}$$

where $r \geq 0$, $m_j > 0$, $m_{r+1} \geq 0$, p_j prime for $j = 1, \ldots, r$ and $p_i \leq p_{i+1}$ for $i = 1, \ldots, r-1$.

In other words, every finitely generated abelian group Γ is (isomorphic to) a direct sum of cyclic p-groups and infinite cyclic groups, and the number of summands of each kind depends only on Γ. Note that the only source of "infiniteness" arises from $\mathbf{Z}$.

The importance of this theorem is that it gives us a way to prove something by induction: if a property holds for all p-groups and for $\mathbf{Z}$, and it is preserved by direct sum and isomorphisms, then it holds for all finitely generated abelian groups.

Example 2.11
Since the group $\mathbf{Z}_6$ is not a p-group (see Example 2.9), we should be able to find a nontrivial decomposition into p-groups. Indeed $\mathbf{Z}_6$ is isomorphic to $\mathbf{Z}_2 \oplus \mathbf{Z}_3$, and this expression is unique up to the order of the terms.

In Theorem 2.10, $p_i < p_{i+1}$ is not good enough, because the group $\mathbf{Z}_4$ is not ismorphic to $\mathbf{Z}_2 \oplus \mathbf{Z}_2$: the former has an element of order four, the second does not. $\diamond$

We shall now give definitions and results that pertain to groups of permutations, stabilizers and orbits.

The group S_3 is an example of a group of permutations. The importance of this type of groups is that every group is isomorphic to a subgroup of some permutation group and hence the theory of groups can also be studied by looking only at permutation groups and their subgroups.

Given a nonempty set X, S_X is the *symmetric group* on X, where the elements are the permutations on X and the operation is simply function composition. A subgroup of a symmetric group is called a *permutation group*. A bijection $f : X \to Y$ can be straightforwardly extended to an isomorphism $a \to f \cdot a \cdot f^{-1}$ between S_X and S_Y. This implies that restricting ourselves to S_n in the finite case is not a restriction at all (see Example 2.1).

Let X be a set and let S be a subgroup of S_X. We define two elements $x, y \in X$ to be *S-equivalent*, denoted by $x \sim_S y$, if there is some permutation $z \in S$ such that $z(x) = y$. The corresponding equivalence classes $[x]$ are called the *orbits* of S. It is sometimes said that the group S *acts* on X.

The notion dual to orbit is that of stabilizer. For an element $x \in X$, the *stabilizer* of x is a subgroup of S defined as

$$\mathrm{Stab}(x) = \{z \in S \mid z(x) = x\} \ .$$

There is a strong relation between the cardinality of the orbit $[x]$ and the stabilizer of x. The following holds for any $x \in X$ if the set X is finite

$$|S| = |\mathrm{Stab}(x)| \cdot |[x]| \ . \tag{2.2}$$

Example 2.12
Let $S = S_3$ and the set $X = \{1, 2, 3\}$. In this case there is only one orbit since each element of X can be mapped to any other. The stabilizer of x is $\mathrm{Stab}(x) = \{r_0, s_{x-1}\}$ for $x = 1, 2, 3$. Filling in the numbers into Equation (2.2) yields for $x = 1$, $6 = |S| = |\mathrm{Stab}(1)||[1]| = 2 \cdot 3$. If we set $S = \{r_0, s_0\}$, then there are two orbits $\{1\}$ and $\{2, 3\}$. The corresponding stabilizers are $\mathrm{Stab}(1) = \{r_0, s_0\} = S$ and $\mathrm{Stab}(2) = \mathrm{Stab}(3) = \{r_0\}$. $\diamond$

2.3 Basic graph theory

The following introduction to graph theory follows the main lines of the book by Harary [28].

For a set V, $E(V) = \{\{v, w\} \mid v, w \in V, v \neq w\}$ denotes the set of all unordered pairs of distinct elements of V. In the following we will usually write vw (and, in the first part of this thesis, also wv) for $\{v, w\}$. The vertices v and w are the *endpoints* of the edge $vw \in E$.

The graphs in this part of the thesis will be undirected, finite and simple, i.e., they contain no loops or multiple edges: a *graph* G is a tuple (V, E), where V is a finite set of *vertices* and $E \subseteq E(V)$ is the set of *edges*. We use $V(G)$ and $E(G)$ to denote V and E respectively and $|V|$ and $|E|$ are called the *order*, respectively, the *size* of G.

Analogously to sets, a graph $G = (V, E)$ is sometimes identified with the characteristic function $G : E(V) \to \mathbf{Z}_2$ of its edges as follows: $G(xy) = 1$ if $xy \in E$, and $G(xy) = 0$ if $xy \notin E$. Later we shall use both of these notations for graphs.

For graphs G and H on the same set of vertices, we can define $G + H$ by

$$(G + H)(e) = G(e) + H(e) \text{ for } e \in E(V) \tag{2.3}$$

where the operation is that of $\mathbf{Z}_2$.

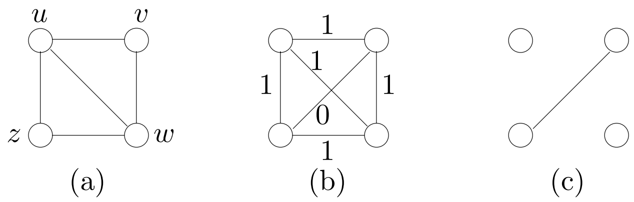


Figure 2.4: A graph, its characteristic function, and its complement

Lemma 2.13
The graphs with vertex set V form an abelian group under the $+$ operation.
Proof:
The set of graphs on V is obviously closed under $+$. The identity element of this group is the graph $(V, \emptyset)$; the inverse of a graph is the graph itself. □

We extend the operation $+$ to graphs on sets of vertices V and V' respectively, by first extending them to graphs on $V \cup V'$ and setting all new edges to 0.

The disjoint union of two graphs G and H on the other hand is denoted $G \cup H$. We use $k \cdot G$ as shorthand for the disjoint union of k copies of G.

For a graph $G = (V, E)$ and $A \subseteq V$, let $G|_A = (A, E \cap E(A))$ denote the *subgraph* of G *induced* by A. Hence, $G|_A : E(A) \to \mathbf{Z}_2$. More generally a graph $G' = (V', E')$ is a *subgraph* of G if $V' \subseteq V$ and $E' \subseteq E$. A *spanning subgraph* is a subgraph containing all vertices V; in other words we obtain a spanning subgraph of a graph by deleting some edges.

Let $G = (V, E)$ be a graph. If $u, v \in V$, then $G + uv$ (respectively $G - uv$) is $(V, E \cup \{uv\})$ (respectively $(V, E - \{uv\})$) the graph in which the edge uv is added to (respectively removed from) G. To add and remove vertices we define $G + u$ as the graph $(V \cup \{u\}, E)$, and $G - u$ as $G|_{V-\{u\}}$. More generally we can write $G - I = G|_{V-I}$ for $I \subseteq V$

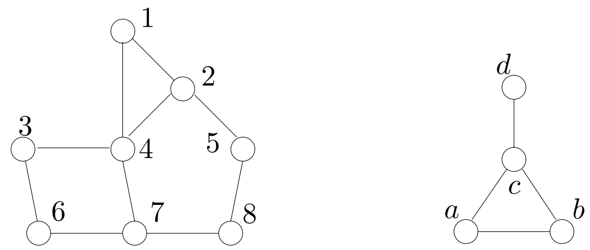
The *complement* of G is $\overline{G} = (V, E(V) - E)$.

Example 2.14
Let $G = (\{u, v, w, z\}, \{\{u, v\}, \{u, w\}, \{u, z\}, \{v, w\}, \{w, z\}\})$, or, in abbreviated form, $E(G)$ can be written as $\{uv, uw, uz, vw, wz\}$. This graph can be pictorially presented as in Figure 2.4(a). The corresponding characteristic function is depicted in Figure 2.4(b) as a complete graph labelled with 0 or 1. The subgraph of G induced by $A = \{u, v, z\}$ is $G|_A = (\{u, v, z\}, \{uv, uz\})$. The graph $G' = (\{u, v, z\}, \{uv\})$ is a subgraph of G, but it is neither induced nor spanning. The complement of G, $\overline{G}$, is given in Figure 2.4(c). ◇

Two graphs $G = (V, E)$ and $H = (V', E')$ are *isomorphic*, denoted by $G \cong H$, if there is a bijection $\psi : V \to V'$ such that

$$\psi(x)\psi(y) \in E' \text{ if and only if } xy \in E \ .$$

If we want to make ψ explicit we write $G \overset{\psi}{\cong} H$. Since ψ, the isomorphism, is a bijection between V and V' the two sets of vertices are necessarily of the same size. In fact, the only difference between isomorphic graphs lies in the identities of the vertices. If G and H are the same graph, then ψ is called an *automorphism*. The set of automorphisms of G is denoted $\mathrm{AUT}(G)$.



(a) (b)

Figure 2.5: Graphs, isomorphisms and embeddings

In graph theory isomorphic graphs are usually identified, hence graphs are considered *up to isomorphism.* However, throughout this thesis we generally take the point of view that *identities of vertices do matter* and consequently consider graphs *up to equality.* If we deviate from this, we shall say so explicitly.

The graph G can be *(fully) embedded into* H if G is isomorphic to an (induced) subgraph of H. The corresponding bijection ψ is an injection from $V(G)$ into $V(H)$ and we denote this by writing $G \overset{\psi}{\hookrightarrow} H$; if the embedding is full, then this shall be made explicit in the text.

Example 2.15
Let G and G' be the graphs of Figure 2.5(a) and (b) respectively. For the bijection $\psi = \{(c,4),(a,1),(b,2),(d,7)\}$, $G' \overset{\psi}{\hookrightarrow} G$ and this embedding is even full. Note that ψ is not the only full embedding of G' into G.

The graph of Figure 2.4(a) cannot be embedded into G. On the other hand, the graph G' can be embedded into the graph of Figure 2.4(a), but not fully.

The graph G' has a nontrivial automorphism, i.e. different from the identity function; it is $\{(a,b),(b,a),(c,c),(d,d)\}$. If $G \overset{\psi}{\hookrightarrow} H$ and $\phi \in \mathrm{AUT}(G)$, then clearly $G \overset{\psi\cdot\phi}{\hookrightarrow} H$. $\diamond$

Let $G = (V, E)$ be a graph. Two vertices $x, y \in V$ are *adjacent* (in G) if $xy \in E$. The set of vertices adjacent to x (in G) is denoted by $N_G(x)$, the neighbours of x. The *degree* of a vertex $x \in V$ is $d_G(x) = |N_G(x)|$. If G is clear from the context, then we may omit it as a subscript. A vertex of degree zero is called *isolated* or a *horizon*, and a vertex of degree one is called a *leaf.* A leaf adjacent to z is called a *leaf at* z.

A sequence of vertices $\pi = (v_1, \ldots, v_k)$, $k > 0$, is a *walk* in G if v_i is adjacent to v_{i+1} for $i = 1, \ldots, k-1$. If a walk traverses only distinct edges, then we call it a *trail.* More formally, a trail is a walk such that $v_i v_{i+1} \neq v_j v_{j+1}$ for all $1 \leq i < j \leq k-1$. Note that for an undirected graph, the edge vw is the same as the edge wv and hence in a trail we cannot traverse an edge twice, even if it would happen in different directions. If $v_i \neq v_j$ for $1 \leq i < j \leq k$, then the sequence π is called a *path.* Note that a path is also a trail, but not necessarily vice versa.

If for a walk $\pi = (v_1, \ldots, v_k)$ it holds that $v_1 = v_k$ then it is *closed.* If π is a closed walk, then it is a *cycle* if $k \geq 4$ and $v_i \neq v_j$ for $1 \leq i < j \leq k-1$ are distinct. Appending the first vertex v_1 of a path $\pi = (v_1, \ldots, v_k)$ to π yields a cycle, $(v_1, \ldots, v_k, v_1)$, if and only if the closing edge $v_k v_1$ exists and $k \geq 3$. The length of π equals $k-1$, the number of edges that occur in it. Abusing our notation we sometimes say $(v_1, \ldots, v_k)$, $v_1 \neq v_k$, is a cycle, meaning that $(v_1, \ldots, v_k, v_1)$ is a cycle.

The graph G is *connected* if from every vertex of G there is a path to every other vertex of G; if this is not the case, then the graph is called *disconnected*. A maximal connected (induced) subgraph of a graph is a *connected component*, or simply a *component*. Note that a component is necessarily an induced subgraph.

A set of vertices $A \subseteq V(G)$ is a *clique* of G if for every $v, w \in A$: if $v \neq w$, then $vw \in E(G)$. The notion complementary to clique is an *independent set*, where each vertex is not adjacent to any other vertex in the set.

We proceed by listing some well known types of graphs. Let $\overline{K}_V = (V, \emptyset)$ and $K_V = (V, E(V))$ be the *discrete graph* and the *complete graph* on V respectively, and let $K_{A,V-A} = (V, A \times (V - A))$ denote the *complete bipartite graph* with the partition $\{A, V - A\}$. In a complete bipartite graph $K_{A,V-A}$ all edges between A and $V - A$ exist, while A and $V - A$ themselves are independent. If the sets of vertices themselves are irrelevant we write K_n and $K_{k,m}$, where $n = |V|$, $k = |A|$, and $m = |V - A|$. Note that A or $V - A$ can be empty; hence the discrete graph is also regarded as a complete bipartite graph. In general, a graph G is *bipartite* if the set of vertices $V(G)$ can be partitioned into two sets V_1 and V_2 such that for all $xy \in E(G)$, $x \in V_1$ and $y \in V_2$ or vice versa. It is well known that a graph is bipartite if and only if it has no cycles of odd length if and only if its vertices can be partitioned into two independent sets.

A graph $G = (V, E)$ is an *acyclic graph*, or a *forest*, if it does not have any cycles. A connected acyclic graph is a *tree*. A well-known property of acyclic graphs is that $|E| = |V| - C$, where C is the number of components of G. Hence in the case of trees, the number of edges is one less than the number of vertices. Another useful property of trees is that every pair of vertices has a unique path connecting them.

The *trivial tree* consists of one vertex. A *rooted tree* is a tree T with an indicated vertex $u = \mathrm{root}(T)$. Recall that the *level* of a vertex v in a rooted tree is the number of vertices on the path from the root to v. Hence the level of the root is 1. The *height* of a tree is the level of the lowest leaf in the tree minus one. Hence the height of the trivial tree is 0.

If a spanning subgraph $G' = (V, E')$ of $G = (V, E)$ is acyclic and adding any edge from $E - E'$ back to G' introduces a cycle, then we call G' a *spanning forest* of G. The edges in $E - E'$ are called *chords*. A basic property is that G' has exactly as many components as does G. If G' is connected, then it is called a *spanning tree*. Because the number of vertices and components are equal for a graph and all of its spanning forests, and hence all spanning forest have the same amount of edges, it should be clear that whatever spanning forest G' is chosen, $|E| - |E'|$ is a constant. This number is called the *cyclomatic number* of G and is denoted with $\xi(G)$.

Example 2.16
Recall the graph G from Figure 2.5(a). A possible spanning tree of G is indicated in Figure 2.6 by the solid edges. The chords are indicated by the dotted lines. Note that adding any of the chords to the spanning tree introduces a unique cycle. The cyclomatic number of G is equal to the number of chords, three in this case. ◇

For any $n > 0$, P_n is a graph on n vertices that is a path of length $n - 1$. A graph C_n is a graph on n vertices that is a cycle of length n; a cycle C_3 is often called a *triangle* for obvious reasons. Another graph that is often encountered is a *star graph* on n vertices; it is a tree $K_{1,n-1}$. Examples of these types can be found in Figure 2.7.

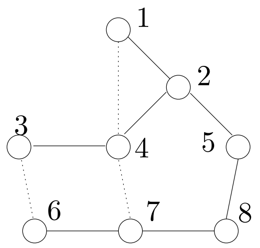


Figure 2.6: A decomposition of the graph of Figure 2.5(a)

.

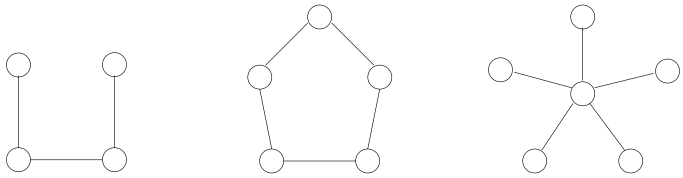


Figure 2.7: P_4, C_5 and $K_{1,5}$ respectively

.

A graph G of order n is *hamiltonian* if it has a spanning cycle (of length n, which thus contains all vertices). We call a graph *pancyclic* if it contains a cycle of length k for all $3 \leq k \leq n$, where $n \geq 3$. In other words, all C_k's ($3 \leq k \leq n$) can be embedded into G. Obviously, every pancyclic graph is also a hamiltonian graph. Bondy conjectured in [4] that almost all nontrivial general graph properties that imply hamiltonicity also imply pancyclicity. In Section 4.1 we shall see an example of this.

Example 2.17
The graphs C_n, for $n \geq 3$, are hamiltonian graphs, but only C_3 is pancyclic. A less trivial example of a pancyclic graph is given in Figure 2.5(a).

The graph in Figure 2.8(a) is called the *crown graph*. One can easily verify that this graph is not hamiltonian. The graph in Figure 2.8(b), however, does have a hamiltonian cycle, more than one in fact. ◇

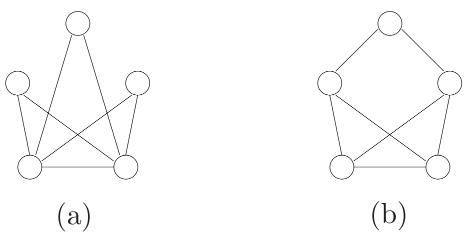


Figure 2.8: A nonhamiltonian and a hamiltonian graph

Part I

Switching Classes of Graphs

Chapter 3

Switching Classes

In this chapter we shall give definitions and some elementary results known from the theory of switching classes; briefly we shall touch on the subject of history and describe a few equivalent models. The switching classes defined in this chapter are an important special case of the switching classes of graphs with skew gains that are the subject of the second part of this thesis. Except for Section 3.4 the results in this chapter are mostly well known and usually rather straightforward. In Section 3.4 we give a few results on the complexity of certain problems of graphs when lifted to switching classes. (The results in this section are from Ehrenfeucht, Hage, Harju and Rozenberg [12] unless otherwise indicated.) For example, we generalize to switching classes a general complexity result by Yannakakis and we show that the embedding problem for switching classes, can G be embedded in some graph in $[H]$, is NP-complete. As an example of the reduction of a problem for graphs to the corresponding problem for switching classes, we show that determining whether a switching class contains a 3-colourable graph is NP-complete.

3.1 Definitions

The initiators of the theory of switching classes and two-graphs were Van Lint and Seidel [37]. Somewhat earlier, signed graphs, a slightly more general variant, were used in psychology by Abelson and Rosenberg [1].

For a survey of switching classes, and especially its many connections to other parts of mathematics, we refer to Seidel [43], Seidel and Taylor [44], and Cameron [7]. Recently a book by Ehrenfeucht, Harju and Rozenberg was published on 2-structures that has a number of chapters on switching classes [14].

For a graph $G = (V, E)$ and a function $\sigma : V \to \mathbf{Z}_2$ (called a *selector*) the *switch* of G by σ is defined as the graph $G^\sigma = (V, E')$, where for each $uv \in E(V)$ with exactly one of u and v in σ, we add uv to E if $uv \notin E$, and we remove uv from E if $uv \in E$. For a singleton selector $\sigma = \{x\}$ we write G^x instead of $G^{\{x\}}$.

Recall from the preliminaries that a selector can be interpreted as a set and we sometimes do so in this part of the thesis.

Interpreting the graph G as the characteristic function of the set of its edges, as explained in the preliminaries, we have for all $xy \in E(V)$,

$$G^\sigma(xy) = \sigma(x) + G(xy) + \sigma(y) \ . \qquad (3.1)$$

Example 3.1
Let G be the graph of Figure 3.1(a) and let $\sigma = \{2, 7, 8\}$, indicated in the picture of

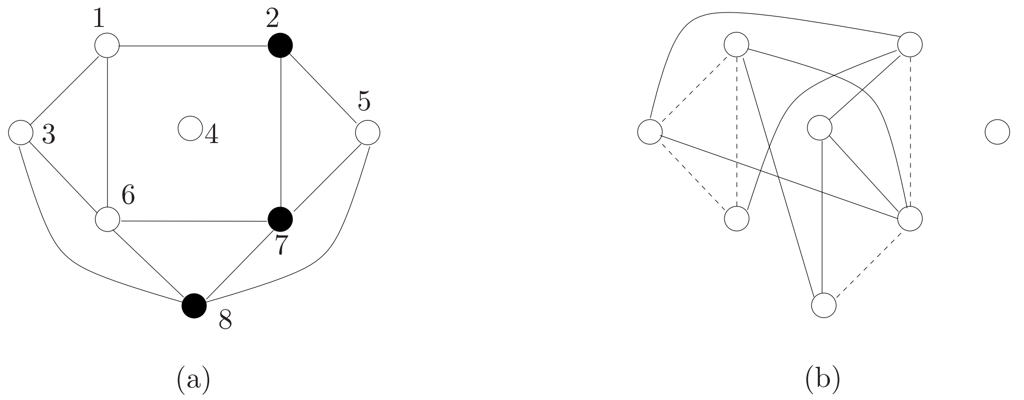


Figure 3.1: A graph and one of its switches

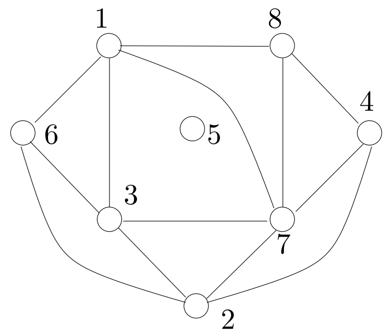


Figure 3.2: The switch of Figure 3.1(b) drawn differently

G by the black vertices. The graph G^σ is the graph of Figure 3.1(b). Note that the dashed edges are just edges of the graph, but they are special in the sense that they lie within either σ or its complement and hence are not changed. In Figure 3.2 we have reorganized the positions of the vertices to show that the switch is very similar to the original graph: up to isomorphism they differ by one edge. $\diamond$

We continue now with some easy, but useful, observations.

Lemma 3.2
For a graph $G = (V, E)$ and $\sigma \subseteq V$, it holds that

i. $G|_\sigma = G^\sigma|_\sigma$,

ii. $G|_{V-\sigma} = G^\sigma|_{V-\sigma}$, and

iii. $G^\sigma = G^{V-\sigma}$.

In view of the first two equalities of Lemma 3.2 we call σ *constant* on all sets X such that $X \subseteq \sigma$ or $X \subseteq V - \sigma$.

The set

$$[G] = \{G^\sigma \mid \sigma \subseteq V\}$$

is called the *switching class* of G. The graph G is called a *generator* of the switching class in some of the literature. In Figure 3.3 we have listed a complete switching class of 32 graphs (up to equality). This switching class is also used in Example 3.19.

We first prove that a switching class is an equivalence class of graphs.

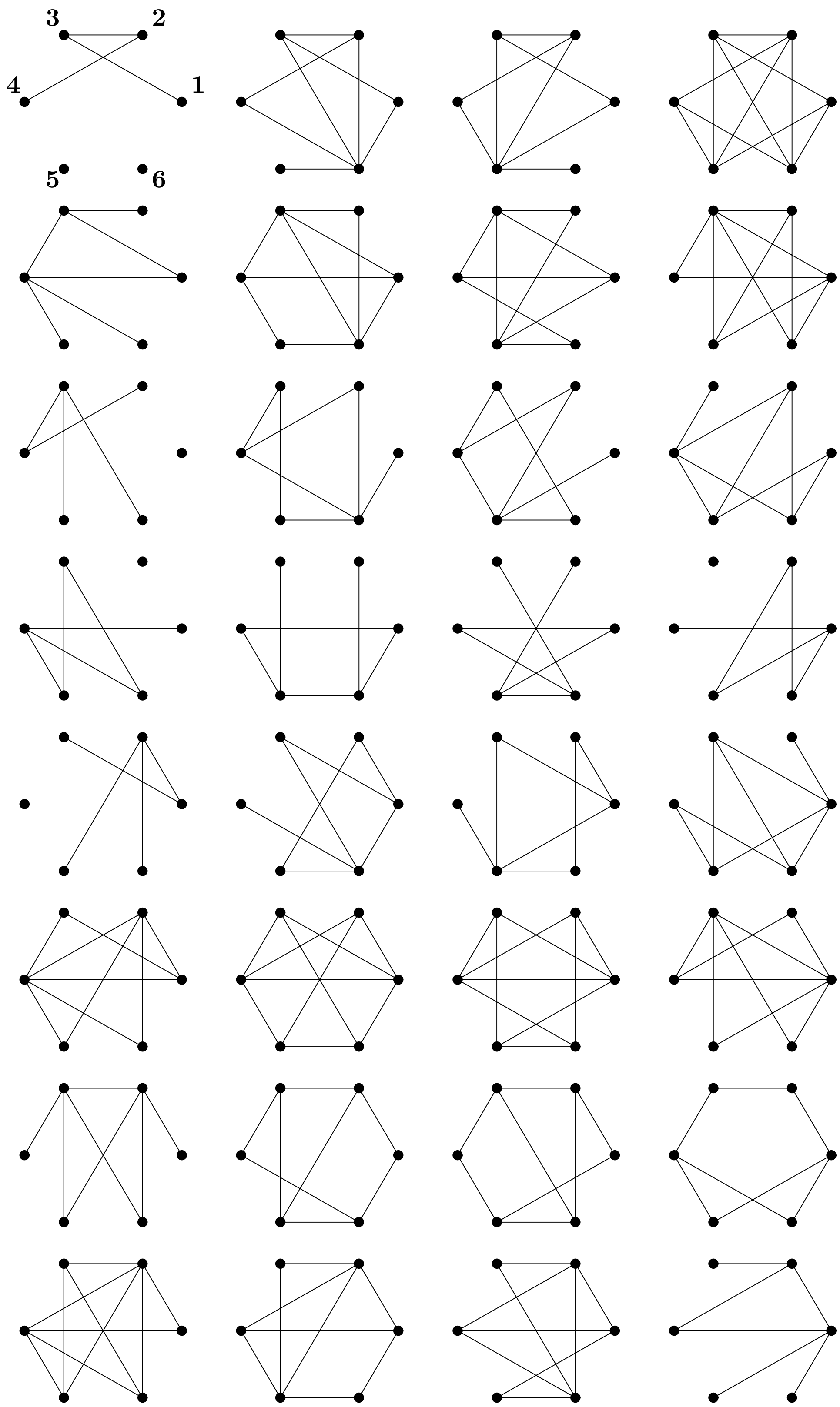


Figure 3.3: A switching class

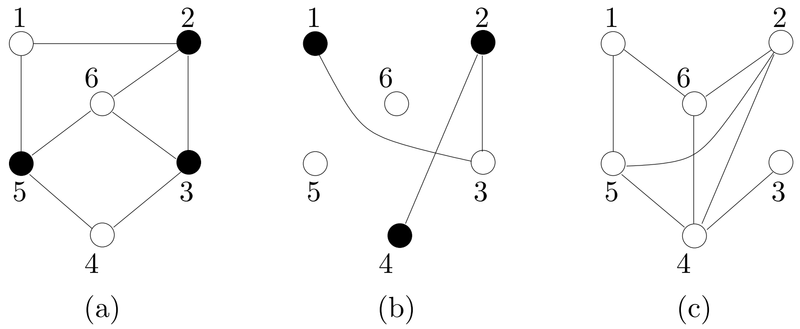


Figure 3.4: Illustrating transitivity

Lemma 3.3
For a graph $G = (V, E)$ and $\sigma_1, \sigma_2 \subseteq V$, $(G^{\sigma_1})^{\sigma_2} = G^{\sigma}$ where $\sigma = \sigma_1 \ominus \sigma_2$. More specifically, $(G^{\sigma})^{\sigma} = G$, and $G^{\emptyset} = G$.
Proof:
For $u, v \in V$, $(G^{\sigma_1})^{\sigma_2}(uv) = \sigma_2(u) + G^{\sigma_1}(uv) + \sigma_2(v) = \sigma_2(u) + \sigma_1(u) + G(uv) + \sigma_1(v) + \sigma_2(v) = \sigma(u) + G(uv) + \sigma(v)$. Clearly, $(\sigma_1 \ominus \sigma_2)(u) = \sigma_1(u) + \sigma_2(u)$ for all u.
The other two equations are trivial. □

Lemma 3.3 proves that switching is a reflexive, symmetric and transitive operation. Hence switching classes are equivalence classes of graphs. The underlying equivalence relation $\sim$ on graphs is simply this: $G \sim G'$ if and only if $G' = G^{\sigma}$ for some selector σ. As a result, a switching class can be represented by any of its elements.

A consequence of transitivity is also that every selector can be mimicked by a suitable number of singleton selectors executed in sequence, one for each vertex in the selector. We shall put this property to good use in Appendix A.4 where we show how to generate the graphs in a switching class efficiently.

Example 3.4
Let G_α be the graphs of Figure 3.4(α), for $\alpha = a, b, c$. Again, the selectors are indicated by the black vertices. We have $\sigma_a = \{2, 3, 5\}$ and $\sigma_b = \{1, 2, 4\}$. The symmetric difference of σ_a and σ_b is $\sigma = \{1, 3, 4, 5\}$ and one can verify that $G_b = G_a^{\sigma_a}$, $G_c = G_b^{\sigma_b}$ and indeed $G_c = G_a^{\sigma}$. ◇

3.2 Basic properties of switching classes

The following lemma is well known, see [43].

Lemma 3.5
The switching class $[\overline{K}_V]$ equals the set of all complete bipartite graphs on V.
Proof:
Given any complete bipartite graph $K_{\sigma,V-\sigma}$, we can obtain it from $\overline{K}_V$ by switching with respect to σ. Clearly, every switch $\overline{K}_V^{\sigma}$ is a complete bipartite graph $K_{\sigma,V-\sigma}$.□

The complete bipartite graphs figure also in another way: the graphs G^{σ} and G differ only on edges between σ and $V - \sigma$ and thus exactly all those edges exist in $G^{\sigma} + G$. Hence

Lemma 3.6
It holds that $G^\sigma + G = K_{\sigma,V-\sigma}$.

Lemma 3.7
It holds that $G^{\sigma_1} = G^{\sigma_2}$ if and only if $\sigma_1 = \sigma_2$ or $\sigma_1 = V - \sigma_2$.
Proof:
By Lemma 3.3, $G^{\sigma_1} = G^{\sigma_2}$ if and only if $(G^{\sigma_1})^{\sigma_1} = (G^{\sigma_2})^{\sigma_1}$, and again by Lemma 3.3, $G = (G^{\sigma_2})^{\sigma_1}$. Then $\sigma_1 \ominus \sigma_2$ must be either empty or equal to V by Lemma 3.2. This is only possible if $\sigma_1 = \sigma_2$ or $\sigma_1 = V - \sigma_2$. □

Corollary 3.8
A switching class on V has $2^{|V|-1}$ graphs.
Proof:
There are exactly $2^{|V|}$ subsets of V, but σ and $V - \sigma$ yield the same switch. □

The most obvious way to list all graphs in $[G]$ is to switch with respect to all subsets of $V(G) - \{x\}$ for some $x \in V(G)$. There is a more efficient way however: in Appendix A.4 we show that there is a sequence of $2^{|V|-1}$ *singleton* selectors that, if applied one after the other, will generate exactly the set of graphs $[G]$.

A very useful result in switching classes is the following, where we show that within a switching class we can "force" a certain vertex to have a given set of vertices as its neighbours. This result can be stated in more general terms, but for our purpose it will suffice. In the second part of this thesis we will formulate this result in all its generality.

Lemma 3.9
Let $G = (V, E)$ be a graph, $u \in V$ and $A \subseteq V - \{u\}$. Then there exists a unique graph $H \in [G]$ such that the neighbours of u in H are the vertices in A.
Proof:
The vertex u is isolated in $G_u = G^{N(u)}$, where $N(u)$ is the set of neighbours of u in G. Switching G_u with respect to A connects u to every vertex in A (and no others) yielding H.

To show that H is unique: let H' be such that $N_{H'}(u) = A$. Since H and H' are in the same switching class, $H + H'$ is a complete bipartite graph by Lemma 3.6, say $K_{B,V-B}$. Since u has the same neighbours in both, u is isolated in $K_{B,V-B}$. Hence, $K_{B,V-B}$ is the discrete graph and, consequently, $H = H'$. □

By taking the empty set for the set of neighbours, we obtain

Corollary 3.10
For $x \in V(G)$, $G^{N_G(x)}$ is the unique switch of G in which x is isolated.

and from this

Corollary 3.11
It holds that $G \in [H]$ if and only if for any $x \in V(G)$, $H^{N_H(x)} = G^{N_G(x)}$.

Example 3.12
A typical example of neighbourhood forcing can be found in Figure 3.4. The neighbours of 6 are $\{2, 3, 5\}$ and switching with respect to this set removes all connections to 6, making it an isolated vertex in Figure 3.4(b). Switching with respect to the set of vertices 6 should be adjacent to, here $\{1, 2, 4\}$ is chosen, then yields the wanted graph of Figure 3.4(c). ◇

The most important result concerning switching classes is the following one [43].

Theorem 3.13
Let $G = (V, E)$, $H = (V, E')$ and $x \in V$. Then $G \in [H]$ if and only if for all tripletons $T = \{x, y, z\} \subseteq V$, the parity of the number of edges in $G|_T$ equals that of $H|_T$.
Proof:
Let $T = \{x, y, z\} \subseteq V$ be a tripleton and let $H = G^\sigma$. Now, either σ is constant on T, in which case no edges are changed and hence the parity stays the same, or σ selects 1 or 2 vertices from T. In both cases, 1 and 2, exactly two edges change and again the parity is the same. Hence, switching leaves the parity of edges in T unchanged.

For the other direction, let G and H be such that they have the same parity of edges for all tripletons $T = \{x, y, z\}$, where x is a fixed vertex. By the first part of the proof, $G' = G^{N_G(x)}$ has the same parity of edges in T as G. The same holds for $H' = H^{N_H(x)}$. If G and H belong to different switching classes, then $G'(vw) \neq H'(vw)$ for some edge $vw \in E(V - \{x\})$ by Corollary 3.11. But then G' and H' differ on the parity of $\{x, v, w\}$, because G' and H' agree on all edges that have x as an endpoint. Hence G and H differ on the parity of edges of T; a contradiction. $\square$

A *repetition-free sequence* $(v_1, \ldots, v_p)$ is such that $v_i \neq v_{i+1}$ for $i = 1, \ldots, p-1$. The same element may occur in such a sequence more than once, but never will two of them be adjacent.

Theorem 3.14
Let $G = (V, E)$ be a graph, σ a selector of G, and $\pi = (v_1, \ldots, v_p)$ be a closed repetition-free sequence over V. Then the parities of edges along π in G and G^σ are the same.
Proof:
Let G, π and σ be as stated. The parity of edges along π in G can be defined in terms of addition modulo two as

$$G(v_1v_2) + G(v_2v_3) + \ldots + G(v_{p-1}v_p) \tag{3.2}$$

In G^σ this value becomes

$$\sigma(v_1) + G(v_1v_2) + \sigma(v_2) + \sigma(v_2) + G(v_2v_3) + \ldots + \sigma(v_{p-1}) + G(v_{p-1}v_p) + \sigma(v_p)$$

which reduces, by the fact that $a + a = 0$ under addition modulo two, to

$$\sigma(v_1) + G(v_1v_2) + G(v_2v_3) + \ldots + G(v_{p-1}v_p) + \sigma(v_p)$$

and because $v_1 = v_p$ and addition is commutative, we finally obtain (3.2). $\square$

We define the *complemented switching class* of a switching class as follows:

$$\overline{[G]} = \{\overline{H} \mid H \in [G]\} \ .$$

Lemma 3.15 [Ehrenfeucht, Hage, Harju and Rozenberg [11]]
For a graph $G = (V, E)$, $[\overline{G}] = \overline{[G]}$. Furthermore, if $|V| \geq 3$ then $[G] \cap [\overline{G}] = \emptyset$.
Proof:
We show first that for a graph $G = (V, E)$ and $\sigma \subseteq V$: $\overline{G^\sigma} = \overline{G}^\sigma$. Indeed, let $x, y \in V$.

Then $\overline{G^\sigma}(xy) = 1 - (\sigma(x) + G(xy) + \sigma(y)) = \sigma(x) + (1 - G(xy)) + \sigma(y) = \overline{G}^\sigma(xy)$, because $(1-a) + (1-b) = a + b$ for $a, b \in \mathbf{Z}_2$.

Let $V' \subseteq V$ be any tripleton. Since the parity of edges in V' in G is obviously different from the parity of edges in V' in $\overline{G}$ the additional claim clearly holds by Theorem 3.13. □

Let $G = (V, E)$ be a *maximum graph* in its switching class, i.e., G has the maximum number of edges among the graphs in its switching class. This graph is not unique. Note that $d_G(u) \geq (n-1)/2$ for all $u \in V$.

Example 3.16
The graphs of Figure 2.8(a) and (b) can be obtained from each other by switching with respect to the middle vertex of the top three. It turns out that they both have the maximum number of edges in their switching class.

To see this let G be the graph of Figure 2.8(a) and $n = 5$ the order of the graph. We need only consider switches of one or two vertices, because the other switches are either complements or trivial. Because the degree of each vertex is at least $(n-1)/2$, we cannot increase the number of edges in the graph by switching at a single vertex. Now, let $\sigma = \{x_1, x_2\}$ be a doubleton selector. Because of symmetry, there are three possibilities: both are top vertices, both are bottom vertices, or we have one of each type. We obtain a triangle with two leaves at one of its vertices, an edge with three isolated vertices, a C_4 with a leaf at one of its vertices, respectively. Hence Figure 2.8(a) and (b) are maximum graphs in their switching class. ◇

A graph G is called an *even* (*odd*) graph if all vertices are of even (odd) degree. A graph is *eulerian* if there exists a closed walk that traverses each edge exactly once. A well known result by Euler is that every eulerian graph is a connected even graph and vice versa.

Seidel [42] proved that if G is of odd order, then the switching class $[G]$ contains a unique even graph. Such a result is interesting because it tells us that every graph having an odd number of vertices can be constructed from an even graph using the rather simple transformation of switching. (See also Mallows and Sloane [39] and Cameron [7] for the connection of eulerian graphs to switching.)

Theorem 3.17 [Seidel [42]]
Let G be a graph of odd order. Then $[G]$ contains a unique even graph.
Proof:
Let G be a graph of odd order. The even graph can be obtained by switching with respect to the set σ of vertices in G that have odd degree, because there are an odd number of vertices of even degree and an even number of vertices of odd degree. For instance, if a vertex v had odd degree and an even number of its neighbours had even degree, then v loses an even number of neighbours, but gains an odd number of neighbours, since the number of vertices of even degree is odd. In sum, v will have an even number of neighbours in G^σ. The other three cases can be treated in a similar way. Hence G^σ has only vertices of even degree.

Let $H = G^\sigma$ be an even graph on vertices V and τ a nontrivial selector. We can assume without loss of generality that τ has even cardinality. Let $v \in \tau$. We can split $V - \{v\}$ into four sets of vertices based on the fact whether a vertex is connected to v or not and the fact whether they are or are not in τ. The vertex v will be connected to all vertices in $A = (V - \tau) \cap (V - N_H(v))$, i.e., the vertices outside τ that it was not connected to, and $B = \tau \cap N_H(v)$, i.e., the neighbours it was connected to. We now show $|A \cup B|$ is odd, which implies that H^τ is not even. Assume $|B|$ is odd. Then $|(V - \tau) \cap N_H(v)|$ is odd, because v was even. Because $|B \cup \{v\}|$ is even and $|\tau|$ is even, we find that $|\tau - (B \cup \{v\})|$ is even. It follows that $|A|$ is even, because the order of H is odd. Similar reasoning for the case that $|B|$ is even proves that H can not be switched into an even graph, unless the switch is trivial, which proves that the switch is unique. □

For graphs of even order, a switching class $[G]$ can contain only noneulerian graphs, e.g. take the switching class $[P_4]$. However, it holds that either $[G]$ has no even and no odd graphs, or exactly half of its graphs are even while the other half are odd, as first proved in Ehrenfeucht, Hage, Harju and Rozenberg [12] (but see also Ehrenfeucht, Harju and Rozenberg [14]).

To see this, define $u \sim_G v$, if $d_G(u) \equiv d_G(v) \pmod 2$, that is, if the degrees of u and v have the same parity. This relation is an equivalence relation on $V(G)$.

Assume then that the order n of G is even. If we consider singleton selectors σ only (hence switching with respect to one vertex), then it is easy to see that $\sim_G$ and $\sim_{G^\sigma}$ coincide for all selectors σ. In other words, if G has even order, then the relation $\sim_G$ is an invariant of the switching class $[G]$. This means that if $[G]$ contains an even graph, then all graphs in $[G]$ are either even or odd. Moreover, if G is even, and $\sigma : V(G) \to \mathbf{Z}_2$ is a singleton selector, then for each $v \in V(G)$, $d_G(v)$ and $d_{G^\sigma}(v)$ have different parity.

Theorem 3.18 [new]
Let G be a graph of even order and such that $[G]$ contains an even graph. Then $[G]$ contains an eulerian graph unless $[G]$ contains a complete graph.
Proof:
We may assume that G itself is even, otherwise switch at an arbitrary vertex. However, this graph may not be connected and we need to find a connected even switch of G. This can be done as follows: let u be any vertex of G. Let $V = V(G)$ and let $O = V - (\{u\} \cup N_G(u))$, the vertices that are not neighbours of u in G. Note that the set O has odd cardinality. Switch G with respect to $\sigma = O - \{v\}$ for an arbitrary $v \in O$. The resulting graph G^σ is again even, but it may still not be connected, but at least $G^\sigma|_{V-v}$ is. However, if $G^\sigma|_{V-v}$ is not the complete graph, we can choose a vertex $w \in V - \{u, v\}$ that is not connected to all vertices in $V - v$. Switching with respect to $\{v, w\}$ now gives us a connected even graph. Note that in fact $[K_n]$ for even n does have even graphs, but no eulerian graphs, because K_n is the only connected graph and it is not even, but odd. □

From the above it follows that the existence of an even graph in a switching class can be determined in time linear in the size of the graph: in the odd order case, the answer is always *yes*, while in the even order case, we need only verify whether G is an even or odd graph. If either of these is the case, then the answer is *yes*, otherwise it is *no*.

We can extend this algorithm to check for eulerian graphs: for graphs G of odd order it is simply a question of finding the unique even graph as in the proof Theorem 3.17 and verifying whether or not it is connected. For the even case we can use the algorithm implied in the proof of Theorem 3.18.

3.3 Two-graphs

In this section we will shortly address a seemingly different way of devising switching classes, two-graphs. We start with the definition of a two-graph taken from [43].

Let V be a finite set and define

$$E_3(V) = \{\{u, v, w\} \mid u, v, w \in V, u \neq v, u \neq w, v \neq w\} \ .$$

A *two-graph* is a pair (V, F), where $F \subseteq E_3(V)$ and every subset of V of cardinality 4 contains an even number of tripletons from F.

In [43] a proof can be found that there is a bijection between two-graphs on V and switching classes on V. We shall illustrate this proof by means of an example.

Example 3.19
Let G be the first graph of Figure 3.3. It is a P_4 with two isolated vertices. We shall now construct the corresponding two-graph. Because a two-graph is a single structure that should result whichever element of $[G]$ we choose as our starting point, it is not surprising that we implicitly use the invariant of Theorem 3.13. In fact, we may choose as tripletons for the two-graphs, exactly those tripletons of vertices that have an odd number of edges. In this case we obtain the two-graph

$$\mathcal{G} = (V, \{\{1,2,4\}, \{1,3,4\}, \{1,3,5\}, \{1,3,6\}, \{2,3,5\}, \{2,3,6\}, \{2,4,5\}, \{2,4,6\}\}).$$

E.g., for the set $\{1, 2, 4, 5\}$ we find that an even number, in fact two, tripletons of $\mathcal{G}$ are contained therein, which is in accordance with the definition of a two-graph.

The other direction is non-deterministic in the sense that different choices will yield different elements of the switching class. First of all, we arbitrarily choose an element $u \in V$, say $u = 3$. Then we must partition the remainder in two sets, say $V_1 = \{1, 4, 5\}$ and $V_2 = \{2, 6\}$. The former of these is the set of vertices u will be adjacent to. From Lemma 3.9 we know that everything is fixed from this point on and all that remains is to fill in the correct edges. In other words, we first choose the vertex – the neighbourhood of which we shall fix – and then we choose the neighbourhood. Note that the non-determinism lies only in the choosing of the neighbourhood.

To continue, if there is a tripleton $T = \{u, v_1, v_2\}$ where $v_1, v_2 \in V_1$ or $v_1, v_2 \in V_2$, then v_1v_2 must be an edge of the graph we are constructing, so that the triangle T has the correct parity. Contrariwise, to connect two vertices $v_1 \in V_1$ and $v_2 \in V_2$, we must have that $\{u, v_1, v_2\}$ is not an element of $\mathcal{G}$. For our choice of V_1 we obtain in this way the graph which is in the third column, sixth row of Figure 3.3. $\diamond$

Another way to code a switching class on n vertices is by a graph on $n - 1$ vertices. There is a bijection between the two: take any graph, add a single new vertex v, not connected to the other vertices and you have there the unique graph in the switching class in which v is isolated, see Corollary 3.10. This also establishes that on n vertices there are $2^{(n-1)(n-2)/2}$ switching classes and this corresponds to the fact that there are $2^{n(n-1)/2}$ graphs on n vertices divided into switching classes of cardinality 2^{n-1}, see Corollary 3.8. Note that it is essential to consider graphs up to equality here.

3.4 Some complexity considerations

A property $\mathcal{P}$ of graphs can be transformed into an existential property of switching classes as follows:

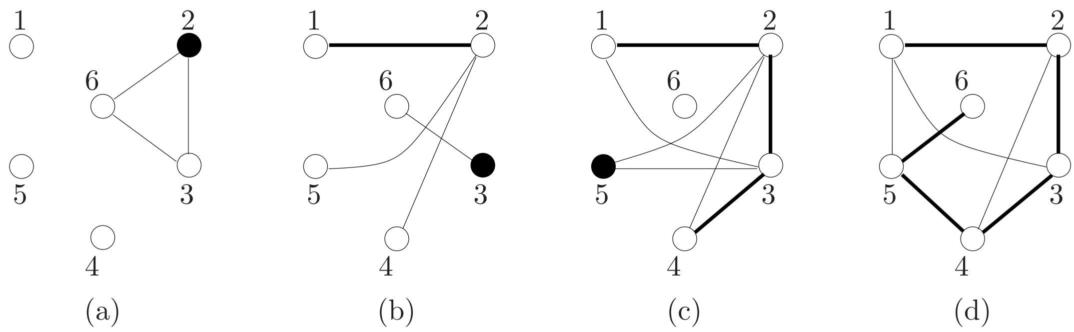


Figure 3.5: Constructing a hamiltonian path

$$\mathcal{P}_\exists(G) \text{ if and only there is a graph } H \in [G] \text{ such that } \mathcal{P}(H).$$

We will also refer to $\mathcal{P}_\exists$ as "the problem $\mathcal{P}$ for switching classes".

In this section we look at a number of (sometimes generic) instantiations of such problems, and determine their complexity. As such, we do not give a unified theory, but just a number of illustrative examples.

We generalize to switching classes a result of Yannakakis [47] on graphs (GT21, the numbering is according to [18]), which is then used to prove that the independence problem (GT20) is NP-complete for switching classes. This problem can be polynomially reduced to the embedding problem (given two graphs G and H, does there exist a graph in $[G]$ in which H can be embedded, GT48). As a result the latter problem is also NP-complete for switching classes. It also turns out that deciding whether a switching class contains a 3-colourable graph is NP-complete (a special case of GT4). We include the proof as an illustration of the proof technique.

3.4.1 Easy problems for switching classes

Let G be a graph on n vertices. Corollary 3.8 shows that there are 2^{n-1} graphs in $[G]$, and so checking whether there exists a graph $H \in [G]$ satisfying a given property $\mathcal{P}$ requires exponential time, if each graph is to be checked separately. However, although the hamiltonian cycle problem for graphs (GT37) is NP-complete, the hamiltonian cycle problem for switching classes can be solved in time $O(n^2)$, since all one needs to check is that a given graph is not complete bipartite of odd order (see Corollary 4.3).

Example 3.20
Let G be the graph of Figure 3.5(a). The objective is to find a graph in $[G]$ that has a hamiltonian path (GT39), in this case $(1, 2, 3, 4, 5, 6)$. It turns out that this is always possible, something which we shall now try to make clear. The technique we use is to change the neighbourhoods of certain vertices using Lemma 3.9 in such a way that every time it is applied we lengthen the path, without changing the already constructed path.

Since 2 is not adjacent to 1 in G we apply the selector $\{2\}$ and obtain the graph G^2 of Figure 3.5(b). After applying the selector $\{3\}$ to G^2 we add both $\{2, 3\}$ and $\{3, 4\}$ to the path and obtain the graph of Figure 3.5(c) with path $(1, 2, 3, 4)$. The last selector we have to apply is $\{5\}$ to obtain the graph H of Figure 3.5(d), which indeed has a hamiltonian path $(1, 2, 3, 4, 5, 6)$. $\diamond$

Using the same method we can obtain a spanning tree with maximum degree $\leq k$ (IND1) and a spanning tree with at least k leaves (for $2 \leq k \leq n-1$) (IND2). Note that these existence problems for graphs are all NP-complete, but like the hamiltonian path problem, easy for switching classes.

Given any property of graphs decidable in polynomial time, we can derive an algorithm for deciding the property for switching classes.

Theorem 3.21
Let $\mathcal{P}$ be a graph property such that $\mathcal{P}(G)$, for G of order n, can be decided in $\mathcal{O}(n^m)$ steps for an integer m. Let $\delta(G) \leq d(n)$ (for the minimum degree $\delta(G)$ of G) for all graphs G with $\mathcal{P}(G)$. Then $\mathcal{P}_\exists$ is in $\mathcal{O}(n^{d(n)+1+\max(m,2)})$.
Proof:
Let $G = (V, E)$ be the graph, $\mathcal{P}$ the property, $d(n)$ the polynomial that bounds the minimum degree of all graphs G for which $\mathcal{P}(G)$ holds, and m the constant as defined in the theorem.

The following algorithm checks $\mathcal{P}$ for all graphs such that there is a vertex v of degree at most $d(n)$. It uses Lemma 3.9.

Algorithm 3.22
```
P∃-DegreeBounded(G)
begin
    for all v ∈ V(G) do
        H = G^{N_G(v)}
        for all subsets σ of V(G) − {v} of cardinality ≤ d(n) do
            if P(H^σ) then
                return true;
            else continue;
        od;
    od;
end;
```

The complexity of the algorithm is easily determined. Let $n = |V(G)|$. The outer loop is executed n times. The first statement of the inner loop is in $\mathcal{O}(n^2)$ being the worst-case complexity for the application of a selector. The inner loop is executed $\mathcal{O}(n^{d(n)})$ times for each value of v. The condition of the "if" takes $\max(n^2, n^m)$, the n^2 being the complexity of switching with respect to σ, and together these yield the complexity as given in the theorem. □

This leads us to the following definition which defines the predicates such that we need only check a polynomial number of candidate graphs in each switching class.

A predicate on (or property of) graphs $\mathcal{P}$ is of *bounded minimum degree* k if and only if for all graphs $G = (V, E)$ such that $\mathcal{P}(G)$ holds, there exists $x \in V$ such that the degree of x in G is at most k.

Corollary 3.23
If $\mathcal{P}$ is a graph property of bounded minimum degree that is decidable in polynomial time, then $\mathcal{P}_\exists$ is decidable in polynomial time.

Example 3.24
It is well known that planarity of a graph can be checked in time linear in the number of vertices, see Hopcroft and Tarjan [32]. Because every planar graph has a vertex of degree at most 5 (Corollary 11.1(e) in [28]), there are only a polynomial number of graphs in a switching class that can possibly have the property. Because

we can enumerate these efficiently, it can be checked in polynomial time whether $[G]$ contains a planar graph.

Also, every acyclic graph has a vertex of degree at most 1 and therefore there is a polynomial algorithm for the acyclicity of switching classes, that is, whether a switching class contains an acyclic graph.

The discrete graph has only vertices of degree zero, so the constant in this case is 0. By the above we have an algorithm to decide whether a switching class contains a discrete graph. Note that in this case we can even omit the outer loop. Also note that this algorithm is equivalent to a check for a complete bipartite graph (Lemma 3.5) ◇

By complementation an analogous result can be formulated for graphs that always have a vertex of degree at least $d(n)$.

Corollary 3.25 [Ehrenfeucht, Hage, Harju and Rozenberg [11]]
Let $\mathcal{P}$ be a graph property such that $\mathcal{P}(G)$, for G of order n can be decided in $\mathcal{O}(n^m)$ steps for an integer m. Let $\Delta(G) \geq d(n)$ for the maximum degree $\Delta(G)$ of G. Then $\mathcal{P}_\exists$ is in $\mathcal{O}(n^{(n-d(n))+1+\max(m,2)})$.
If $d(n) = n - k$ for some constant k, then $\mathcal{P}_\exists$ is decidable in polynomial time.

3.4.2 Hard problems for switching classes

Let $\mathcal{P}$ be a property of graphs that is preserved under isomorphisms. We say that $\mathcal{P}$ is

(i) *nontrivial*, if there exists a graph G such that $\mathcal{P}(G)$ does not hold and there are arbitrarily large graphs G such that $\mathcal{P}(G)$ does hold;

(ii) *switch-nontrivial*, if $\mathcal{P}$ is nontrivial and there exists a switching class $[G]$ such that for all $H \in [G]$, $\mathcal{P}(H)$ does not hold;

(iii) *hereditary*, if $\mathcal{P}(G|_A)$ for all $A \subseteq V(G)$ whenever $\mathcal{P}(G)$.

In the following we shall look at nontrivial hereditary properties. The fact that there is a graph for which $\mathcal{P}(G)$ does not hold implies with the fact that $\mathcal{P}$ is hereditary, that there are arbitrarily large graphs for which $\mathcal{P}$ does not hold, which is why this condition was omitted from the definition of nontriviality. The fact that $\mathcal{P}$ is nontrivial directly implies that there are arbitrarily large switching classes for which $\mathcal{P}_\exists$ does hold.

Example 3.26
The following are examples of nontrivial hereditary properties of graphs that are also switch-nontrivial: G is discrete, G is complete, G is bipartite, G is complete bipartite, G is acyclic, G is planar, G has chromatic number $\chi(G) \leq k$ where k is a fixed integer, G is chordal, and G is a comparability graph. ◇

Yannakakis proved in [47] the following general result on NP-hardness and NP-completeness.

Theorem 3.27
Let $\mathcal{P}$ be a nontrivial hereditary property of graphs. Then the problem for instances (G, k) with $k \leq |V(G)|$ whether G has an induced subgraph $G|_A$ such that $|A| \geq k$ and $\mathcal{P}(G|_A)$, is NP-hard. Moreover, if $\mathcal{P}$ is in NP, then the corresponding problem is NP-complete.

Example 3.28
If we take $\mathcal{P}$ to be the discreteness property, then Theorem 3.27 says that given (G, k) the problem to decide whether G has a discrete induced subgraph of order at least k is NP-complete. Note that this problem is exactly one of the standard NP-complete problems: the independence problem (GT20). In this way the result of Yannakakis proves in one sweep that this problem is NP-complete, without having to resort to reduction. $\diamond$

We shall establish a corresponding result for switching classes. For this, let $\mathcal{P}$ be a switch-nontrivial hereditary property. The property $\mathcal{P}_\exists$ is nontrivial, and $\mathcal{P}_\exists$ is hereditary, since

$$(G|_A)^\sigma = G^\sigma|_A \tag{3.3}$$

for all $A \subseteq V(G)$ and $\sigma : V(G) \to \mathbf{Z}_2$.

Theorem 3.29
Let $\mathcal{P}$ be a switch-nontrivial hereditary property. Then the following problem for instances (G, k) with $k \leq |V(G)|$, is NP-hard: does the switching class $[G]$ contain a graph H that has an induced subgraph $H|_A$ with $|A| \geq k$ and $\mathcal{P}(H|_A)$? If $\mathcal{P} \in$ NP then the corresponding problem is NP-complete.
Proof:
Since $\mathcal{P}_\exists$ is a nontrivial hereditary property, we have by Theorem 3.27 that the problem for instances (G, k) whether G contains an induced subgraph of order at least k satisfying $\mathcal{P}_\exists$, is NP-hard. This problem is equivalent to the problem stated in the theorem, since by (3.3), for all subsets $A \subseteq V(G)$, $\mathcal{P}_\exists(G|_A)$ if and only if there exists a selector σ such that $\mathcal{P}((G|_A)^\sigma)$.

If $\mathcal{P}$ is in NP, then we can guess a selector σ and check whether $\mathcal{P}(G^\sigma)$ holds in nondeterministic polynomial time. Hence, $\mathcal{P}_\exists$ is NP-complete and so the problem is NP-complete by (the second part of) Theorem 3.27. $\square$

We shall now investigate the problem of 3-colourability, which we shall show to be NP-complete for switching classes. It is a nice illustration of proving NP-completeness of problems for switching classes by reducing the corresponding problem for graphs to the problem for switching classes.

For a given graph $G = (V, E)$ a function $\alpha : V \to C$ for some set C is a (proper) *colouring* of G if for all $uv \in E$, $\alpha(u) \neq \alpha(v)$. The *chromatic number* of G is the minimum cardinality over the ranges of possible proper colourings of G and it is denoted by $\chi(G)$. The fact that the sets $\alpha^{-1}(c)$, for all $c \in C$, are independent in G follows directly from the definition. Note that a graph is bipartite if and only if it has chromatic number less than or equal to 2.

Lemma 3.30 [new, Hage and Harju [22]]
Let G be a graph with $\chi(G) = k$. Then for all switches G^σ of G, $k/2 \leq \chi(G^\sigma) \leq 2k$.
Proof:
To see this, let $X_1, \ldots, X_k$ be a partition of G for a colouring that uses the minimal amount of k colours. Then a selector splits each X_i up into at most two sets: $X_i \cap \sigma$ and $X_i - \sigma$. Hence $\chi(G^\sigma) \leq 2\chi(G)$. By symmetry we also have $\chi(G)/2 \leq \chi(G^\sigma)$. $\square$

As a consequence, if a switching class has a graph with chromatic number larger than 4, then it does not contain a bipartite graph. However, not every 4-colourable graph generates a switching class with a bipartite graph, the graph $K_4 \cup K_1$ being a counterexample.

Theorem 3.31 [new, Hage and Harju [22]]
If $m \leq k \leq M$, where m and M are the minimum and maximum chromatic numbers in $[G]$. For every $m \leq k \leq M$ there exists an $H \in [G]$ with $\chi(H) = k$.
Proof:
Let $\chi(G_0) = m$, and $G_{i+1} = G_i^{z_i}$ for $i = 0, 1, \ldots, k-1$ with $\chi(G_k) = M$. (Here $G_0, G_1, \ldots, G_k$ is a sequence obtained by switching with respect to one vertex of the previous graph). Now, $\chi(G_{i+1}) \leq \chi(G_i) + 1$ for all i because at most we have to change the colour for the vertex that is switched. Therefore there exists a subsequence $G_{i_0}, G_{i_1}, \ldots, G_{i_{M-m}}$ such that $\chi(G_{i_j}) = m + j$. Of course, the chromatic number can sometimes decrease, but that does not matter, because eventually it grows to M. □

Example 3.32
Every odd cycle has chromatic number 3. Even cycles have chromatic number 2.

If a graph G has a clique of cardinality k, then $\chi(G) \geq k$, since k colours are needed to colour the clique. Hence a lower bound for the chromatic number of a graph is the size of its biggest clique. However, the odd cycles show that this lower bound is not always optimal. In fact, there are triangle-free graphs that have an arbitrarily high chromatic number (Theorem 12.5 of [28]). Because of Lemma 3.30 this also holds for switching classes: there is a switching class with a triangle-free graph such that the minimum chromatic number over the graphs in the switching class is arbitrarily high.

The graph K_4 has chromatic number 4. Switching with respect to any two vertices of this graph gives $P_2 \cup P_2$, which is 2-colourable. Hence, the limits of Lemma 3.30 are obtainable.

On a different note, consider the class $[K_5]$. The graph K_5 has chromatic number 5, while the other graphs in the switching class $K_4 \cup K_1$ and $K_3 \cup K_2$ have chromatic numbers 4 and 3 respectively. This shows that the maximum difference – a factor two – between chromatic numbers is not always realized. ◇

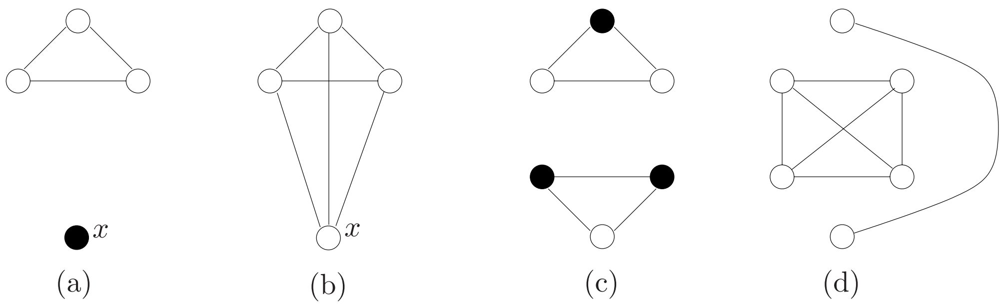


Figure 3.6:

Example 3.33
The graph 3-colourability problem (for a graph G, is $\chi(G) \leq 3$?) is NP-complete. We now consider the problem for switching classes: does there exist a 3-colourable graph in the switching class $[G]$? Obviously, this problem is in NP: we can guess the selector σ and the 3-colouring at once and verify that G^σ has the proper colouring.

Let $G = (V, E)$ be any graph, and let $G_9 = G \cup 3 \cdot C_3$ be the graph which is a disjoint union of G and three disjoint triangles. Let A be the set of the nine vertices of the added triangles.

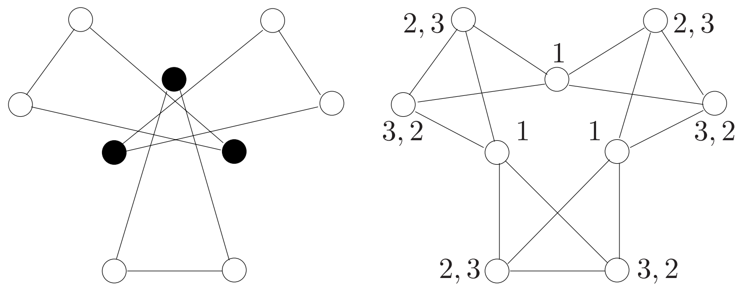
We claim that $\chi(G) \leq 3$ if and only if $[G_9]$ contains a graph H such that $\chi(H) = 3$. Since the transformation $G \mapsto G_9$ is in polynomial time, the claim follows.

It is clear that if $\chi(G) \leq 3$ then $\chi(G_9) = 3$.

Suppose then that there exists a selector σ such that $\chi(G_9^\sigma) = 3$, and let the function $\alpha : V \cup A \to \{1,2,3\}$ be a proper 3-colouring of G_9^σ.

If σ is constant on V, then G is a subgraph of G_9^σ, and, in this case, $\chi(G) \leq 3$.

Assume that σ is not constant on V. Since G_9^σ does not contain K_4 as a subgraph, it follows that σ is not constant on any of the added triangles, see Figure 3.6(a) and (b). Further, each of these triangles contains equally many selections of 1 (and of 0, of course), since otherwise the subgraph $G_9^\sigma|_A$ would contain K_4 as its subgraph, see Figure 3.6(c) and (d).



(a) (b)

Figure 3.7:

We may assume that each of the added triangles contains exactly one vertex v with $\sigma(v) = 1$ (otherwise consider the complement of σ, $V(G_9) - \sigma$), see Figure 3.7(a). Let these three vertices constitute the subset $A_1 \subset A$.

In the 3-colouring α the vertices of A_1 obtain the same colour, say $\alpha(v) = 1$ for all $v \in A_1$; and in each of the added triangles the other two vertices obtain different colours, 2 and 3, since they are adjacent to each other and to a vertex of A_1 in G_9^σ, see Figure 3.7(b).

Each $v \in V$ with $\sigma(v) = 1$ is connected to all $u \in A - A_1$; consequently, $\alpha(v) = 1$ for these vertices. Therefore the set $B_1 = \sigma^{-1}(1) \cap V$ is an independent subset of G_9^σ. Since σ is constant on B_1, B_1 is also independent in G. The vertices in $V - B_1$ (for which $\sigma(v) = 0$) are all adjacent to the vertices in A_1 in G_9^σ, and therefore these vertices are coloured by 2 or 3. The subsets $B_2 = \alpha^{-1}(2) \cap V$ and $B_3 = \alpha^{-1}(3) \cap V$ are independent in G_9^σ. Again, since σ is constant on both B_2 and B_3, these are independent subsets of G. This shows that $\chi(G) \leq 3$. ◇

3.4.3 The embedding problem

We consider now the embedding problem for switching classes. Recall that a graph H can be embedded into a graph G, denoted by $H \hookrightarrow G$, if H is isomorphic to a subgraph M of G, that is, there exists an injective function $\psi : V(H) \to V(G)$ such that

$$\{\psi(u)\psi(v) \mid uv \in E(H)\} \subseteq E(M)$$

We write $H \hookrightarrow [G]$, if $H \hookrightarrow G^\sigma$ for some selector σ. The embedding problem for graphs is known to be NP-complete, and below we show that it remains NP-complete for switching classes.

For a subset $A \subseteq V(G)$ and a selector $\sigma : V(G) \to \mathbf{Z}_2$ we have by (3.3) that $[G|_A] = [G]|_A$, where

$$[G]|_A = \{G^\sigma|_A \mid \sigma : V(G) \to \mathbf{Z}_2\}$$

is called the *subclass* of G induced by A.

Hence the switching class $[G]$ contains a graph H which has an independent subset A if and only if the induced subgraph $G|_A$ generates the switching class $[\overline{K}_A]$. As stated in Lemma 3.5 $[\overline{K}_A]$ equals the set of all complete bipartite graphs on A.

An instance of the independence problem consists of a graph G and an integer $k \leq |V(G)|$, and we ask whether there exists a graph $H \in [G]$ containing an independent set A with k or more vertices. This problem is NP-complete for graphs (that is, the problem whether a graph G contains an independent subset of cardinality $\geq k$) and, by Theorem 3.29, it remains NP-complete for switching classes.

Theorem 3.34
The independence problem is NP-complete for switching classes. In particular, the problem whether a switching class $[G]$ has a subclass $[\overline{K}_m]$ with $m \geq k$, is NP-complete.

Recall that a graph $G = (V, E)$ has *clique size* $\geq k$ if there is a set $A \subseteq V$ such that all edges exist between (different) vertices of A and $|A| \geq k$. By Lemma 3.15 and Theorem 3.34 the following corollary holds.

Corollary 3.35
For an instance (G, k), where G is a graph and k an integer such that $k \leq |V(G)|$, the problem whether $[G]$ contains a graph with clique size $\geq k$, is NP-complete.

From the simple observation that if K_n embeds into a graph G, then it is isomorphic to a subgraph of G, we obtain

Corollary 3.36
The embedding problem, $H \hookrightarrow [G]$, for switching classes is NP-complete for the instances (H, G) of graphs.

Since we can instantiate G with the clique on V and then use it to solve the clique problem of Corollary 3.35 using the same value for k, we can conclude the following.

Corollary 3.37
For an instance (G, H, k) for graphs G and H on the same domain V of cardinality n and k an integer with $3 \leq k \leq n - 1$, the problem whether there is a set $A \subseteq V$ with $|A| \geq k$ such that $H|_A \in [G|_A]$ is NP-complete.

Example 3.38
To illustrate the problem posed in Corollary 3.37, let $G = K_V$, where $V = \{u, v, w, z\}$ and let H be the graph of Figure 2.4.

First of all H cannot be embedded into any graph in $[G]$, since the parity of the edges in each tripleton of G is 1 and the parity of, for instance, $\{u, v, z\}$ is 0. So if we take $k = 4$, then the answer to the problem posed in Corollary 3.37 is “no” for G and H. For $k = 3$ however the answer is “yes”, since both graphs contain the triangle $\{u, v, w\}$. Hence the maximum subgraph match between G and H has size 3. ◇

We write $[H] \hookrightarrow [G]$, if for all $H' \in [H]$ there exists $G' \in [G]$ such that $H' \hookrightarrow G'$. By instantiating H with a clique of cardinality k, the property $[H] \hookrightarrow [G]$ becomes the clique problem of Corollary 3.35. Hence,

Corollary 3.39
For instances (H, G) of graphs the switching class embedding problem $[H] \hookrightarrow [G]$ is NP-hard.

Note, however, that the problem to decide whether a given graph H is an induced subgraph of a graph in $[G]$ is easy. In this case the only difficulty lies in checking whether G can be switched so that H appears within the switch (see Chapter 7). It occurs as a special case of the embedding problem treated here, the case where the embedding is the identity function on $V(H)$. Hence the NP-completeness arises from the number of possible injections. For Corollary 3.37 the problem also does not lie in the number of injections, because here it is also the identity: the NP-completeness arises from the number of possible subsets A. In this regard the embedding problem as treated above and the problem considered in Corollary 3.37 are orthogonal generalizations of the membership problem of Chapter 7 (when restricted to graphs).

Chapter 4

Cyclicity Considerations

Cycles are important in switching classes. One of the basic results in the field of switching classes is that the parity of edges along any cyclic sequence of vertices does not change under the application of a selector (Theorem 3.14).

In this chapter we consider a variety of problems that involve cycles. In Section 4.1 we characterize the switching classes that contain a pancyclic graph. These results are originally from Ehrenfeucht, Hage, Harju and Rozenberg [13]. The characterization leads to an efficient algorithm for detecting whether a switching class contains a pancyclic graph. As a corollary we obtain a similar result for hamiltonian graphs. Note that both these problems are NP-complete for graphs.

In Section 4.2 we prove that a switching class contains only one tree up to isomorphism and a strictly limited number of acyclic graphs. The original material is from Hage and Harju [24]. Following up our investigation into acyclic graphs we characterize those switching classes that contain an acyclic graph by a set of forbidden induced subgraphs in Section 4.3. This material is based on Hage and Harju [25].

4.1 Pancyclicity in switching classes

Let G be a graph of order n. Recall that G is pancyclic if it has a cycle of length k for all $3 \leq k \leq n$. As such it generalizes the definition of hamiltonian graph.

We prove, following the main lines of J. Kratochvíl, J. Nešetřil, and O. Zýka [35] as communicated to us by J. Kratochvíl [34], that a switching class has a hamiltonian graph if and only if n is even or the class is different from the switching class $[\overline{K}_V]$ of all complete bipartite graphs on V. This can be checked in time quadratic in the number of vertices (see Example 3.24) which should be contrasted with the fact that checking hamiltonicity for graphs is NP-complete. We actually prove a stronger result, which states that all switching classes different from $[\overline{K}_V]$ contain a pancyclic graph. This result is in accordance with Bondy's metaconjecture in [4] which declares that almost all nontrivial general graph properties that imply hamiltonicity imply also pancyclicity. In our result there is only one (trivial) exception: the switching classes of the complete bipartite graphs of even orders contain hamiltonian graphs but do not contain any pancyclic graphs. The NP-completeness of pancyclicity was only recently established by Li, Corneil and Mendelsohn [36].

The *closure* of a graph G is defined inductively as the graph G_k obtained from a sequence of graphs $G = G_0, G_1, \ldots, G_k$, where $G_{i+1} = G_i + u_i v_i$, $d_{G_i}(u_i) + d_{G_i}(v_i) \geq n$ with $u_i v_i \notin E(G_i)$, and $d_{G_k}(u) + d_{G_k}(v) < n$ for all $uv \notin E(G_k)$, see [5].

The first case of the following lemma is due to Dirac [10], the second to Bondy [4], and the third to Bondy and Chvátal [5].

Lemma 4.1
Let G be a graph and let n be the order of G.

i. If $d_G(v) \geq n/2$ for all $v \in V(G)$, then G is hamiltonian.

ii. If G is hamiltonian and $|E(G)| \geq n^2/4$, then G is pancyclic or $G = K_{n/2,n/2}$.

iii. G is hamiltonian if and only if $G+uv$ is hamiltonian, whenever $d_G(u)+d_G(v) \geq n$ for $uv \notin E(G)$. Hence, G is hamiltonian if and only if the closure of G is hamiltonian.

Theorem 4.2
If $n \geq 3$, then $[G]$ contains a pancyclic graph if and only if $[G] \neq [\overline{K}_V]$.
Proof:
Let $G = (V, E)$ be a maximum graph in its switching class. From this we find that for all $\sigma \subseteq V$, there are at least

$$\frac{1}{2}|\sigma|(n - |\sigma|) \qquad (4.1)$$

edges leaving σ, for, otherwise, switching with respect to σ would yield a graph of greater size.

If n is even, then G is hamiltonian by Lemma 4.1.i, because by (4.1), $d_G(v) \geq n/2$ for all $v \in V$. In this case, the graph has at least $n^2/4$ edges and so by Lemma 4.1.ii, we have that G is either pancyclic or $K_{n/2,n/2}$.

Suppose then that n is odd, and let $A_G = \{v \mid d_G(v) = (n-1)/2\}$. If $A_G = \emptyset$, then as above we conclude that G is pancyclic or it is complete bipartite. Assume thus that $A_G \neq \emptyset$.
Claim 1 A_G is independent in G.
Indeed, let $B \subseteq A_G$ be a clique of G. For each $v \in B$, there are exactly $(n-1)/2 - (|B|-1)$ edges that leave B, and hence by (4.1),

$$|B|(\frac{1}{2}(n-1) - (|B|-1)) \geq \frac{1}{2}|B|(n-|B|)$$

which is possible if and only if $|B| = 1$.
Claim 2: Every switching class contains a maximum graph G such that $|A_G| \leq (n-1)/2$ or $G = K_{(n-1)/2,(n+1)/2}$.
Indeed, since for $v \in A_G$, $d_G(v) = (n-1)/2$, it follows that $|A_G| \leq (n+1)/2$, because A_G is independent in G. If $|A_G| = (n+1)/2$, then let $v \in A_G$, and switch with respect to $\sigma = \{v\}$. We get a maximum graph G^σ with $|A_{G^\sigma}| \geq 1$, since $d_{G^\sigma}(v) = (n-1)/2$. By the above, we know that $|A_{G^\sigma}| \leq (n+1)/2$.

We show that if $|A_{G^\sigma}| = (n+1)/2$, then $G = K_{(n+1)/2,(n-1)/2}$. If $G|_{V-A_G}$ contains an edge, then so does $G^\sigma|_{A_{G^\sigma}}$, because $A_G \cap A_{G^\sigma} = \{v\}$ and $|V - A_{G^\sigma}| = |V - A_G| = (n-1)/2$. Consequently $A_{G^\sigma} = \{v\} \cup (V - A_G)$, but, by Claim 1, the latter is independent. So $V - A_G$ is independent in G and hence $G = K_{(n-1)/2,(n+1)/2}$.

Assume then that G is a maximum graph in its switching class such that $|A_G| \leq (n-1)/2$, and thus that G is not complete bipartite. We prove that G is hamiltonian. Because $|V - A_G| > (n-1)/2$, for each $v \in A_G$ there exists a $u \in V - A_G$ such that $vu \notin E(G)$, and

$$d_G(v) + d_G(u) \geq (n-1)/2 + (n+1)/2 = n.$$

Now $d_{G+uv}(v)$ equals $(n+1)/2$ and by Lemma 4.1.iii, G is hamiltonian since its closure is the complete graph K_n.

Knowing that G is hamiltonian, we can prove that it is, in fact, pancyclic:

$$2|E(G)| = \sum_{v \in V} d_G(v) \geq |A_G|\frac{1}{2}(n-1) + (n - |A_G|)\frac{1}{2}(n+1) = \frac{1}{2}n(n+1) - |A_G|$$
$$\geq \frac{1}{2}n(n+1) - \frac{1}{2}(n-1) = \frac{1}{2}(n^2+1)$$

and thus $|E(G)| \geq (n^2+1)/4$. By Lemma 4.1.ii, we conclude that G is pancyclic. □

Note that Claim 2 in the above proof is necessary. The crown of Figure 2.8(a) is an example of a graph that has maximum size among the graphs in its switching class, but which is not hamiltonian. One of its switches, the graph of Figure 2.8(b) does have a hamiltonian cycle and is also a maximum graph in its switching class, see Example 3.16.

Since for even n, $K_{n,n}$ is hamiltonian, we have proved

Corollary 4.3
A switching class $[G]$ contains a hamiltonian graph if and only if G is not a complete bipartite graph of odd order.

Corollary 4.4
Let G be a graph of order n. Then either G is a complete bipartite graph or for each $i = 3, \ldots, n$, there is a cycle C_i (of $K_{V(G)}$) on which the parity of edges of G is the same as the parity of i.
Proof:
Clearly, we may suppose that the order n of G is at least three. Suppose that G is not complete bipartite, and thus that $G \notin [\overline{K}_{V(G)}]$. By Theorem 4.2, there exists a pancyclic graph $H \in [G]$, and thus H has a subgraph C_i for each $3 \leq i \leq n$. By Theorem 3.14, the parity of edges of G and H on C_i is the same, which proves the claim. □

When the above corollary is applied to the complement graph of G we obtain

Corollary 4.5
Let G be a graph that is not a disjoint union of two cliques. Then for each $i = 3, \ldots, |V(G)|$, there is a cycle C_i (of $K_{V(G)}$) such that G has an even number of edges in C_i.

4.2 Counting acyclic graphs in switching classes

In this section we first prove Theorem 4.7 which says that every switching class contains at most one tree up to isomorphism. In the process we characterize the types of trees for which the switching class contains more than one tree up to equality and give the selectors that map these trees into isomorphic copies.

We then proceed along the same lines with disconnected acyclic graphs, i.e., acyclic graphs that have at least two components. A switching class can contain more than one nonisomorphic disconnected acyclic graphs, but this happens only for one special type of disconnected acyclic graph as shown in Section 4.2.4.

The connection between trees and switching classes has been considered from another point of view by Cameron [8].

In order to show that each switching class $[G]$ contains at most one tree up to isomorphism, we only need to show that for each tree T the trees in the switching class $[T]$ are isomorphic to T.

Clearly, not all switching classes contain trees. For example, if G contains a complete graph of five vertices as a subgraph then $[G]$ has no trees (in fact, no triangle-free graphs).

4.2.1 Preparation

We begin with a simple example. For this let us consider the graph G of Figure 4.1(a). Both of the (isomorphic) paths T_1 and T_2 in Figure 4.1(b) and (c) belong to $[G]$. Here the black vertices indicate the elements of σ_1 and σ_2 for which $G^{\sigma_1} = T_1$ and $G^{\sigma_2} = T_2$, respectively.

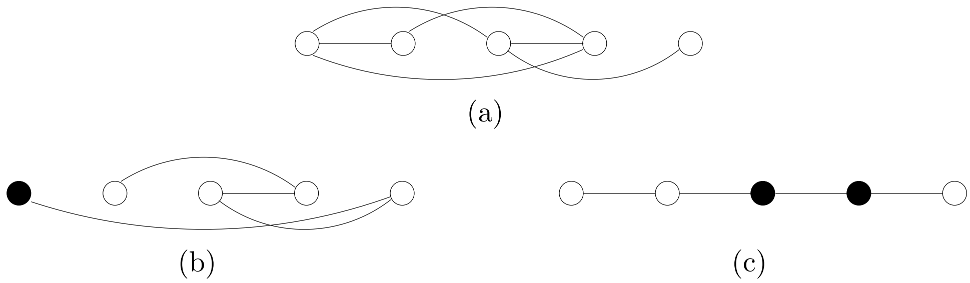


Figure 4.1: Two trees in a switching class $[G]$

In this section we use a number of types of graphs:

- $P_t(m, k)$ is the tree that is obtained from the path P_t of t vertices when the leaves are substituted by $K_{1,m}$ and $K_{1,k}$, see Figure 4.2(b) for $P_2(m, k)$ and Figure 4.2(d) for $P_4(m, k)$.
- $K^*_{1,m}$ denotes the tree, where the leaves of $K_{1,m}$ are substituted by edges P_2, see Figure 4.2(a).
- $K_{1,3}(m, k)$ denotes the tree, where two of the leaves of $K_{1,3}$ are substituted by the stars $K_{1,m}$ and $K_{1,k}$, see Figure 4.2(c).

Note that $K_{1,m} = P_2(0, m-1)$ for all stars $K_{1,m}$ with $m \geq 1$.

We begin with a general result on acyclic graphs.

Lemma 4.6
Let $F = (V, E)$ and F^σ be acyclic graphs for a selector $\sigma \subseteq V$.

i. If C is a connected component of $F|_\sigma$, then C consists of at most two vertices. Moreover, if $C = \{x, y\}$ with $x \neq y$, then $xy \in E$, and

$$\forall z \in V - \sigma, \text{ either } zx \in E \text{ or } zy \in E \text{ but not both } . \tag{4.2}$$

ii. Either $F|_\sigma$ or $F|_{V-\sigma}$ is discrete.

iii. For any $x, y \in \sigma$ with $x \neq y$ there exists at most one $z \in V - \sigma$ such that $zx, zy \in E$.

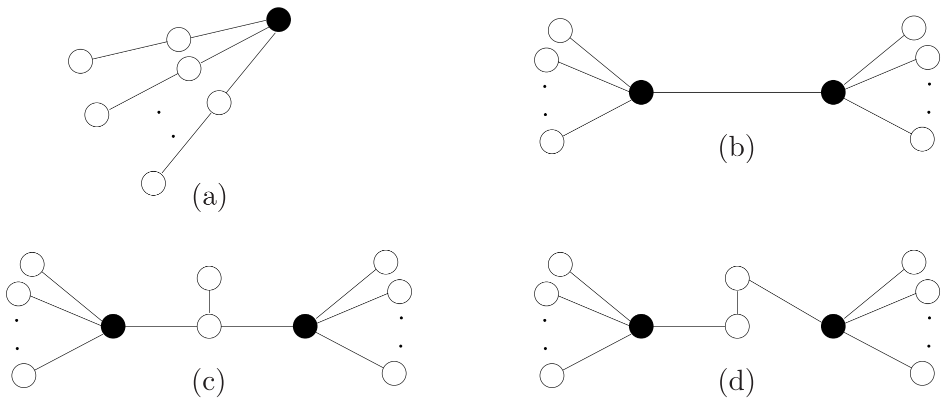


Figure 4.2: Types of trees having self isomorphic switches

Proof:
Clearly, $F^\sigma|_\sigma$ and $F|_\sigma$ have the same connected components. Let $F|_C, C \subseteq \sigma$ be a connected component of $F|_\sigma$. By acyclicity, for each $z \in V - \sigma$ there can be at most one edge zx of F and similarly of F^σ such that $x \in C$. On the other hand, each zx with $x \in C$ is an edge either in F or in F^σ. This shows the first claim. The second claim follows immediately from this: if xy is an edge in $F|_\sigma$ and wx an edge in $F|_{V-\sigma}$, then by the first claim z is connected to either x or y and the same holds for w, which gives a cycle.

The third claim is clear, since if $z_1, z_2 \in V - \sigma$ with $z_i x, z_i y \in E$, then (x, z_1, y, z_2) would be a cycle in F. □

4.2.2 Trees

We continue now with the first main result of this section, proving that a switching class contains only isomorphic trees. We also list the trees that have an isomorphic switch in their switching class including the corresponding switches. In addition to the graphs of Figure 4.2 we also find two exceptional trees on seven vertices, P_7 and T_7 of Figure 4.3 respectively.

Figure 4.3: The only two 3-by-4-bipartite trees on seven vertices yielding a tree

Theorem 4.7
Every switching class contains at most one tree up to isomorphism. If it contains more than one tree up to equality, then the tree is one of $K^*_{1,m+1}$, $P_2(m, k+1)$, $K_{1,3}(m, k)$, $P_4(m, k)$ for $m, k \geq 0$ (Figure 4.2), or one of the two special trees P_7 or T_7 of Figure 4.3.
Proof:
Suppose that $T = (V, E)$ is a tree for which there exists a selector $\sigma \subseteq V$ such that T^σ is also a tree. We may suppose that σ is not constant on V, since otherwise

$T^\sigma = T$. Furthermore, by Lemma 4.6(ii), we may assume that $T|_{V-\sigma}$ is discrete, since $T^\sigma = T^{V-\sigma}$.

Let $n = |V|$, $p = |\sigma|$ and suppose $T|_\sigma$ contains r edges, x_iy_i for $i = 1, 2, \ldots, r$ with $x_i, y_i \in \sigma$, where $\{x_i, y_i\}$ are the nonsingleton connected components of $T|_\sigma$. Since T is a tree, it has $n-1$ edges, and so there are $n-1-r$ edges of T in $\sigma\times(V-\sigma)$. Also, T^σ has $n-1$ edges, and there are $p(n-p)-(n-1-r)$ edges of T^σ in $\sigma\times(V-\sigma)$. Therefore, the number of edges of T^σ is $n-1 = p(n-p) - (n-1-r) + r$, that is,

$$(p-2)n = (p-2)(p+1) + (p-2r)\ . \tag{4.3}$$

If $p = 1$, then $r = 0$, and $n = 1$, which is a trivial case.

If $p = 2$, then $p = 2r$ and hence $r = 1$, and in this case $\sigma = \{x_1, y_1\}$ with $x_1y_1 \in E$, and, by (4.2), $V - \sigma = B_1 \cup B_2$, where $B_1 = \{z \in V - \sigma \mid zx_1 \in E\}$ and $B_2 = \{z \in V - \sigma \mid zy_1 \in E\}$ form a partition of $V - \sigma$. Therefore T is a $P_2(m, k)$ with $m \geq 0$ and $k \geq 1$ of Figure 4.2(b), where the black vertices are in σ. Here T^σ is also a $P_2(m, k)$, and thus isomorphic to T.

Assume then that $p > 2$. Now equation (4.3) becomes

$$n = p + 1 + \frac{p-2r}{p-2}\ . \tag{4.4}$$

It is immediate that either $2r = p$, or $r = 1$, or $r = 0$. These cases give us the following solutions.

If $2r = p$, then $n = p + 1$. Now, $T|_\sigma$ consists of r edges and it has no singleton connected components, and $T|_{V-\sigma}$ is a singleton graph. Therefore T is a $K^*_{1,r}$ (with $r \geq 1$) of Figure 4.2(a), where the black vertex is in $V - \sigma$. Clearly, also in this case T^σ is isomorphic to T.

If $r = 1$, then $n = p + 2$, and thus $T|_\sigma$ has one edge x_1y_1 and $p - 2$ isolated vertices, and $T|_{V-\sigma}$ is a discrete graph of two vertices, say z_1, z_2. By (4.2), there are now two choices: z_1 and z_2 are connected to the same or different vertices of $\{x_1, y_1\}$. From these we obtain that T is either $K_{1,3}(m, k)$ or $P_4(m, k)$ with $m, k \geq 0$ of Figure 4.2(c) and 4.2(d), respectively. Again, as is easy to see, T^σ is isomorphic to T in both of these cases.

If $r = 0$, then $p = 3$ or $p = 4$. In this case $n = 7$, and there are eleven nonisomorphic trees on seven vertices, see Harary [28]. Of these trees seven are 3-by-4-bipartite; they are listed in Figure 4.4 and Figure 4.3.

Now, if T and T^σ are both trees, then none of the vertices should be connected to all vertices in the other part of the partition, because such vertices would become isolated; this excludes Figure 4.4(b), (c), (d) and (e). Also T contains no independent set with two vertices in σ and two vertices in $V - \sigma$, because that would give a C_4; now also (a) is excluded. We are then left with only two trees T of seven vertices, which are T_7 and P_7 of Figure 4.3. For both of these trees T^σ is isomorphic to T. □

4.2.3 Trees into disconnected acyclic graphs

Let $k, m \geq 0$. For the following theorem we will find the following graphs to be exceptional cases:

- $S_{k,m}$ denotes the tree, which is obtained from a star $K_{1,k+m}$ by substituting k leaves by an edge, see Figure 4.5(a) for the graph and its disconnected acyclic switch.

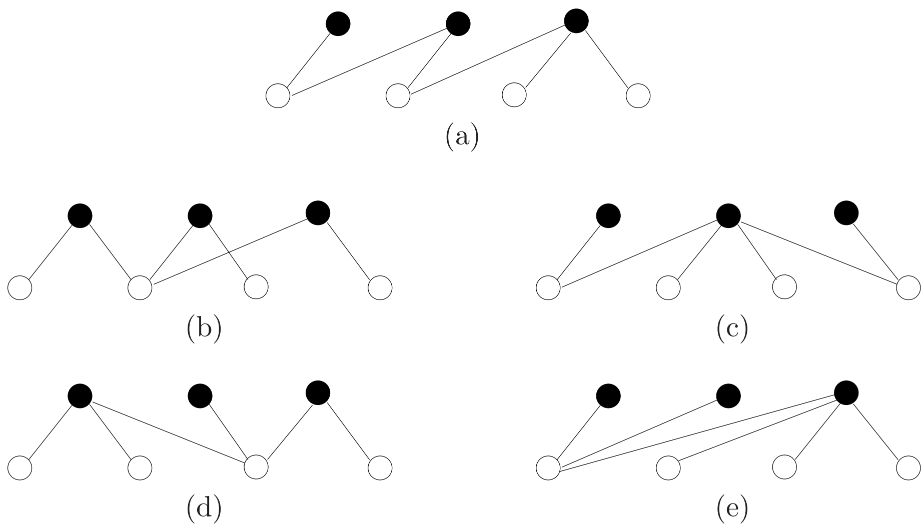


Figure 4.4: The other 3-by-4-bipartite trees on seven vertices

- $P_3(k, m)$ which is a special case of $P_t(k, m)$ defined earlier, see the left graph of Figure 4.5(b).
- P_6, the left graph of Figure 4.5(c), which is simply the path on six vertices.

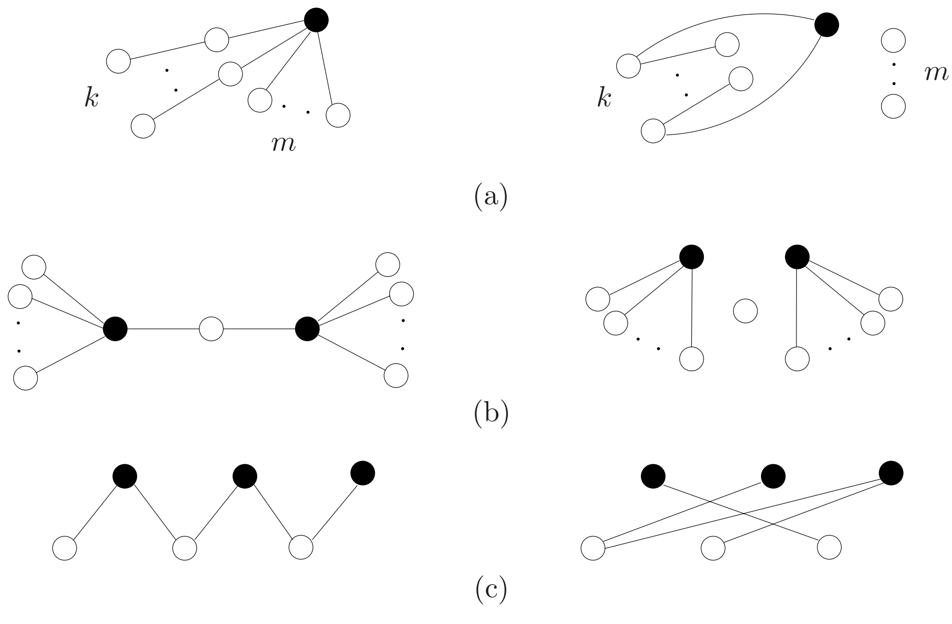


Figure 4.5: Trees and their disconnected switches

We consider now the case where a tree T produces a disconnected acyclic graph T^σ. Again we have indicated the switches to obtain, in this case, a disconnected acyclic graph by the black vertices.

Theorem 4.8
Let $T = (V, E)$ be a tree such that T^σ is a disconnected acyclic graph. Then T is $P_3(m, k)$, $S_{k,m+1}$, or P_6, for some $m, k \geq 0$.

Proof:
Suppose that $T = (V, E)$ is a tree such that T^σ is a disconnected acyclic graph. Obviously, σ is a nonconstant switch. As above we may assume that $T|_{V-\sigma}$ is discrete. Let again $n = |V|$, $p = |\sigma|$ and suppose $T|_\sigma$ contains r edges. Now, T^σ has less than $n-1$ edges, and (4.3) is transformed into

$$(p-2)n < (p-2)(p+1) + (p-2r) \ . \tag{4.5}$$

If $p = 1$, then clearly $T = K_{1,n-1} = S_{0,n-1}$, and hence T^σ is the discrete graph.

If $p = 2$, say $\sigma = \{x, y\}$, then (4.5) becomes $2(1-r) = p - 2r > 0$, and therefore $r = 0$. Since T is connected there exists a vertex $z \in V - \sigma$ such that $zx, zy \in E$, and by Lemma 4.6, the vertex z is unique. Consequently, $V - \sigma = N(x) \cup N(y)$ with $N(x) \cap N(y) = \{z\}$ for the sets $N(x)$ and $N(y)$ of neighbours of x and y. Hence T is a $P_3(m, k)$, where the middle vertex of the P_3 is z, see Figure 4.5(b) and σ consists of the two black vertices. In this case T^σ is a disconnected acyclic graph, where z is isolated, and the edges are xu and yv for all $u \in N(y)$ and $v \in N(x)$.

If $p > 2$, then (4.5) gives

$$n < p + 1 + \frac{p-2r}{p-2} \ ,$$

which is possible only if $r < p/2$; if $r = p/2$ then we would have $p \geq n$, which cannot happen. Assume first that $p = n-1$, i.e., $|V - \sigma| = 1$. This case holds always if $r > 0$. The corresponding tree is $T = S_{k,m}$ with $m > 0$, see Figure 4.5(a), where the black vertex is in $V - \sigma$. Note that this also includes the star graph, namely when $k = 0$.

If $r = 0$ we get

$$n < p + 1 + \frac{p}{p-2} \ ,$$

in which case $p = 3$ yields $n < 7$ and $p = 4$ also yields $n < 7$. Larger values for p yield $n = p + 1$, a case we have already treated.

If $r = 0$, then $T|_\sigma$ is discrete, so the case $p = n + 2$ reduces to the case $p = 2$.

The only remaining cases are trees with $n = 6$ and $p = 3$. These are P_6 (see Figure 4.5(c)), $S_{2,1}$ and $P_2(2, 2)$. If we switch the latter according to the bipartition we obtain a cyclic graph, because it has an independent set of four vertices, two in σ and two in $V - \sigma$. The other two, P_6 and $S_{2,1}$, switch into a disconnected acyclic graph.

□

Note that P_6 is also a $P_4(1, 1)$, but we have two different acyclic switches. Also $S_{1,m} = P_3(0, m)$ for $m \geq 1$.

4.2.4 Disconnected acyclic graphs

In this section we prove a result analogous to the result about trees: every switching class contains at most one disconnected acyclic graph up to isomorphism excepting one special kind of disconnected acyclic graph.

The counterexample is the disconnected acyclic graph $S_{k,m,\ell}$ which is formed by adding ℓ isolated vertices to $S_{k,m}$ of Figure 4.5(a) (see Figure 4.6(a)). Of course, for $S_{k,m,\ell}$ to be a disconnected acyclic graph it is necessary that $\ell > 0$.

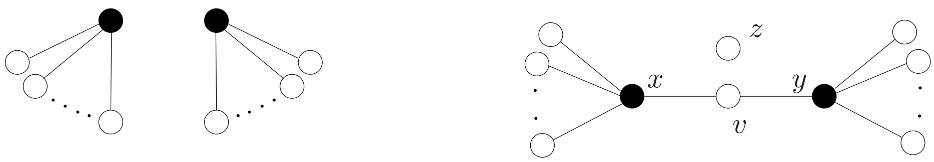
If $S = S_{k,m,\ell}$ and we take σ to be the one black vertex in the figure, then $S^\sigma = S_{k,\ell,m}$ is an acyclic graph of the same type (see Figure 4.6(b)), but S^σ is isomorphic to S if and only if $m = \ell$.

(a) (b)

Figure 4.6: The graphs $S_{k,m,\ell}$ and $S_{k,\ell.m}$

Two other types of graphs are listed in Figure 4.7. These give the disconnected acyclic graphs that switch into isomorphic graphs.

- $K_{1,k} \cup K_{1,m}$ is simply the disjoint union of two stars (with k and m rays respectively) and it is depicted in Figure 4.7(a), and
- $P_3(k,m) \cup K_1$ can be obtained by taking the leftmost graph of Figure 4.5(b) and adding an isolated vertex.



(a) (b)

Figure 4.7: Two cases of self isomorphism

Theorem 4.9
Every switching class contains at most one disconnected acyclic graph up to isomorphism, unless it is a class containing $S_{k,m,\ell}$ with $m \neq \ell$ and $m, \ell > 0$. If it contains more than one disconnected acyclic graph up to equality, then they are $K_{1,k} \cup K_{1,m}$, $P_3(k,m) \cup K_1$ or $S_{k,m,\ell}$ with $m, \ell > 0$.
Proof:
Let $F = (V,E)$ be a disconnected acyclic graph, n the order of F, $\sigma \subseteq V$ and $p = |\sigma| > 0$, and assume that F^σ is a disconnected acyclic graph. We can suppose that $p \leq n/2$, since $F^\sigma = F^{V-\sigma}$. We prove that F^σ is isomorphic to F, unless $F = S_{k,m.\ell}$ with $m \neq \ell$.

We prove that $p = 1$ or $p = 2$.

By Lemma 4.6, we know that either $F|_\sigma$ or $F|_{V-\sigma}$ or both are discrete, and that the connected components of $F|_\sigma$ and $F|_{V-\sigma}$ are either singletons or single edges.

Because F and F^σ both have at least two components and no cycles, together they contain at most $2(n-2)$ edges. Therefore, $p(n-p) \leq 2(n-2)$, since there are $p(n-p)$ edges in $K_{\sigma,V-\sigma}$ and all of these are in either F or F^σ (but not both). We find that $n(p-2) \leq p^2 - 4 = (p-2)(p+2)$. Now, either $p = 1$ or $p = 2$, since if $n \leq p+2$, then $n \leq 4$, because we can assume that $p \leq n/2$.

We finish by considering the two cases $p = 1$ and $p = 2$.

Consider first the case $p = 1$, and let $\sigma = \{x\}$. If $|V - \sigma| = 1$, then $n = 2$, and hence $F = S_{0,0,1}$ is discrete, and $F^\sigma = S_{0,1,0}$ is a tree.

Assume then that $|V - \sigma| \geq 2$. By Lemma 4.6(i), all components of $F|_{V-\sigma}$ are either singletons or edges. In the case of an edge yz, x is connected to exactly one of y and z (otherwise we have a triangle in either F or F^σ). As a consequence, $F = S_{k,m,\ell}$ for some $k, m \geq 0$ and $\ell \geq 1$, and, consequently, $F^\sigma = S_{k,\ell,m}$. Hence, in this case, F^σ is a tree if and only if $m = 0$, and otherwise F^σ is a disconnected acyclic graph. In the latter case, F^σ is isomorphic to F if and only if $m = \ell$.

The case that $p = 2$ can be treated as follows. Let $\sigma = \{x, y\}$.

In this case $K_{\sigma,V-\sigma}$ contains exactly $2(n-2)$ edges, while F and F^σ both contain at most $n-2$ edges. This implies that F and F^σ contain exactly $n-2$ edges and all these edges are between σ and $V - \sigma$. Hence F and F^σ are disjoint unions of trees. Note that all edges go between σ and $V - \sigma$, and $F|_{V-\sigma}$ is discrete.

Suppose first that x and y belong to different connected components. Now F becomes decomposed into two stars, the leaves of which are the neighbours of x and y, respectively. In this case, F and F^σ are both disjoint unions $K_{1,r} \cup K_{1,s}$ where $r + s = n - 2$, see Figure 4.7(a).

On the other hand, if x and y belong to the same connected component, then, by Lemma 4.6, there exists a unique vertex $v \in V - \sigma$ such that $vx \in E$ and $vy \in E$. Since F is disconnected, there is a vertex $z \in V - \sigma$ such that $xz, yz \notin E$. Furthermore, because F^σ is acyclic and xz, yz are edges of F^σ, this vertex z must, like v, be unique. This implies that F and F^σ are isomorphic, the isomorphism is the permutation $(x, y)(v, z)$, which leaves all other vertices intact, see Figure 4.7(b).

This completes the proof of the theorem. □

The class $[S_{1,0,m}]$ for $m > 0$ contains both $S_{1,m,0}$ and $S_{0,m,2}$. The latter because $S_{1,0,m} = S_{0,2,m}$. For $m \geq 3$, the switching class $[S_{1,0,m}]$ contains three acyclic graphs up to isomorphism: $S_{1,0,m} = S_{0,2,m}$, $S_{0,m,2}$ and $S_{1,m,0}$.

From the above we obtain also the following corollary.

Corollary 4.10
Every switching class contains

i. at most two disconnected acyclic graphs up to isomorphism and the upper bound is reached if and only if it contains $S_{k,m,\ell}$ with $m \neq \ell$ and $m, \ell > 0$.

ii. at most three acyclic graphs up to isomorphism. The upper bound is optimal and can only be reached if it contains two disconnected acyclic graphs up to isomorphism.

Proof:
The first claim follows from Theorem 4.9 and the fact that although a graph $S_{k,m,\ell}$ may have more than one switch into a disconnected acyclic graph, either the switches are isomorphic, or the original graph is isomorphic to one of the switches. Of course, if $m = \ell$, then $S_{k,m,\ell}$ is isomorphic to $S_{k,\ell,m}$. For disconnectedness of both $S_{k,m,\ell}$ and $S_{k,\ell,m}$ we need that $m, \ell > 0$.

For the second part, the switching class $[S_{1,0,m}]$ is an example that reaches the upper bound. If we would have four acyclic switches, then either we have at least two trees up to isomorphism, which is forbidden by Theorem 4.7, or we have three disconnected acyclic graphs, which contradicts the first claim. □

4.3 Characterizing acyclic switching classes

In this section we solve a problem raised by Acharya [2] and by Zaslavsky in his dynamic survey in 1999 [51], which asks for a characterization of those graphs that have an acyclic switch.

The graphs that do not have an acyclic switch are called *forbidden*. Obviously, if a forbidden graph occurs in another graph, then the latter is also forbidden. For this reason we are interested in the graphs that are minimal in this respect: they do not have an acyclic switch, but all their proper induced subgraphs do have an acyclic switch. We call these graphs and the corresponding switching class *critically cyclic*. A switch of a critically cyclic graph is also critically cyclic so the latter notion is well-defined for switching classes.

Forbidden graphs for perfect graphs in switching classes were treated by Hertz [30].

We show that apart from the simple cycles C_n for $n \geq 7$, there are only finitely many critically cyclic graphs. In fact, we shall prove that a critically cyclic graph $G \notin [C_n]$ has order at most 9. These graphs are partitioned into 24 switching classes, and altogether there are 905 critically cyclic graphs of order at most 9 (up to isomorphism and excluding switches of the cycles C_n).[3]

In order to save the reader from long – and occassionally tedious – technical constructions for the small graphs, we rely on a computer program (in fact, two independent ones as explained in Appendix B) for the cases of order at most 9. Therefore our purpose is to prove that if G is a critically cyclic graph of order $n \geq 10$, then $G \in [C_n]$. The proof of this result uses the characterization from Section 4.2 of the acyclic graphs G – henceforth called the *special graphs* – that have a non-trivial acyclic switch.

After reintroducing the special graphs from Section 4.2 we proceed with our actual results proving that critically cyclic graphs can have only a limited number of isolated vertices and as a consequence, a vertex in a critically cyclic graph has only a limited number of leaves adjacent to it. We prove that each critically cyclic switching class, except $[C_n]$ for $n \geq 8$, contains a (critically cyclic) graph, which is, except for two vertices, a special graph. By verifying that for each type of special graph a contradiction results – under the condition that the order of the graph is at least 10 – we finally prove our result. At the end we consider the question why not all of the critically cyclic switching classes are used in our proof.

Let G be a critically cyclic graph. By definition, for all $x \in V$, there is a switch G^σ such that $G^\sigma - x$ is acyclic. As a consequence, all cycles in G^σ go through x and there is at least one such cycle. Note that this also holds for $(G^\sigma)^x$. Note that it does not hold that in every critically cyclic graph G there is a vertex x so that $G - x$ is acyclic; the graph $K_{3,3} \cup 3 \cdot K_1$ of Figure 4.11(9-2) is a counterexample.

Example 4.11
Let G be the graph of Figure 4.8(a) (see also Figure 4.13(7-3')). We want to prove that it is a critically cyclic graph. For this we must show that it has no acyclic switches and removing any of the vertices allows for an acyclic switch. For the latter it is sufficient to observe that the vertices $2, \cdots, 6$ are all on the only cycle of G, and $G^{\{2,5\}} - 7$ and $G^{\{3,6\}} - 1$ are acyclic, see Figure 4.8(b) and Figure 4.8(c) respectively.

[3] Contact the author for the ps-file containing all of them or visit the Technical Reports section of the LIACS website `http://www.liacs.nl/`.

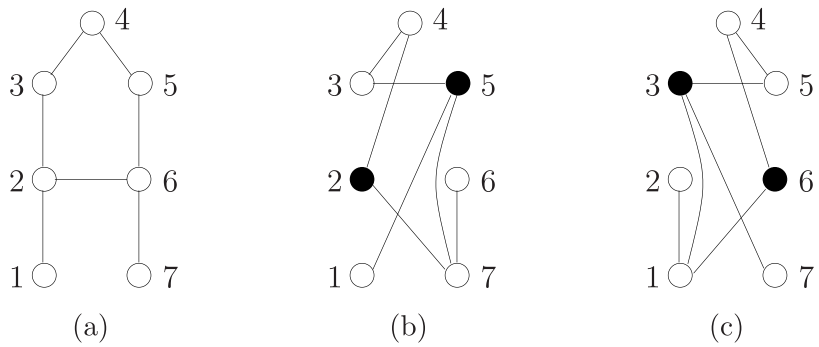


Figure 4.8: The graph G and two of its switches

To prove that G has no acyclic switch observe that G has seven edges and an acyclic graph can have at most six. We shall now prove that applying any selector will not decrease the number of edges, and thereby we have proved that there is no acyclic switch.

First of all, the degree of every vertex in G is at most $3 = (n-1)/2$. Hence applying a singleton selector cannot decrease the number of edges.

For doubleton selectors, $\sigma = \{x_1, x_2\}$, we can do the same: the number of edges that changes is $|\sigma| \cdot (7 - |\sigma|) = 10$. We must make sure then that every selector makes at most five edges disappear. The only possible way, knowing that the maximum degree is three, is to take $\sigma = \{2, 6\}$, but in that case only four edges are removed, because one edge occurs in $G|_\sigma$.

For selectors of size 3, finally, twelve edges will change. Hence we must look for selectors that create less than six edges (or, in other words, make more than six edges disappear). For this, the selector must contain a vertex of degree three, say $\{2\}$. If we would also have $6 \in \sigma$, then the number of edges to be removed is four and there are no other vertices of degree three. Adding two vertices of degree two to σ results always in a selector having at most six edges going to its complement, because always either the two of them are adjacent, or one of them is adjacent to vertex 2.

Because of the symmetry in the graph, the same holds if we start with $6 \in \sigma$. $\diamond$

Note that C_n for $n \leq 6$ have an acyclic switch: take an independent set of cardinality $\lfloor n/2 \rfloor$. However, as was already proved by Acharya [2]

Lemma 4.12
The cycles C_n for $n \geq 7$ are critically cyclic.
Proof:
First of all, removing any vertex gives us an acyclic graph P_{n-1} and hence we have to prove that all switches of $C_n, n \geq 7$, have a cycle.

Let $\{x_1, \ldots, x_n\}$ be the vertices of C_n in order around the cycle. We first treat the selectors that select the same value, say 1, in two adjacent vertices, say x_1 and x_2. We need only consider nonconstant selectors and without loss of generality we may assume that $\sigma(x_n) = 0$. Now $\sigma(x_3) = 0$, because otherwise G^σ has a triangle $\{x_n, x_2, x_3\}$. Then $\sigma(x_4) = 1$, because otherwise $\{x_1, x_2, x_4\}$ is a triangle. The same holds for x_5 and now we have a triangle, $\{x_n, x_4, x_5\}$ in G^σ, since $n \geq 7$ implies that x_5 is not a neighbour of x_n. This takes care of all C_n, where $n \geq 7$ is odd.

The only case left is the selector σ that selects exactly the odd numbered vertices of C_n. It is easy to verify that C_8^σ is isomorphic to itself, and if $n \geq 10$ and even, then $\{x_1, x_2, x_4, x_5, x_7, x_8\}$ induces a C_6 in C_n^σ. □

We now state the result of our computer search for the critically cyclic graphs (see Appendix B for information on the programs used).

Theorem 4.13
There are 27 switching classes of critically cyclic graphs of order $n \leq 9$. Representatives of these are given in the Figures 4.9, 4.10 and 4.11.

The main theorem proved in this section is the following.

Theorem 4.14
The switching classes $[C_n]$ are the only critically cyclic switching classes of order $n \geq 10$.

In our proofs we shall refer to the graphs from Figure 4.9, 4.10, 4.11 and 4.13. The black vertices in the latter figure indicate how these graphs can be switched into the corresponding unprimed graphs from the former three figures. We shall use Theorem 4.13 to the extent that they are in fact critically cyclic graphs. The proof does not rely on the computer result that these are in fact *all* of them of order at most 9.

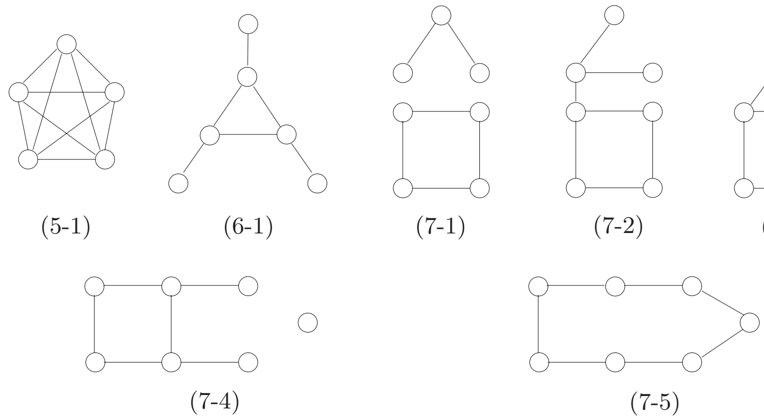


Figure 4.9: The critically cyclic graphs on five, six and seven vertices

4.3.1 The special graphs

We reintroduce here (many of) the special graphs of Section 4.2 (see Figure 4.12). We shall use these graphs often in our proofs. Recall that these graphs have in common that they can be switched into an acyclic graph by a nontrivial selector.

The graph in Figure 4.12(1s) is denoted by $S_{k,m,\ell}$. It is a graph $K_{1,k+m}$ where k of the $k+m$ leaves are substituted by an edge, and to which ℓ isolated vertices have been added. We let, see also Figure 4.12(1s),

(S1) z be the centre of S,

(S2) $H = \{z, y_i, x_i \mid i = 1, 2, \ldots, k\}$ be the vertices of the extended star of S rooted at z,

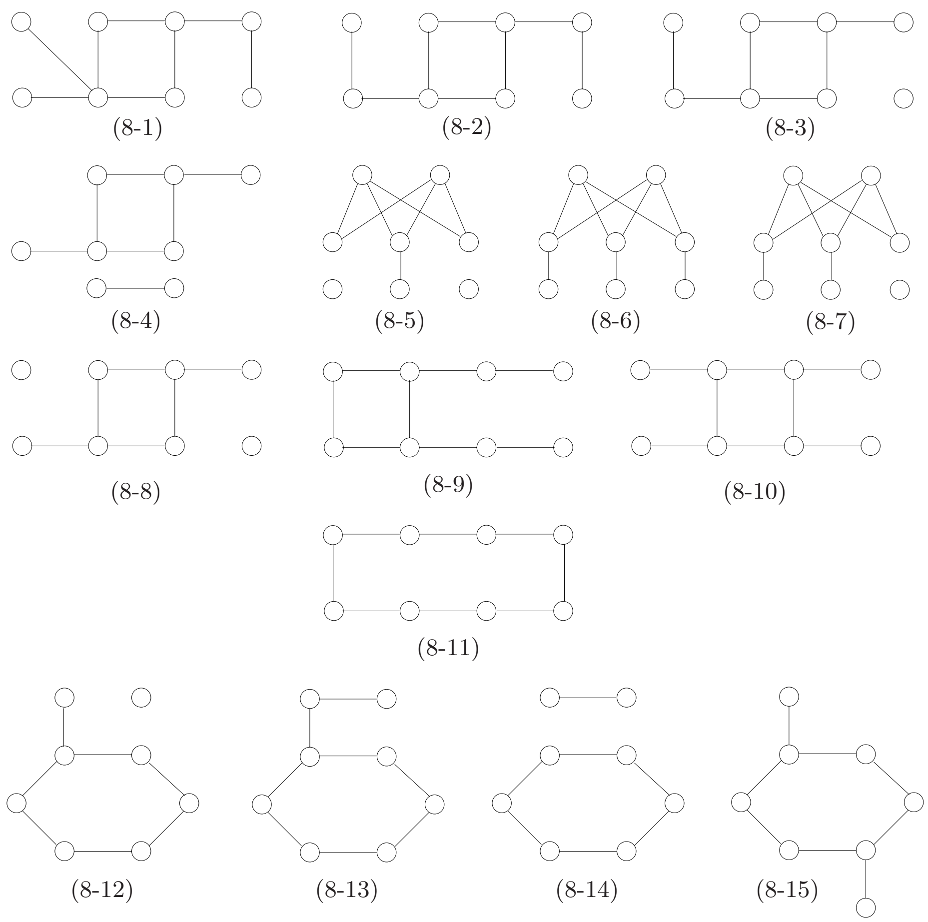


Figure 4.10: Critically cyclic graphs on eight vertices

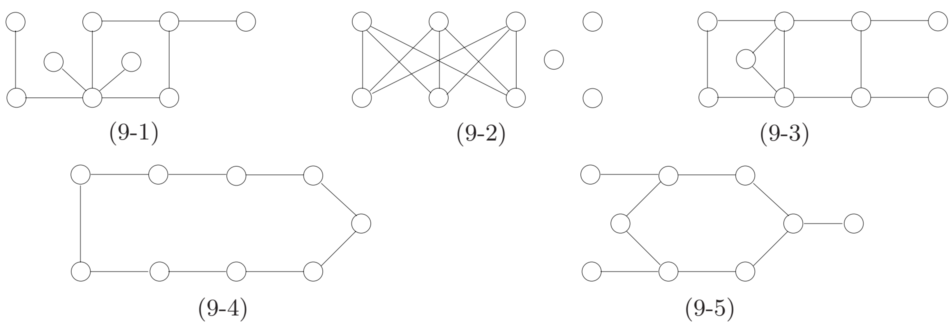


Figure 4.11: Critically cyclic graphs on nine vertices

(S3) $I = \{u_1, u_2, \ldots, u_\ell\}$ be the set of isolated vertices of S, and

(S4) $M = \{v_1, v_2, \ldots, v_m\}$ be the set of leaves adjacent to z in S.

The types (2s)-(8s), see Figure 4.12, of graphs are denoted by $S(k, m)$, where k and m indicate the number of leaves of the (black) vertices z_1 and z_2. Because of the symmetry in k and m in each of these graphs we may assume that $k \geq m$.

A graph of type (2s) is simply the disjoint union $K_{1,k} \cup K_{1,m}$. Adding an isolated vertex to a graph of this type gives a graph of type (3s).

We denote by $P_t(m, k)$ the tree that is obtained from the path P_t of t vertices when the leaves are substituted by $K_{1,m}$ and $K_{1,k}$, see Figure 4.12(4s) for $P_3(k, m)$ (adding an isolated vertex gives a graph of type (5s)), Figure 4.12(6s) for $P_2(k, m)$ and Figure 4.12(8s) for $P_4(k, m)$. Furthermore, $K_{1,3}(k, m)$ denotes the tree, where two of the leaves of $K_{1,3}$ are substituted by the stars $K_{1,k}$ and $K_{1,m}$, see Figure 4.12(7s).

The acyclic graphs P_7, T_7, P_6 and $P_4 \cup P_2$ are listed in Figure 4.12(9s), (10s), (11s) and (12s) respectively. Their role is strictly limited, because of their low order. Notice that P_6 equals $P_4(1, 1)$ of the type (8s), but we wish to treat this small instance independently.

We reformulate the results from Section 4.2 that shall be used in this section.

Theorem 4.15

i. Every switching class contains at most one tree up to isomorphism. The trees that have a nonconstant switch into a tree are fully characterized by (6s)-(10s), and (1s) for $m, \ell = 0$.

ii. Every switching class contains at most three acyclic graphs up to isomorphism. The acyclic graphs that have a nonconstant acyclic switch are fully characterized by (1s)-(12s) (the switches are indicated by the black vertices).

The graphs of all except a few of the types, switch into an isomorphic copy of themselves if we apply the selector indicated by the black vertices, the *centres* of the special graphs. There are five exceptions: a graph $S_{k,m,\ell}$ of type (1s) switches into $S_{k,\ell,m}$ and these are only isomorphic if $m = \ell$, and a graph of type (3s) switches into a graph of type (4s) (and vice versa). Finally, the graphs (11s) and (12s) switch into each other.

In the following we shall often want to use the fact that a certain special graph has a unique nontrivial switching into an acylic graph. For instance, the graph $S_{1,2,0}$ is of type (1s), but also (4s), (6s) and (7s). These give rise to a number of "extra" selectors that map $S_{1,2,0}$ into an acyclic graph. In this case the extra selectors are $\{x_1, z\}$, $\{y_1, z\}$, and $\{y_1, v_1\}$ respectively.

We want to avoid situations such as these in our proofs and as it will turn out, it will not bother us. However, to be precise, we shall list conditions on each of the types, that guarantee that the acyclic switch is unique. Please note also that it is not simply a question of overlap between two different types, but a graph such as $S_{1,0,2}$ overlaps with itself: there are two choices for the vertex z. Hence $S_{1,0,2}$ is the same graph as $S_{0,2,2}$.

Lemma 4.16
A special graph $S_{k,m,\ell}$ has a unique nonconstant switch into an acyclic graph if $k \geq 3$, or $k = 2$ and $m + \ell \geq 2$, or $k \leq 1$ and $m, \ell \geq 3 - k$.

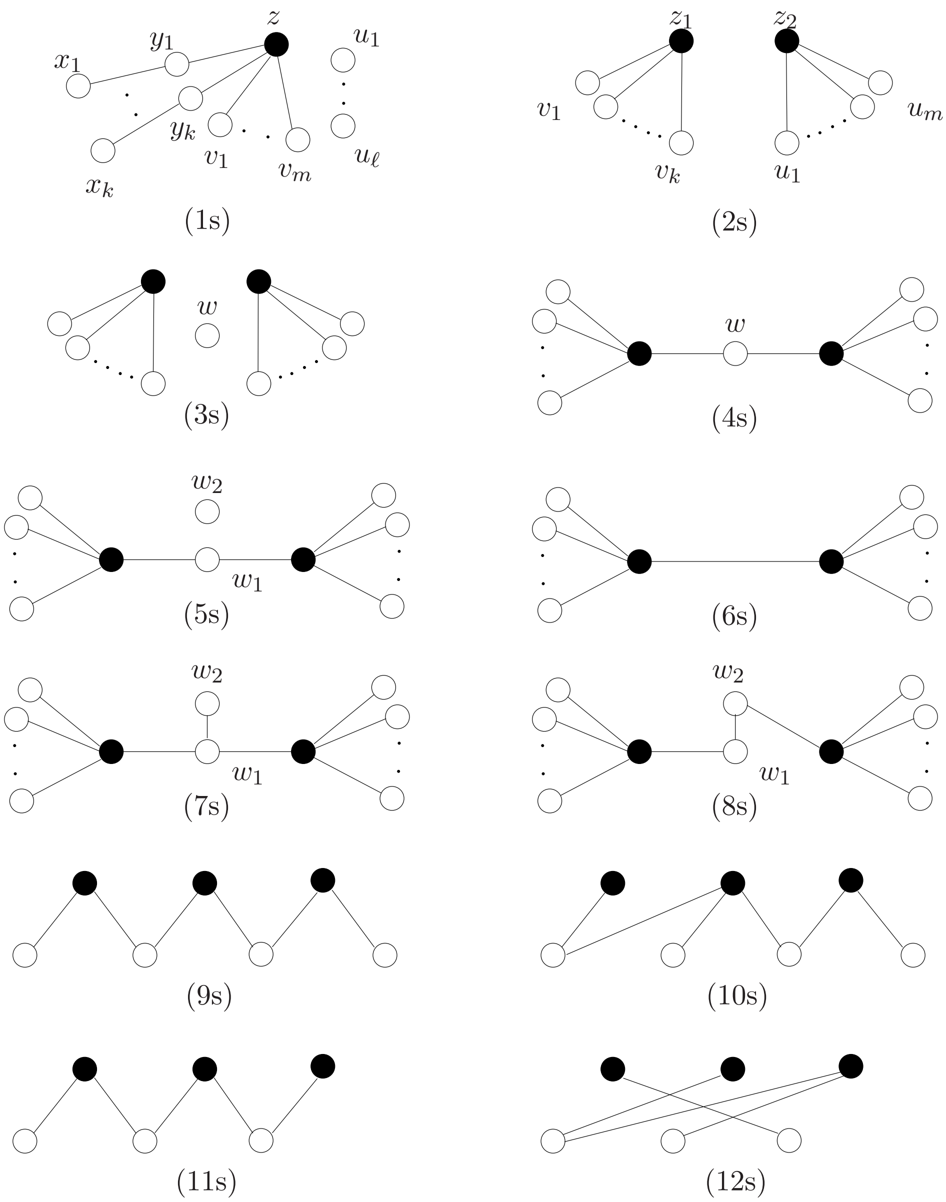


Figure 4.12: The special graphs (1s)-(12s)

Proof:
We remark first that the graphs $3 \cdot P_2$ and $2 \cdot P_2 \cup 2 \cdot K_1$ have no nonconstant switch to an acyclic graph. This follows from Theorem 4.15 and the fact that they are not special.

Let σ be a nonconstant selector containing z. Suppose σ is not the switch $\{z\}$. If $k \geq 2$, then we have $k = 2$ and $|I \cup M| \leq 1$ by the previous remark. In these cases the choice for z is unique: it is the middle vertex of the induced P_5's.

Let $k = 1$ and $m, \ell \geq 2$. Now, $S = S_{k,m,\ell}$ has three components and hence the only possibility for overlap is with (1s) and (3s). The fact that $k = 1$ and $m \geq 2$ exclude the possibility that the nontrivial component of $S_{k,m,\ell}$ is a star, so (3s) is now taken care of. If $k = 1$ and $m = 2$ the choice for z is unique, it is the only vertex of degree at least 3.

For $k = 0$ and $m, \ell \geq 3$, $S_{k,m,\ell}$ is exclusively of type (1s), because no other type of special graph has more than three components. Also, if $m \geq 3$, then the choice for z in type (1s) is unique. □

The types (9s) and (10s) are obviously unique. The graph (11s) is also of type (8s), and hence has two nonconstant acyclic switches. The same holds for the graph (12s). For the other types (2s)-(8s) we now list the conditions.

Lemma 4.17
Under the following conditions do the special graphs $S(k, m)$, $k \geq m$ have a unique nonconstant switch to an acyclic graph.

- (2s)-(4s) need $k, m \geq 2$,
- (5s), (7s), (8s) need $k \geq 2, m \geq 1$,
- (6s) needs $k, m \geq 3$.

Proof:
Let $S = S(k, m)$ be a special graph and let $\{z_1, z_2\} \subseteq \sigma$ with σ nonconstant. We prove that $\sigma = \{z_1, z_2\}$ if S^σ is to be an acyclic graph.

First of all, $K_{1,2} \cup K_{1,2}$ has one nonconstant acyclic switch (either select the leaves, or select the two inner vertices). For all types, except (6s), it now follows that $k, m \geq 2$ implies the existence of a unique nonconstant switch to an acyclic graph.

For (2s)-(4s) this is all we can do. For (2s), $K_{1,2} \cup K_{1,1}$ has two switches: the choice of z_2 in $K_{1,1}$ is arbitrary. Something similar holds for (3s). Additionally, note that (4s) for $k = 2$ and $m = 1$ is also (8s) with $k = 2$ and $m = 0$ and the switches are different.

In the cases (5s), (7s) and (8s) we do have a unique switch for $k \geq 2, m \geq 1$, because the vertex z_2 can only be chosen in one way: it is the vertex in $K_{1,1}$ that is not a leaf in S.

In the case of (6s) we get $k, m \geq 2$, because of overlap with (1s). Because (6s) for $k \geq m, m = 2$ overlaps with (7s) we arrive at the condition $k, m \geq 3$. □

Note, that there are cases that do overlap, but in which case the switches happen to be equivalent: (5s)($k = 0 = m$) and (2s)($k = 2, m = 0$) are the same graph, but the corresponding switches are complements.

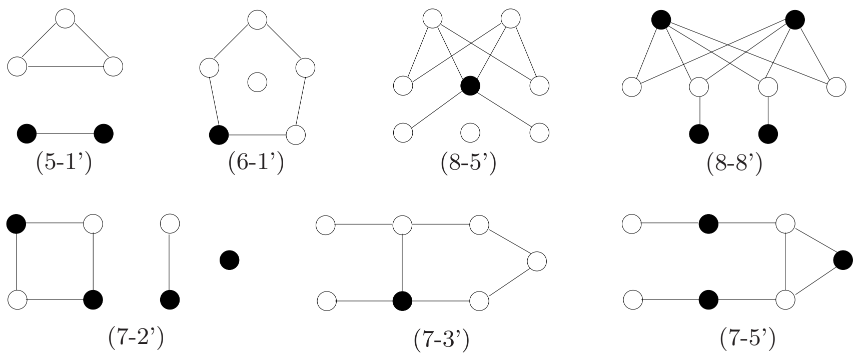


Figure 4.13: Switches of known critically cyclic graphs that are used in the proofs

4.3.2 Isolated vertices

In this section we give constraints for the isolated vertices in critically cyclic graphs. In particular, we prove our main tool for the final proof: if G is critically cyclic such that $G - x$ is acyclic for a vertex x, then $G - x$ has no isolated vertices.

Lemma 4.18
Let G be a critically cyclic graph. Then G has at most two isolated vertices or $G = K_{3,3} \cup 3 \cdot K_1$ ((9-2) in Figure 4.11).
Proof:
Let $I = \{x_1, x_2, \ldots, x_m\}$ be the set of isolated vertices of G, and assume that $m \geq 3$. Now $G - x_1$ is not acyclic, and it has an acyclic switch $(G - x_1)^\tau$. Hence τ is not constant on $G - I$, say $\tau(v_0) = 0$ and $\tau(v_1) = 1$ for some $v_0, v_1 \notin I$.

If two vertices of $I - \{x_1\}$ have the same value for τ, say $\tau(x_2) = i = \tau(x_3)$, then v_{1-i} is the unique vertex of $V - I$ with $\tau(v_{1-i}) = 1 - i$. Indeed, if it were $\tau(v) = 1 - i$ for another $v \in V - I$, then (x_2, v_{1-i}, x_3, v) would form a cycle in $(G - x_1)^\tau$. Moreover, in this case, there exists a vertex of I, say x_4, such that $\tau(x_4) = 1 - i$, for, otherwise, extending τ by setting $\tau(x_1) = i$ would result x_1 to be a leaf of G^τ contradicting the fact that all cycles of G^τ go through x_1. However, now (x_2, v_{1-i}, x_3, x_4) forms a cycle in $(G - x_1)^\tau$, which is a contradiction. In particular, $m \leq 3$ to avoid triangles with x_2 or x_3. The switching class of the discrete graph $\overline{K}_n$ consists of the complete bipartite graphs of order n by Lemma 3.5 and therefore $m = 3$, and $\tau(x_2) \neq \tau(x_3)$. Since the graph $(G - x_1)^\tau$ is acyclic and $G^\tau(x_2x_3) = 1$, it follows that $V - I$ is independent in $(G - x_1)^\tau$. Therefore $G = K_{r,s} \cup 3 \cdot K_1$ for some $r, s \geq 2$. Since $K_{3,3} \cup 3 \cdot K_1$ is a critically cyclic graph, and each $K_{2,s} \cup 3 \cdot K_1$, for $s \geq 4$, has an acyclic switch (by switching one of the vertices in the part of size 2 of $K_{2,s}$), the claim follows. □

Lemma 4.19
Let G be critically cyclic of order $n \geq 10$. Then no vertex $z \in V$ is adjacent to more than two leaves of G.
Proof:
If a set L of leaves of G is adjacent to a vertex z, then the vertices of L become isolated in G^z. The result follows easily from Lemma 4.18. □

Lemma 4.20
Let G be a critically cyclic graph of order $n \geq 10$. Then G has at most one isolated vertex.

Proof:
Suppose that G has exactly two isolated vertices, $I = \{x_1, x_2\}$. Let $(G - x_1)^\tau$ be acyclic, where we assume that $\tau(x_2) = 0$ without restriction. The set τ is independent in G and in $(G - x_1)^\tau$, for, otherwise, there would be a triangle (containing x_2) in $(G - x_1)^\tau$. In fact, τ contains at most one vertex from each connected component of $(G - I)^\tau$. Notice that these connected components are trees, because $(G - x_1)^\tau$ is acyclic.

Let $\tau = \{z_1, \ldots, z_r\}$, and set $\tau(x_1) = 0$. Then

$$G^\tau = (H + (T_1 \cup T_2 \cup \ldots \cup T_r)) \cup F,$$

where $H = K_{2,r}$ has the bipartition $(\{x_1, x_2\}, \{z_1, \ldots, z_r\})$, and the induced subgraphs T_i are disjoint trees with $H \cap T_i = \{z_i\}$; and F is an acyclic induced subgraph or it is empty. Since G^τ is not acyclic, we must have $r \geq 2$.

- By (7-1) and (7-2'), either F is discrete or it is a path P_2. In both cases, $|F| \leq 2$, by Lemma 4.18.
- By (8-6), there can be at most two nontrivial trees among $T_1, \ldots, T_r$.
- Let T_i be nontrivial a tree. By (7-1) the height of T_i is at most 3 and there are no vertices of degree more than 2 at a level higher than 2. The graph (7-2) excludes the possibility that a child of z_i has degree larger than two, and by (7-2') the tree cannot contain both an induced P_4 and an induced P_3 in case they have no edge in common. Hence each nontrivial tree T_i has the form

$$T_i = S_{k_i,s_i,0} \text{ or } P_4(s_i, 0),$$

  where $S_{k_i,s_i,0}$ (for $k_i \geq 0$) is one of the special trees with z_i as its centre, and in $P_4(s_i, 0)$, z_i is the centre adjacent to the s_i leaves. By Lemma 4.19, $s_i \leq 2$.

We shall now consider the three cases for zero, one and two nontrivial T_i.

(0) If G^τ has no nontrivial components among $T_1, \ldots, T_r$, then G^τ equals either $K_{2,r}$, $K_{2,r} \cup K_1$, $K_{2,r} \cup 2 \cdot K_1$ or $K_{2,r} \cup P_2$. All these have an acyclic switch; a contradiction.

(1) Suppose G^τ has exactly one nontrivial tree among $T_1, \ldots, T_r$, say T_1.
Let $T_1 = P_4(s_1, 0)$.

- By (7-1), $r = 2$ (otherwise remove z_1).
- By (7-2'), $|F| = 0$ (otherwise remove the vertices of T_1 adjacent to z_1).

However, now $n \leq 9$ contradicts our assumption on n.

Let $T_1 = S_{k_1,s_1,0}$ with $k_1 > 0$, and let $r \geq 3$.

- By (7-2'), $|F| = 0$, $s_1 = 0$ and $k_1 = 1$ (otherwise remove z_1).

In this case T_1 is a path P_3, and G^τ has an acyclic switch for all $r \geq 3$ (switch all z_i's and the other endpoint of T_1); a contradiction.

Then the case for $r = 2$. In this case, by (7-2'), F cannot be a path P_2, and so it is discrete. Now G^τ has an acyclic switch (switch at z_1).

Finally, if $T_1 = S_{0,s_1,0}$, then $|T_1| \leq 3$ (by Lemma 4.19), and therefore $r \geq 4$, since $|F| \leq 2$.

- By (7-2'), F is discrete (otherwise remove z_1).
- By (8-5), $|F| \leq 1$, and by (8-5'), if $|F| = 1$, then $s_1 = 1$ (and in this case, T_1 is a path P_2).

The remaining cases, $s_1 = 1$ and $|F| = 1$, and $s_1 = 2$ and $F = \emptyset$, have acyclic switches for all r (switch with respect to x_1, x_2 and a leaf at z_1); a contradiction.

(2) Suppose that G^τ has exactly two nontrivial trees in $T_1, \ldots, T_r$, say T_1 and T_2, and assume without loss of generality that $|T_1| \geq |T_2|$.

- By (8-8'), $r \leq 3$.
- By (8-4) and (8-8), $|F| \leq 1$.
- By (8-7), if $r = 3$, then $|F| = 0$.

Hence $r + |F| \leq 3$. Since $n \geq 10$, it follows that $|T_1| + |T_2| \geq 10 - r - |F| \geq 7$.

First we treat the trees of height at most 1. In this case, $T_1 = S_{0,s_1,0}$. By Lemma 4.19 $s_1 \leq 2$, and hence $|T_1| \leq 3$. Therefore $|T_2| \geq 4$, which contradicts the assumption $|T_1| \geq |T_2|$.

Let t_1 be the height of T_1 and suppose that $t_1 \geq 2$. Now by (8-1) and (8-2), $|T_2| = 2$, that is, T_2 is a path P_2, and consequently $|T_1| \geq 5$. If $T_1 = S_{k_1,s_1,0}$ ($k_1 > 0$), then $k_1 = 1$, otherwise we have (8-3) by removing a middle vertex from a P_3.

It follows that $|T_1| = t_1 + 1 + s_1 \geq 5$. Recall that $s_1 \leq 2$. However, the case $t_1 \geq 2$ and $s_1 = 2$ is excluded by (9-1), and the cases $t_1 = 3$ and $1 \leq s_1 \leq 2$ are excluded by (8-4) (remove the child of z_1 on the path of length t_1). □

As in Lemma 4.19, we have

Lemma 4.21
Let G be a critically cyclic graph of order $n \geq 10$. Then no vertex $z \in V$ is adjacent to more than one leaf of G.

Lemma 4.22
Let G be a critically cyclic graph of order $n \geq 10$ and let $x \in G$.

i. $G - x$ can have at most two isolated vertices. Moreover, if $G - x$ has two isolated vertices, then x is adjacent to exactly one of these in G.

ii. If a vertex $z \neq x$ is adjacent to m leaves of $G - x$, then $m \leq 2$. Moreover, if $m = 2$, then x is adjacent to exactly one of these.

Proof:
For (i) we only need to observe that if $G - x$ has three isolated vertices, then in either G^x or G at least two of these are isolated and we can apply Lemma 4.20. The same holds if the number of isolated vertices is two, but x is not adjacent to exactly one of them in G.

For (ii), assume that there is a vertex $z \neq x$ adjacent to more than two leaves. The vertex x is adjacent to at most one of these in either G or G^x and the result then follows from Lemma 4.21. □

We say that a vertex $y \in V$ is *compatible* with x, if

- $G - x$ is acyclic,

- $G - y$ and $G^x - y$ are not acyclic.

Note that if y is compatible with x, then all cycles in G (and G^x) go through x, but not all of them go through y.

Lemma 4.23
Let G be a critically cyclic graph such that $G - x$ is acyclic.

i. If y is compatible with x, then $G - \{x, y\}$ is a special graph.

ii. If G is of order $n \geq 8$, then there exists a vertex $y \in V$ that is compatible with x unless $G \in [C_n]$.

Proof:
Let $(G - y)^\tau$ be acyclic and set $S = G - \{x, y\}$. Because S and S^τ are both acyclic graphs it follows that either (a) S is special or (b) τ is constant on S.

In the case (b) all cycles go through x and y which contradicts the fact that $G - y$ is not acyclic. To see this, let there be a cycle that does not go through y. There are two selectors of $G - y$ constant on S. The first of these is $\tau = S \cup \{x\}$. But then $(G - y)^\tau$ equals $G - y$ which is a contradiction, because the former is acyclic and the latter is not. If on the other hand $\tau = S$, then $(G - y)^\tau = (G^x - y)^{S \cup \{x\}} = G^x - y$ and again we have a contradiction.

For the second part, suppose $G \notin [C_n]$. Since G has no acyclic switches, there are cycles in G and G^x, and they all pass through x, because $G - x$ is acyclic. Moreover, since C_k is critically cyclic for $k \geq 7$ by Lemma 4.12, the induced cycles of G and G^x have length at most 6.

If G or G^x has an induced cycle C_5 or C_6, then let y be a vertex that is not on such a cycle. It is clear that $G - y$ and $G^x - y$ both contain cycles, and therefore each such y is compatible with x.

If G and G^x have both an induced cycle of length at most 4, then these two cycles have altogether at most 7 vertices (since they share the vertex x), and, by $n \geq 8$, there exists a vertex y that is not on these cycles. For each such vertex y, both $G - y$ and $G^x - y$ are not acyclic. This proves the claim. □

Lemma 4.24
Let G be critically cyclic of order $n \geq 10$ such that $G - x$ is acyclic. Then $G - x$ has no isolated vertices.

Proof:
Assume to the contrary of the claim that u is isolated in $G - x$. In this case u is either a leaf adjacent to x (or isolated) in G and isolated (or a leaf adjacent to x) in G^x. Hence $G - u$ and $G^x - u$ are not acyclic and by Lemma 4.23(i), $S = G - \{x, u\}$ is a special graph.

In this case, by Lemma 4.22(ii) and the fact that $n \geq 10$, S must be either of type (1s) or one of (5s), (7s), (8s) with $k = 2 = m$.

In the latter three cases S has a unique acyclic switch at the two centres $\tau = \{z_1, z_2\}$ by Lemma 4.17, and it is easy to see that $(G - u)^\tau$ is not acyclic, since x is adjacent to exactly one leaf adjacent to both centres in S and remains to be so in S^τ.

Consider then the case $S = S_{k,m,\ell}$ and adopt the notations (S1)-(S4) for it. The selector τ is such that $(G - u)^\tau$ is acyclic. Without restriction we can assume that $\tau(z) = 1$. Extend τ to the whole domain by setting $\tau(u) = 0$.

We have $n = (2k + 1) + m + \ell + 2 \geq 10$, and thus $k \geq \frac{1}{2}(7 - (m + \ell))$. By Lemma 4.22, $m \leq 2$ and $\ell \leq 1$. (Recall that u is an isolated vertex of $G - x$.) In particular, $k \geq 2$, and if $k = 2$, then $m = 2$, $\ell = 1$ and $n = 10$. In these cases, by Lemma 4.16 the special acyclic graph S has a unique acyclic switch S^ρ, where $\rho = \{z\}$. By the uniqueness of ρ, we have that $\rho(v) = \tau(v)$ for all $v \notin \{x, u\}$.

Now, the only vertices in G that can become adjacent to u in G^τ are x and z and because G^τ is not acyclic, these connections must exist: $G^\tau(ux) = 1 = G^\tau(uz)$ and they are the only edges of G^τ incident with u. Moreover, x is adjacent in G^τ to exactly one vertex $v \in H \cup I$, since G^τ contains a cycle but $G^\tau - u$ does not.

Let $v = x_i$, say $v = x_1$. If $\ell \geq 1$, then $\{x, x_1, z, u, y_1, u_1, y_2\}$ induces a (7-4) in G^τ. Therefore $\ell = 0$. If $|M| \geq 1$, then $\{x, x_1, z, u, y_1, w, v_1\}$ induces a (7-4) in G^τ for $w = x_2$ or $w = y_2$ depending on the value $G^\tau(xv_1)$. Therefore also $m = 0$. Now $k \geq 4$, and G^τ contains an induced (7-4) obtained by removing x_2.

If $v = y_i$, say $v = y_1$, then $G^\tau|_{\{x,y_1,x_1,z,u\}}$ is an induced C_5, and hence G^τ has an induced (6-1') obtained by removing x_2.

Let $v = u_i$, say $v = u_1$. To avoid (8-3) as being induced by $\{x, u_1, z, u, x_1, y_1, y_2, v_i\}$ (for any $v_i \in M$), we must have $G^\tau(xv_i) = 0$ (if $m > 0$). Now, however, $(G^\tau)^z$ is acyclic.

If $v = z$, then G^τ has an acyclic switch for $\{z\}$. This contradiction completes the proof of the lemma. □

4.3.3 The cases

In this section, let G be a critically cyclic graph of order $n = |V| \geq 10$ and let $x \in V$ be a fixed vertex.

Since G is critically cyclic, there exists an acyclic switch $(G-x)^\sigma$ of the subgraph $G - x$. Because the switches of critically cyclic graphs are criticially cyclic, we can assume that σ is constant on V, and therefore that $G - x$ is acyclic already.

Assume that y is a vertex compatible with x, that is, $G - y$ and $G^x - y$ are both not acyclic. We know by Lemma 4.23(ii) that vertices such as x and y defined above exist if the switching class does not contain C_n. In the following we shall consider every type of special graph in turn and show that each case leads to a contradiction, thereby proving our main theorem, Theorem 4.14, that besides graphs in $[C_n]$ there no critically cyclic graphs of order $n \geq 10$.

By Lemma 4.23(i), $S = G - \{x, y\}$ is a special acyclic graph, and $(G - y)^\sigma$ is acyclic for a nonconstant selector σ. The special graph S cannot be of type (9s), (10s), (11s) or (12s), because the order of S should be at least 8 to ensure that $n \geq 10$.

Without restriction we can assume that $\sigma(x) = 0$. This follows from the symmetry in the definition of compatibility, i.e. the fact that both $G - y$ and $G^x - y$ are not acyclic. We extend σ to the whole domain by setting $\sigma(y) = 0$. Note that $(G - y)^\sigma = G^\sigma - y$.

In the following proofs a number of simple properties are often used, and we note them here: first of all, the vertex y is adjacent to at most one vertex of each component of S. If not, $G - x$ would not be acyclic. Also, there must be a cycle in G that does not contain y, because $G - y$ is not acyclic. This also holds for $G^x - y$.

We shall now formulate a few conditions that hold for the, still remaining, special graphs (1s)-(8s). For a graph H, let $L_H(z)$ be the set of leaves adjacent to z in H, and let I_H denote the set of isolated vertices in H.

Lemma 4.25
Given the definitions above, we have that

i. $I_S \subseteq N_G(y)$.

ii. For all $z \in S$, $|L_S(z)| \leq 3$. Moreover, $|L_S(z)| = 3$ implies $|N_G(x) \cap L_S(z)| \geq 1$ and $|N_G(y) \cap L_S(z)| = 1$.

Proof:
Claim (i) follows from Lemma 4.24.

We have $|N_G(y) \cap L_S(z)| \leq 1$, since $G - x$ is acyclic. If $|L_S(z)| \geq 3$, then, by Lemma 4.22(ii), $|L_S(z) - N_G(y)| \leq 2$, and x is adjacent to at most one vertex of $L_S(z) - N_G(y)$. Hence, in this case, we must have $|L_S(z)| = 3$ and in that case x and y are each adjacent to at least one vertex. In the case of y it is exactly one vertex. □

Note how Lemma 4.25 restricts the values of k and m for the types (2s)-(8s) and m for (1s). On the other hand $n \geq 10$ gives a lower bound on these values for most types.

4.3.4 The case (1s)

We shall now consider first the most difficult case, $S_{k,m,\ell}$. Suppose that $S = S_{k,m,\ell}$, and adopt the notations of (S1)-(S4) for it. Without restriction we may assume that $\sigma(z) = 1$.

Lemma 4.26
We have

i. $k = 2$,

ii. $1 \leq \ell, m \leq 2$ and $m + \ell \geq 3$,

iii. $M \subseteq N_G(x)$,

iv. if $\ell = 2$, then $|N_G(x) \cap I| = 1$,

v. if $m = 2$, then $|N_G(y) \cap M| = 1$,

vi. $|N_G(x) \cap (H \cup I - \{z\})| \leq 1$.

Proof:
By Lemma 4.20, $|N_G(x) \cap I| \leq 1$ for, otherwise, switching with respect to $\{x, y\}$ gives two isolated vertices. By Lemma 4.22(ii) we have both $\ell \leq 2$ and Claim (iv).

If $k = 0$, then $m + \ell \geq 7$, contradicting the bound $m \leq 3$ from Lemma 4.25(ii) and the bound $\ell \leq 2$.

If $k = 1$, then $m + \ell \geq 5$, since $n \geq 10$. In this case, $\ell = 2$ and $m = 3$. If $k = 2$, then $m + \ell \geq 3$. Therefore by Lemma 4.16, in all cases $k \geq 1$, S^z is the unique acyclic switch of S. It follows that $\sigma|_S = \{z\}$, and therefore $M \subseteq N_G(x)$, for, otherwise the acyclic graph $G^\sigma - y$ would have an isolated vertex contradicting Lemma 4.24 (remember that we have $\sigma(x) = 0 = \sigma(y)$). Lemma 4.22(ii) then implies $m \leq 2$, and as a consequence $k \geq 2$, because as was shown above, if $k = 1$, then we must have $m = 3$. Claim (v) follows from Lemma 4.22(ii).

Claim (vi) follows from the fact that $G^\sigma|_{H \cup I}$ is connected and $G^\sigma - y$ is acyclic.

Suppose then that $k \geq 3$. By Claim (vi) it follows that there are at least two pairs $x_i y_i$ such that $G(xx_i) = 0 = G(xy_i)$, say for $i = 1, 2$. Let the selectors τ_i be such that $(G - x_i)^{\tau_i}$ are acyclic, where we may choose $\tau_i(z) = 1$. The special graph $S - x_i$, which is $S_{k-1,m+1,\ell}$, has a unique acyclic switch $(S - x_i)^z$ by Lemma 4.16, since $n \geq 10$ and $k \geq 3$ (note that $\ell = 0$ implies $m \geq 4$, because $n \geq 10$).

It is then clear that $\tau_i = \sigma$ when we set $\tau_i(x_i) = 0$. By Lemma 4.24, the vertex y_i is not isolated in $G^{\tau_i} - x_i$, and therefore $G^{\tau_i}(yy_i) = G(yy_i) = 1$ for i=1,2 (since $G(xy_i) = 0 = G^{\tau_i}(xy_i)$) and we have a cycle in $G - x$. This contradiction proves Claim (i) and Claim (ii). □

Notice that Lemma 4.26 implies that $n \leq 11$.

We finish the case $S = S_{k,m,\ell}$.

Assume $G(xu_1) = 1$. Then $G(xz) = 1$, since otherwise (x, u_1, z) would be a triangle in $G^\sigma - y$. Also, $G(xx_i) = 0 = G(xy_i)$ for $i = 1, 2$, because $G^\sigma - y$ is acyclic.

We have $G(xy) = 0$, for, otherwise (x, y, u_1) is a triangle in G, and to avoid (5-1) with the edges $G(x_i y_i) = 1$, we would have to have that y is adjacent to two vertices in $H - \{z\}$ giving a cycle to $G - x$. Note that now all edges involving x are known.

Now (x, z, v_1) is a triangle in G, and to avoid (7-5'), necessarily $G(yz) = 1$ or $G(yv_1) = 1$, and y is adjacent to no other vertices of $H \cup M$.

(1) If $G(yz) = 1$, then $|M| = 1$, because otherwise y must be adjacent to either one of the v_i (Lemma 4.26(v)), but then (y, z, v_i) is a cycle of $G - x$. Lemma 4.26(ii) implies $|I| = 2$ and $\{x, u_1, y, z, y_1, x_2, u_2\}$ induces a (7-4). Remember, we have $G(xu_2) = 0$ by Lemma 4.26(iv).

(2) If $G(yv_1) = 1$, then $\{u_1, y, x, v_1, z, y_1, x_2\}$ induces a (7-3).

Therefore $G(xu_1) = 0$, and consequently $I = \{u_1\}$ by Lemma 4.26(iv).

By Lemma 4.26(ii), $m = 2$, and we have $G(xv_1) = 1 = G(xv_2)$, $G(yv_1) = 1$, $G(yv_2) = 0$, and $G(yu_1) = 1$, $G(xu_1) = 0$.

In this case $G(yw) = 0$ for all $w \in S - \{u_1, v_1\}$, since $G - x$ is acyclic.

To avoid a cycle in $G^\sigma - y$, $G(xx_i) = 0 = G(xy_i)$ for $i = 1$ or 2, say $i = 1$. There are two cases here.

(1) $G(xz) = 0$. Now $G(xy) = 1$, since otherwise $\{x, v_1, v_2, z, y, x_1, y_1, u_1\}$ induces an (8-9) in G.

(2) $G(xz) = 1$. To avoid $\{x, z, v_1, y_1, x_1, y, u_1\}$ inducing a (7-5') we must have $G(xy) = 1$.

In both cases, $G(xy) = 1$. But $\{x, y, v_1, x_1, y_1\}$ induces a (5-1'). This contradiction proves the present case.

4.3.5 The other cases

Let $S = S(k, m)$ where we assume that $k \geq m$. Let z_1 and z_2 be the two centres of S, and $L = \{v_1, \ldots, v_k\}$ and $M = \{u_1, \ldots, u_m\}$ be the sets of leaves of S adjacent to z_1 and z_2, respectively. Remember in the following that $L_H(z)$ is the set of leaves in H adjacent to $z \in V(H)$.

Lemma 4.27

i. If S is of type (3s)-(8s), then $|L_S(z_i)| \leq 2$ for $i = 1$ or 2.

ii. If S^σ is the unique acyclic switch of S such that $S \neq S^\sigma$ and $z \in S$ has $|L_S(z)| = 3$, then x and y are each adjacent to exactly one, but different leaf at z.

Proof:
For Claim (i), assume both z_1 and z_2 have three leaves adjacent to them in S. By Lemma 4.25(ii), y is adjacent to one leaf at z_1 and one at z_2 giving a cycle in $G-x$ for the types (4s)-(8s). For (3s) we can apply the same reasoning, but taking y instead of x: $G^\sigma - y$ has a cycle. Note that we need that σ is the unique nonconstant selector mapping S into an acyclic graph. However, we have $k = 3 = m$ and by Lemma 4.17 the result follows.

To avoid a cycle in $G^\sigma - y$, x is adjacent to at most one of the leaves. Now, Claim (ii) follows from Lemma 4.22(ii) and Lemma 4.25(ii). □

Note that by Lemma 4.25(ii), Lemma 4.26(i) and (ii), and Lemma 4.27(i) it already follows that there are no critically cyclic graphs of order at least 12 unless they are in $[C_n]$ for $n \geq 12$.

4.3.6 The cases (2s)-(4s)

By the fact that $n \geq 10$ and Lemma 4.25(ii), we have $k = 3$ and $2 \leq m \leq 3$. In all these cases the unique nonconstant switch mapping S into an acyclic graph is $\sigma = \{z_1, z_2\}$ by Lemma 4.17. Recall that we still have $\sigma(x) = 0 = \sigma(y)$.

By Lemma 4.27(ii), x is adjacent to one of the v_i, say v_1, and y is adjacent to an other v_i, say v_3. To avoid a cycle in $G^\sigma - y$, x must be adjacent to z_2, and y is not adjacent to any of the other v_i or z_1.

We now go over the cases one by one.

(2s) $S = S(k,m) = K_{1,k} \cup K_{1,m}$. Because $n \geq 10$ and the bounds on k and m, we know that $k = 3 = m$. By Lemma 4.27, x is adjacent to a leaf u_i, say u_1 and y to a leaf u_i different from u_1, say u_3. Because of the unicity of σ, x must be adjacent to z_1 to ensure that $G^\sigma - y$ is acyclic.

The only remaining unknown is $G(xy)$. If $G(xy) = 0$, then we have the graph (5-1) $\{x, v_1, z, u_3, y\}$, and if $G(xy) = 1$, then we have (7-4) $\{u_1, x, y, v_3, z_1, v_2, u_2\}$.

(3s) $S = S(k,m) = K_{1,k} \cup K_{1,m} \cup K_1$. Because of the uniqueness of σ, S is mapped into a tree of type (4s). To avoid cycles in G^σ, necessarily $G(xz_1) = 1$, $G(xw) = 0$ (for the isolated vertex w of G) and $G(xu_i) = 0$ for all $u_i \in M$. By the above, $G(xz_2) = 1$ and $G(xv_2) = 0 = G(xv_3)$.

By Lemma 4.22(ii), $m = 2$ and y is adjacent to one of the u_i, say u_2.

The only unknown is the edge xy. If $G(xy) = 0$, then we have (7-5') for the vertices $\{v_1, x, z_2, u_1, y, v_3, z_1\}$, and if $G(xy) = 1$, then we have (7-4) $\{v_1, x, y, u_2, z_2, v_2, u_1\}$.

(4s) Now $S = S(k,m) = P_3(k,m)$. Because S is connected, y is not adjacent to any other vertex of S (except v_3). Hence, $m = 2$ and x is adjacent to one of the u_i, say u_1 (Lemma 4.22(ii)). To prevent cycles in $G^\sigma - y$, x must be adjacent to z_1. If $G(xy) = 0$, then we have (5-1) $\{x, u_1, z_2, v_3, y\}$ and if $G(xy) = 1$, then we have (7-4) in G^x $\{v_1, z_1, v_2, v_3, x, u_1, y\}$.

4.3.7 The cases (5s)-(8s)

(6s) $S(k,m) = P_2(k,m)$. In this case, $n \geq 10$ implies $k = 3 = m$, but then $G - x$ is not acyclic, because of Lemma 4.27(ii).

In the remaining cases (5s), (7s) and (8s), let w_1 be the neighbour of z_1 of degree 2 and let w_2 be the single unnamed vertex ($d_S(w_2)$ equals 0, 1 or 2 depending on the case), see Figure 4.12(5s), (7s) and (8s).

By Lemma 4.25(ii) and $n \geq 10$, $2 \leq k \leq 3$, $m \geq 1$, and $k + m \geq 4$. In all these cases the unique nonconstant switch mapping S into an acyclic graph is $\sigma = \{z_1, z_2\}$ by Lemma 4.17.

We can assume that x is adjacent to a vertex in L, say $G(xv_1) = 1$. This follows from Lemma 4.27(ii) if $k = 3$. On the other hand, if $k = 2$, then necessarily $m = 2$, since $n \geq 10$, and in this case $N_G(y) \cap L = \emptyset$ or $N_G(y) \cap M = \emptyset$ in order to avoid a cycle in $G - x$. By Lemma 4.22(ii), $N_G(x) \cap M \neq \emptyset$ or $N_G(x) \cap L \neq \emptyset$, respectively. Since now $k = m$ $(= 2)$, the assumption is validated.

Claim 1: $G(xz_1) = 1 = G(xz_2)$, and $G(xu) = 0$ for all $u \notin \{v_1, z_1, z_2, w_2, y\}$. Moreover, $G(xw_2) = 0$ if $d_S(w_2) \neq 0$ (that is, excepting the case (5s)).

Proof:

Recall that $\sigma(x) = 0$, and, indeed, $\sigma = \{z_1, z_2\}$. The claim follows, since $G^\sigma - y$ is acyclic.

Claim 2: $G(yv) = 1$ holds for exactly one vertex $v \in S - \{w_2\}$, and either (i) $v \in L$, say $G(yv_3) = 1$, in which case $k = 3$ and $m = 1$, (ii) $v \in M$, say $G(yu_2) = 1$, in which case $k = 2$, $m = 2$. Moreover, $G(yw_2) = 1$ holds only in the case (5s).

Proof:

The first statement follows from the fact that $G - x$ is acyclic. Now if y is not adjacent to a vertex of M, then $|M| = 1$ by Lemma 4.22(ii) and the fact that $G(xu) = 0$ for all $u \in M$. It follows that $k = 3$, and, consequently, y is adjacent to a vertex of L. On the other hand, if $G(yu) = 1$ for a $u \in M$, then $G(yv) = 0$ for all $v \in L$ to avoid a cycle in $G - x$, and in this case, $k = 2$ by Lemma 4.22. That $G(yw_2) = 1$ in the case (5s) follows from Lemma 4.25(i). In the other two cases, $G(yw_2) = 1$ would result in a cycle in $G - x$.

These two claims together determine G with the exception of the value for $G(xy)$.

The cases are all excluded:

(5s) x is not adjacent to w_1 and neither is y. Hence in $G^\sigma - y$ the vertex w_1 is isolated contradicting Lemma 4.24.

(7s) In both cases, $G(xy) = 1$ to avoid (7-4) as being the subgraph induced by the vertices $\{x, z_1, w_1, z_2, v_2, w_2, y\}$. Now G contains a switch of (7-4) if $k = 3$ and $m = 1$ (this is $G^{z_1} - \{v_1, v_3, u_2\}$), and G contains (7-5') if $k = 2 = m$ (this is $G - \{u_1, v_2, z_2\}$).

(8s) In both cases, $G(xy) = 1$ to avoid (6-1) as being the subgraph induced by the vertices $\{x, z_1, w_1, w_2, z_2, y\}$. Now $\{x, z_1, w_1, w_2, z_2, y, u_1\}$ induces (7-3').

This completes the proof of Theorem 4.14.

4.3.8 Concluding remarks

Finding the critically cyclic graphs was done as follows: a program was written in `C` that listed for a number n of vertices a representative of each switching class that did not contain any acyclic switches. In a later phase, when we were looking for critically cyclic graphs on n vertices, we only had to make sure that all critically cyclic graphs of lower order did not occur in these graphs. The program was run in this way for up to 12 vertices. We used here the files from Spence [45] which list representatives for the switching classes up to isomorphism and up to complementation for up to 10 vertices.

A computer program in the functional language Scheme verified that the critically cyclic graphs found were in fact critically cyclic. (For more information on this see Appendix B.) Also, the authors verified this by hand.

In our proofs, not all of the critically cyclic graphs were used. The graphs that were not used are (8-10)-(8-15) and (9-3)-(9-5). Lemma 4.23 excludes the cycles C_8 (8-11) and C_9 (9-4). For the other graphs, except (8-12), the reason is that if they are induced subgraphs of any graph G of order at least 10, then G also contains one of the cyclic graphs from Figure 4.9, 4.10 and 4.11, but without (8-10), (8-13)-(8-15), (9-3) and (9-5).

As an aside we note that our program found that the graphs (8-9) and (8-12) have a similar property: adding two vertices to either of these graphs in any way, always results in a graph that contains a switch of one of the other critically cyclic graphs.

The graph (8-12) does not occur in our proofs, because it is overruled by Lemmas 4.23 and 4.24, that is, if G is a forbidden graph of order 10 that does not have 2 isolated vertices and such that $G - x$ is acyclic and $G - \{x, y\}$ is special, then G contains an induced critically acyclic graph that was used in the proofs.

Part II

Switching Classes of Graphs with Skew Gains

Chapter 5

Gain Graphs

In this chapter we generalize the graphs of the first part of the thesis to graphs with skew gains. Along the same lines we generalize selectors and switching classes. Analogues of many of the results in Chapter 3 are shown to hold in this chapter.

In Section 5.2 and Section 5.3 we also prove some results about anti-involutions, which are anti-automorphisms of order at most two. (These results are originally from Hage and Harju [23], unless otherwise indicated. For the results in Section 5.3 we had help from A. Tijdeman and W. Kosters.) These bijections on the group generalize the group inversion and constitute an extension of the gain graphs of Zaslavsky [50] and the voltage graphs of Gross and Tucker [20] to skew gain graphs.

5.1 Definitions

Let Γ be a group. A function $\delta : \Gamma \to \Gamma$ is an *anti-involution*, if it is an anti-automorphism of order at most two, that is, δ is a bijection and for all $x, y \in \Gamma$, $\delta(xy) = \delta(y)\delta(x)$ and $\delta^2(x) = x$. We write (Γ, δ) for a group Γ with a given anti-involution δ. The set of anti-involutions on Γ is denoted by $\mathrm{INV}(\Gamma)$. For abelian groups anti-involutions coincide with involutions, i.e., automorphisms of order at most two.

Example 5.1
(1) The group inversion of a group is an anti-involution of that group.
(2) Let $\Gamma = S_3$. This group has the following four anti-involutions:

a i $f_i(a)$	r_0	r_1	r_2	s_0	s_1	s_2
0	r_0	r_2	r_1	s_0	s_1	s_2
1	r_0	r_1	r_2	s_1	s_0	s_2
2	r_0	r_1	r_2	s_0	s_2	s_1
3	r_0	r_1	r_2	s_2	s_1	s_0

Note that f_0 is the group inversion. $\diamond$

The graphs in the first part of this thesis were undirected. In this part of the thesis they will be directed. Define $E_2(V) = \{(u, v) \mid u, v \in V\}$, the set of nonreflexive, directed edges over V. We usually write uv for the edge (u, v) like in the first part of the thesis, but now $uv \neq vu$. For an edge $e = uv$, the *reverse* of e is $e^{-1} = vu$

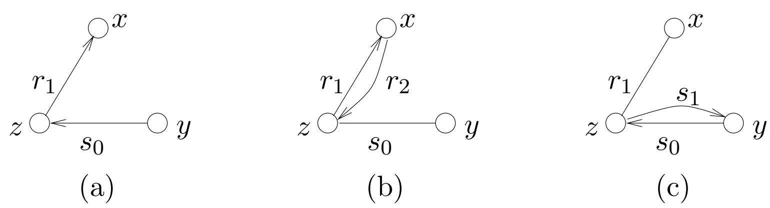


Figure 5.1: A "near" gain graph, an element of $\mathbf{L}_{P_3}(S_3, f_0)$, and an element of $\mathbf{L}_{P_3}(S_3, f_1)$

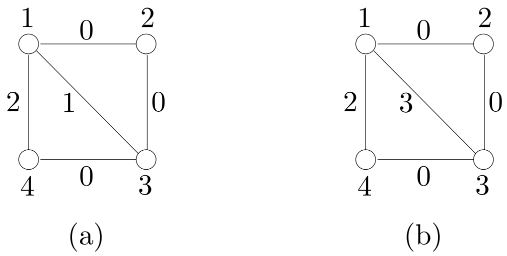


Figure 5.2: Two elements of $\mathbf{L}_G(\mathbf{Z}_4, id)$

We consider graphs $G = (V, E)$ where the set of edges $E \subseteq E_2(V)$ satisfies the following *symmetry condition*:

$$\text{if } e \in E \text{ then also } e^{-1} \in E.$$

Such graphs can be considered as undirected graphs where the edges have been given a two-way orientation.

Let $G = (V, E)$ be a graph and (Γ, δ) a group with anti-involution. A pair (G, g) where g is a mapping $g : E \to (\Gamma, \delta)$ into the group Γ is called a (Γ, δ)-*gain graph* (on G) (or a *graph with skew gains* or a *skew gain graph*), if g satisfies the following *reversibility condition*

$$g(e^{-1}) = \delta(g(e)) \quad \text{for all } e \in E \ . \tag{5.1}$$

In the future we will refer to a skew gain graph (G, g) simply by g unless confusion arises. We adopt in a natural way some of the terminology of graph theory for graphs with skew gains.

The class of (Γ, δ)-gain graphs on G will be denoted by $\mathbf{L}_G(\Gamma, \delta)$ or simply by $\mathbf{L}_G$. A *gain graph* is a $(\Gamma, {}^{-1})$-gain graph; these are also called *inversive* skew gain graphs.

The set of *gains* of g is

$$A(g) = \{g(e) \mid e \in E(G)\} \subseteq \Gamma \ .$$

We call g *abelian* if $A(g) \subseteq Z(\Gamma)$, where, as you might recall, $Z(\Gamma)$ is the centre of Γ.

Example 5.2
(1) In Figure 2.4(a) a graph is depicted. The labelled graph in Figure 2.4(b) is not just a pictorial representation of this graph, but also a $(\mathbf{Z}_2, {}^{-1})$-gain graph where the underlying graph is complete. Shortly, we will define a notion of switching that

corresponds, for the case of $\mathbf{Z}_2$, exactly to the notion of switching in the first part of this thesis.
(2) In Figure 5.1, three labelled graphs are depicted. The first of these does not adhere to the symmetry condition, but it should be clear that if the anti-involution is known we can add the missing reverse edges and determine their labels uniquely. If we know the anti-involution to be the group inverse, f_0 in the table of Example 5.1, then the result of adding the missing edges will be the (S_3, f_0)-gain graph in Figure 5.1(b). If the anti-involution is f_1 from the table of Example 5.1, then the result will be the (S_3, f_1)-gain graph in Figure 5.1(c).
(3) In Figure 5.2 we have listed two examples of (Γ, δ)-gain graphs with the group $\Gamma = \mathbf{Z}_4$ and the anti-involution equal to the identity function. The underlying graph is the graph of Figure 2.4(a). ◇

A function $\sigma : V \to \Gamma$ is called a *selector*. For each selector σ we associate with g a (Γ, δ)-gain graph g^σ on $G = (V, E)$ by letting, for each $uv \in E$,

$$g^\sigma(uv) = \sigma(u)g(uv)\delta(\sigma(v)) \ . \tag{5.2}$$

We use $\mathbf{S}(V, \Gamma)$ or simply $\mathbf{S}$, to denote the set of the selectors from V to Γ.

Note that the definition of selector is in line with our definition of selector in the first part of this thesis, since a subset of the set of vertices is equivalent to selecting in each node an element of $\mathbf{Z}_2$ and $\delta = {}^{-1} = id$. It is easy to verify that in this restricted case (5.2) corresponds to (3.1) in the first part of the thesis.

Example 5.3
Let g_1 and g_2 be the $(\mathbf{Z}_4, id)$-gain graphs of Figure 5.2(a) and (b) respectively. The second of these, g_2, can obtained from g_1 by applying the selector σ that maps 1 and 3 to 3, and 2 and 4 to 1. For example, the label of the edge $(1, 3)$ is computed as follows: $g_2(1, 3) = g_1^\sigma(1, 3) = \sigma(1)g_1(1, 3)\delta(\sigma(3)) = 3 + 1 + \delta(3) = 3 + 1 + 3 = 3$, where $+$ is of course addition modulo 4. ◇

We note that g^σ satisfies the reversibility condition (5.1), by the following lemma.

Lemma 5.4
For each $g \in \mathbf{L}_G(\Gamma, \delta)$ and selector $\sigma : V \to \Gamma$, also $g^\sigma \in \mathbf{L}_G(\Gamma, \delta)$.
Proof:
Indeed,

$$\begin{aligned} g^\sigma(uv) &= \sigma(u)g(uv)\delta(\sigma(v)) = \sigma(u)\delta(g(vu))\delta(\sigma(v)) = \sigma(u)\delta(\sigma(v)g(vu)) \\ &= \delta(\sigma(v)g(vu)\delta(\sigma(u))) = \delta(g^\sigma(vu)) \ , \end{aligned}$$

which shows the claim. □

The class $[g] \subseteq \mathbf{L}_G(\Gamma, \delta)$ defined by

$$[g] = \{g^\sigma \mid \sigma : V \to \Gamma\}$$

is called the *switching class* generated by g.

In Figure 5.3 we have included an example of a switching class that will be used in this and the coming chapters, e.g., Example 5.6, 6.18 and 7.27.

The set $\mathbf{S}(V, \Gamma)$ of selectors can be made into a group in a natural way by defining for all selectors σ and τ,

$$(\sigma\tau)(u) = \sigma(u)\tau(u)$$

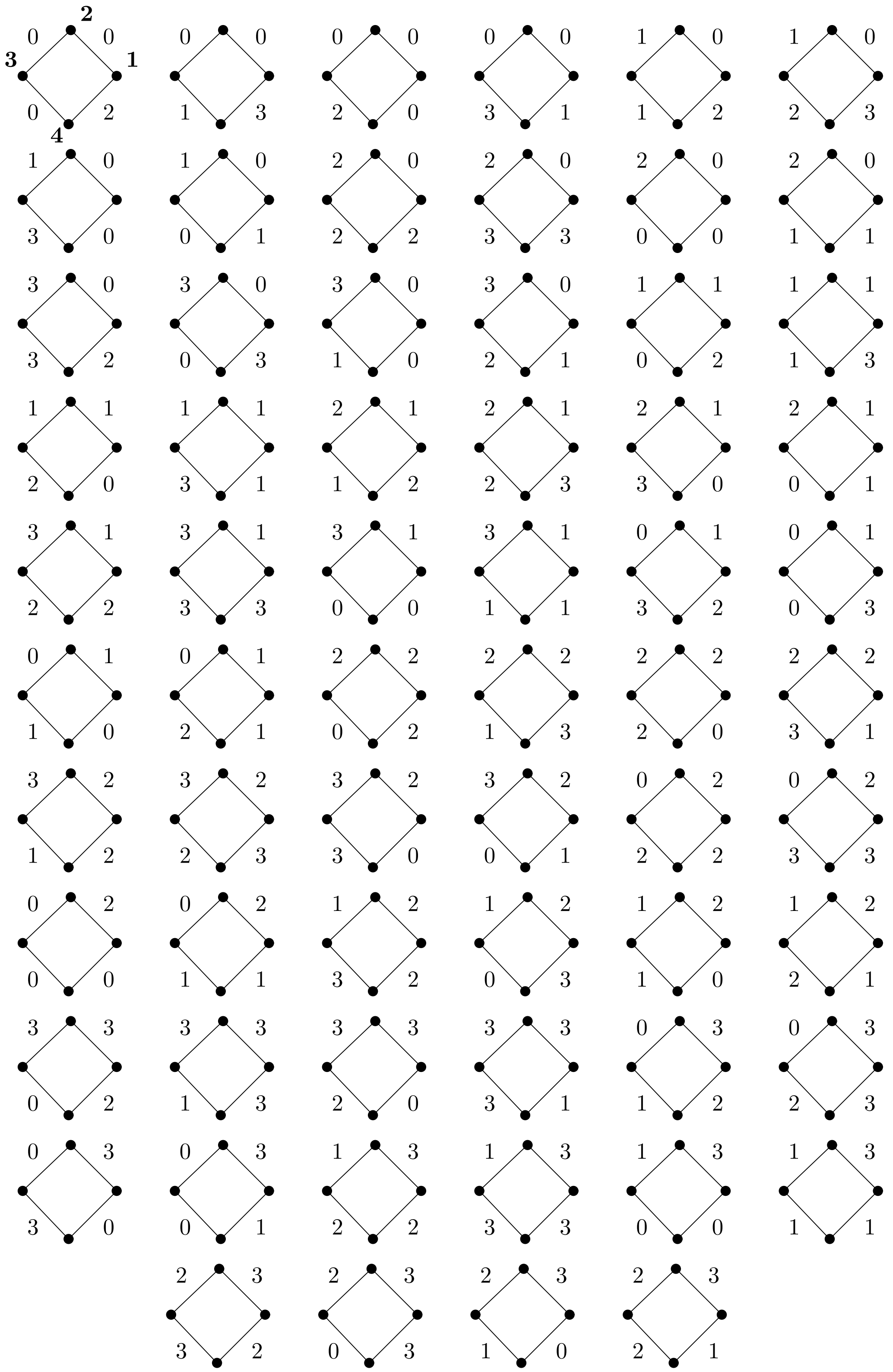


Figure 5.3: A complete switching class

for all $u \in V$.

Closure under composition of selectors is something that we would expect in our model: it is a consequence of Axiom A1 from the introduction.

Lemma 5.5
For each $g \in \mathbf{L}_G(\Gamma, \delta)$ and selectors $\sigma, \tau \in \mathbf{S}(V, \Gamma)$, $g^{\sigma\tau} = (g^\tau)^\sigma$.
Proof:
Let $uv \in E(G)$. Then

$$\begin{aligned}(g^\tau)^\sigma(uv) &= \sigma(u)\tau(u)g(uv)\delta(\tau(v))\delta(\sigma(v)) \\ &= \sigma(u)\tau(u)g(uv)\delta(\sigma(v)\tau(v)) \\ &= (\sigma\tau)(u)g(uv)\delta((\sigma\tau)(v)) = g^{\sigma\tau}(uv)\ .\end{aligned}$$

□

Hence $\mathbf{S}(V, \Gamma)$ is a group that *acts on* the (Γ, δ)-gain graphs (on the left), that is, $\mathbf{S}(V, \Gamma)$ can be thought of as a permutation group on $\mathbf{L}_G(\Gamma, \delta)$. It follows then that each switching class $[g]$ is generated by each of its elements.

In the group $\mathbf{S}(V, \Gamma)$ the *trivial selector* σ_1, for which $\sigma_1(u) = 1_\Gamma$ for all $u \in V$, is the group identity of $\mathbf{S}(V, \Gamma)$; and the inverse of a selector σ, denoted by σ^{-1}, is found by inverting the selected values in the vertices, that is, $\sigma^{-1}(u) = \sigma(u)^{-1}$ for $u \in V$. Hence, the relation $g \sim h$, which holds if there exists a selector σ such that $h = g^\sigma$, is, as in the first part of this thesis, an equivalence relation on the (Γ, δ)-gain graphs.

Note that in the terminology of permutation groups, a switching class generated by g is the orbit of g. The stabilizer of g corresponds to the set of selectors that leave g unchanged. Naturally, this includes the trivial selector, but in general it may include others as the following example shows.

Example 5.6
Let g be the first $(\mathbf{Z}_4, id)$-gain graph of Figure 5.3. To find the selectors σ so that $g^\sigma = g$ we first note that if $\sigma(1) = a$ for some $a \in \mathbf{Z}_4$, then necessarily $\sigma(2) = \delta(a^{-1}) = a^{-1}$. In the same way, $\sigma(3) = \sigma(2)^{-1} = \sigma(1) = a$ and $\sigma(4) = \sigma(2)$. The selected values guarantee that the labels on the path $(1, 2, 3, 4)$ stay the same. The only possibility for change lies in the edge $(1, 4)$ (and $(4, 1)$).

If we want to leave the path $(1, 2, 3, 4)$ unchanged, then the value selected in 1 determines the other selected values, and so there are only 4 selectors left to consider. If $\sigma(1) = 0$, then σ is the identity selector and obviously $g^\sigma = \sigma$. However, this is also the case if $\sigma(1) = i$ for $i = 1, 2, 3$ as the reader can easily verify by computing that always $\sigma(1)g(1, 4)\delta(\sigma(4)) = 2$. Defining σ_i to be a selector of the sort just described with $\sigma(1) = i$, we find that $\text{Stab}(g) = \{\sigma_0, \sigma_1, \sigma_2, \sigma_3\}$.

Note that g_1 of Example 5.3 can be obtained from g if we add an edge $\{1, 3\}$ labelled with 1 (see Figure 5.2(a)). In Example 5.3 we saw that σ_3 maps g_1 to g_2 of Figure 5.2(b) and not to itself. This also holds for σ_1, but σ_0 and σ_2 do map g_1 to itself. Hence $\text{Stab}(g_1) = \{\sigma_0, \sigma_2\}$. As we shall see later this is a consequence of the fact that the underlying graph is not bipartite anymore. ◇

For a gain graph g, let $g^{-1} \in \mathbf{L}_G$ be such that $g^{-1}(uv) = g(uv)^{-1}$ for all $uv \in E(G)$, and for g, g', gg' is defined edgewise by

$$(gg')(uv) = g(uv)g'(uv) \text{ for all } uv \in E(G)\ .$$

Note that gg' does not necessarily satisfy Equation (5.1).

Lemma 5.7
The labelled graph gg' satisfies Equation (5.1) for all $g, g' \in \mathbf{L}_G(\Gamma, \delta)$ if and only if Γ is abelian.
Proof:
Compute $gg'(vu) = \delta(gg'(uv)) = \delta(g(uv)g'(uv)) = \delta(g'(uv))\delta(g(uv)) = g'(vu)g(vu)$, which is true only if the labels of g and g' commute. □

In the general case the set of graphs on G with labels from Γ form a group, and if Γ is abelian, then $\mathbf{L}_G(\Gamma, \delta)$ is also a group.

Some remarks about the history are now appropriate, as we can now encompass many of the notions available in the literature in our framework. The switching classes of the first part of our thesis, as devised by Van Lint and Seidel [37], is the most restricted case where the group is $\mathbf{Z}_2$, the anti-involution is by necessity the group inversion, and the underlying graph is a complete graph. The important references here are the survey papers of Seidel [43] and Seidel and Taylor [44].

The *signed graphs* of Harary [27], but see also Zaslavsky [48] and [49], are slightly more general in that the underlying graph is now arbitrary. The name "signed" comes from the fact that edges are labelled by $+$ and $-$ and switching is done according to the rules of multiplication, i.e., '$+$' times '$-$' equals '$-$'. This is clearly isomorphic to the operation of addition modulo 2 in $\mathbf{Z}_2$.

In topological graph theory $(\Gamma, {}^{-1})$-gain graphs on arbitrary graphs are called *voltage graphs* (see Gross and Tucker [20]). These are equivalent to the *gain graphs* of Zaslavsky (see [50], in which also a more general model, that of *biased graphs*, can be found). The model of *dynamic labelled 2-structure* as devised by Ehrenfeucht and Rozenberg [16] was the first to include an anti-involution, but the underlying graphs were complete.

5.2 Anti-involutions

Let $\mathrm{AUT}_2(\Gamma)$ denote the set of automorphisms of Γ of order at most two.

Lemma 5.8
A function δ is an anti-involution of Γ if and only if $\alpha \in \mathrm{AUT}_2(\Gamma)$ for $\alpha = {}^{-1} \cdot \delta$.
Proof:
Let δ be an anti-involution and define α as above. Now, $\alpha(ab) = \delta(ab)^{-1} = (\delta(b)\delta(a))^{-1} = \delta(a)^{-1}\delta(b)^{-1} = \alpha(a)\alpha(b)$.

The other direction is as follows: let α be an automorphism of Γ. Define $\delta = {}^{-1} \cdot \alpha$. Then, $\delta(ab) = \alpha(ab)^{-1} = (\alpha(a)\alpha(b))^{-1} = \alpha(b)^{-1}\alpha(b)^{-1} = \delta(b)\delta(a)$, so δ is an anti-involution. □

Corollary 5.9
Let $a \in \Gamma$.

i. $\delta(1_\Gamma) = 1_\Gamma$,

ii. ${}^{-1}$ is an anti-involution,

iii. Γ is abelian if and only if the identity function is an anti-involution,

iv. $\delta(a^{-1}) = \delta(a)^{-1}$,

v. $\delta(a^n)^m = \delta(a)^{nm}$ for integers n, m,

vi. $\#\delta(a) = \#a$,

vii. $|\mathrm{AUT}_2(\Gamma)| = |\mathrm{INV}(\Gamma)|$

The next lemma says that each anti-involution δ, that is not the identity nor the group inversion, possesses a nontrivial fixed point, $\delta(a) = a$, and a nontrivial 'inverse point', $\delta(b) = b^{-1}$.

Lemma 5.10
Let δ be an anti-involution of a finite group Γ.

i. Either there exists an element $a \neq 1_\Gamma$ such that $\delta(a) = a$, or δ is the inversion of Γ and Γ has odd order.

ii. Either there exists an element $a \neq 1_\Gamma$ such that $\delta(a) = a^{-1}$, or δ is the identity function and Γ is an abelian group of odd order.

Proof:
For (i), assume that δ has no nontrivial fixed points. We note that for all $a \in \Gamma$, $\delta(a\delta(a)) = a\delta(a)$, and hence each $a\delta(a)$ is a fixed point of δ. By assumption $a\delta(a) = 1_\Gamma$ for all a; hence $\delta(a) = a^{-1}$, and hence δ is the group inversion. In this case, the order of Γ is odd, because if it were even, then (by Cauchy's theorem) Γ has an element x of order 2, $x^2 = 1_\Gamma$, and then $\delta(x) = x^{-1} = x$.

For (ii), we deduce that if $a\delta(a) = b\delta(b)$, then $\delta(a^{-1}b) = \delta(b)\delta(a^{-1}) = b^{-1}a = (a^{-1}b)^{-1}$. On the other hand, if $a\delta(a) \neq b\delta(b)$ for all $a \neq b$, then clearly $\Gamma = \{a\delta(a) \mid a \in \Gamma\}$ and thus for each $b \in \Gamma$ there exists an a such that $b = a\delta(a)$, which implies that for all b, $\delta(b) = b$. Thus δ is the identity function. In this case, Γ is abelian, and it has odd order, since $\delta(1_\Gamma) = 1_\Gamma$ and the other elements come in pairs $\{a, a^{-1}\}$. □

Theorem 5.11
The centre $\mathrm{Z}(\Gamma)$ of Γ is closed under every anti-involution of Γ. Furthermore, for each anti-involution δ either $\delta(z) = z$ for all $z \in \mathrm{Z}(\Gamma)$ or there exists an element $x \in \mathrm{Z}(\Gamma)$ such that $\delta(x) = x^{-1}$ with $x \neq 1_\Gamma$.
Proof:
For all $x \in \mathrm{Z}(\Gamma)$ and $y \in \Gamma$, $\delta(x)y = \delta(\delta(y)x) = \delta(x\delta(y)) = y\delta(x)$, which shows that also $\delta(x) \in \mathrm{Z}(\Gamma)$. Hence, if δ is an anti-involution of Γ, then δ is an anti-involution of $\mathrm{Z}(\Gamma)$. The second claim follows from Lemma 5.10. □

Now it is shown that computing the direct product of two groups involves taking the cartesian product of the sets of anti-involutions of both groups.

We write also

$$\delta^{[i]} : \Gamma_i \to \Gamma_i$$

for the restriction of $\delta^{(i)}$ onto the subgroup Γ_i for $i = 1, 2$.

The following example shows that an anti-involution of a direct product cannot necessarily be obtained by projections of its components.

Example 5.12
Let $\Gamma = \Gamma_1 \times \Gamma_1$ for a group Γ_1, and let δ be the reversed inversion on Γ, that is,

$$\delta(a_1, a_2) = (a_2^{-1}, a_1^{-1})$$

for all $a_1, a_2 \in \Gamma_1$. Then δ is an anti-involution of Γ. Indeed, it is clear that $\delta^2 = id$, and, moreover, for all $a_i, b_i \in \Gamma_1$,

$$\begin{aligned}\delta(a_1,a_2)\cdot\delta(b_1,b_2) &= (a_2^{-1},a_1^{-1})(b_2^{-1},b_1^{-1}) = (a_2^{-1}b_2^{-1},a_1^{-1}b_1^{-1})\\ &= ((b_2a_2)^{-1},(b_1a_1)^{-1}) = \delta(b_1a_1,b_2a_2)\\ &= \delta((b_1,b_2)\cdot(a_1,a_2)).\end{aligned}$$

However, δ is not of the form $\delta = (\delta_1, \delta_2)$ for any functions (let alone anti-involutions) δ_1 and δ_2 of Γ_1, if Γ_1 is nontrivial. ◇

However, we have

Theorem 5.13 [new, Hage and Harju [22]]
Let $\Gamma = \Gamma_1\Gamma_2$ be the (inner) direct product of the normal subgroups Γ_1 and Γ_2.

i. If $\delta_i \in \mathrm{INV}(\Gamma_i)$ for $i = 1, 2$, then the function $\delta : \Gamma \to \Gamma$ defined by

$$\delta(a) = \delta_1(a_1)\delta_2(a_2) \quad \text{for } \ a = a_1a_2,\ a_i \in \Gamma_i$$

is an anti-involution of Γ.

ii. If $\delta \in \mathrm{INV}(\Gamma)$, then there are normal subgroups Δ_1 and Δ_2 of Γ such that $\Gamma = \Delta_1\Delta_2$ is a direct product with $|\Delta_1| = |\Gamma_1|$, $|\Delta_2| = |\Gamma_2|$ for which $\delta^{[i]} : \Delta_i \to \Delta_i$ is an anti-involution of Δ_i for $i = 1, 2$.

Proof:
In order to prove (i), let δ_i be anti-involutions as stated. Let $a = a_1a_2$ and $b = b_1b_2$ for $a_1, b_1 \in \Gamma_1$ and $a_2, b_2 \in \Gamma_2$. Now, for the function δ,

$$\begin{aligned}\delta(ab) &= \delta(a_1a_2b_1b_2) = \delta(a_1b_1a_2b_2) = \delta_1(a_1b_1)\delta_2(a_2b_2)\\ &= \delta_1(b_1)\delta_1(a_1)\delta_2(b_2)\delta_2(a_2) = \delta_1(b_1)\delta_2(b_2)\delta_1(a_1)\delta_2(a_2)\\ &= \delta(b_1b_2)\delta(a_1a_2) = \delta(b)\delta(a)\end{aligned}$$

and thus δ is an anti-automorphism of Γ. Further, the condition $\delta^2(a) = a$ is easily checked.

For (ii) suppose first that $\delta \in \mathrm{INV}(\Gamma)$, and define

$$\Delta_1 = \{\delta(a) \mid a \in \Gamma_1\} \ \text{ and } \ \Delta_2 = \{\delta(b) \mid b \in \Gamma_2\}\ .$$

Clearly, $a \in \Delta_1$ (resp. in Δ_2) if and only if $\delta(a) \in \Gamma_1$ (resp. $\delta(a) \in \Gamma_2$). Since an anti-involution is a bijection, we have immediately that $|\Delta_i| = |\Gamma_i|$ for $i = 1, 2$.

We show then that Δ_1 and Δ_2 are normal subgroups of Γ. Indeed, let $y = aua^{-1}$ for some $a \in \Gamma$ and $u \in \Delta_1$. Now, $\delta(y) = \delta(a)^{-1}\delta(u)\delta(a) \in \Gamma_1$, since $\delta(u) \in \Gamma_1$ and Γ_1 is a normal subgroup of Γ. This shows that Δ_1 is normal in Γ. The case for Δ_2 is symmetric.

Next we observe that $\Delta_1 \cap \Delta_2 = \{1_\Gamma\}$ is the trivial subgroup of Γ. Further, if $a \in \Gamma$, then $a = a_2a_1$ for some $a_i \in \Gamma_i$, because $\Gamma = \Gamma_2\Gamma_1$. Therefore, $\delta(a) = \delta(a_1)\delta(a_2)$, where $\delta(a_1) \in \Delta_1$ and $\delta(a_2) \in \Delta_2$. Since each element $b \in \Gamma$ is an image $b = \delta(a)$, we have shown that $\Gamma = \Delta_1\Delta_2$ is a direct product of Γ.

It is clear, that $\delta^{[i]}$ is an anti-involution of Δ_i for both $i = 1$ and $i = 2$. □

In particular, if Γ is an abelian group, then it is a direct product (sum) of cyclic groups, and thus Theorem 5.13 states that the anti-involutions of an abelian group can be obtained from the cyclic groups $\mathbf{Z}_{p^k}$ that are its direct components. However, counting the number of anti-involutions of Γ is *not* reduced in this way to the number of anti-involutions of its direct components, because part (ii) of Theorem 5.13 uses 'swappings' of subgroups in the construction of the Δ_i, $i = 1, 2$.

Example 5.14
Let $\Gamma = \mathbf{Z}_2 \times \mathbf{Z}_2$. Then $\Gamma = \Gamma_1\Gamma_2$ is an inner direct product of $\Gamma_1 = \mathbf{Z}_2 \times \{0\}$ and $\Gamma_2 = \{0\}\times\mathbf{Z}_2$. Note that Γ_1 and Γ_2 are normal subgroups of Γ with $\Gamma_1\cap\Gamma_2 = \{(0,0)\}$. The groups Γ, Γ_1 and Γ_2 have the identity as their anti-involution (it is equal to the group inversion), but in the case of the former, it is not the only one. The function δ is an involution of Γ:

$$\delta((a, b)) = (b, a) \text{ for } a, b \in \{0, 1\} \; .$$

We now proceed along the lines of Theorem 5.13. The swapped subgroups are $\Delta_1 = \Gamma_2$ and $\Delta_2 = \Gamma_1$, and the reader can verify that $\delta^{[1]}$ is an involution of $\Delta_1 = \Gamma_2$ as follows: take for instance $(1, 0) \in \Delta_1$. We have to verify that $\delta^{[1]}((1, 0)) \in \Delta_1$. Recall from Section 2.2 that $\delta(x) = \delta^{(1)}(x)\delta^{(2)}(x)$ where $\delta^{(1)}(x) \in \Delta_1$ and $\delta^{(2)}(x) \in \Delta_2$. In this case $\delta((0, 1)) = (1, 0) = (1, 0)(0, 0) = \delta^{(1)}((0, 1))\delta^{(2)}((0, 0))$. Hence $\delta^{[1]}((0, 1)) = (0, 1)$. The same holds for $(0, 0)$ and so $\delta^{[1]}$ is the identity on Δ_1 which is an involution of that group. Hence the swapping of subgroups compensates for the swapping done by the involutions. ◇

5.3 Anti-involutions of cyclic groups

This section is devoted to an investigation of the anti-involutions of cyclic groups, see also Chapter VIII, p. 98 in Hardy and Wright [29].

For the group $\mathbf{Z}_n$, suppose $\delta(1) = k$, where 1 is the generator of the group. Now, $\delta(i) = ik \bmod n$ and so $\delta(k) = k^2 \bmod n$, i.e.,

$$k^2 \equiv 1 \bmod n. \tag{5.3}$$

As an example let $n = 16$. The first anti-involution, the identity, is found by taking $k = 1$, which works for any cyclic group. If (5.3) holds for k then it also holds for $n - k$, so $k = 15$ is also possible; this is the group inversion. It is easy to see that if $k > 1$ then $k \geq \sqrt{n+1}$ and hence k cannot be equal to 2, 3 or 4. From Corollary 5.9 we know that $\#1 = \#k$ and hence k generates $\mathbf{Z}_n$ as well, implying $(n, k) = 1$. This means that other possible values for k that have to be examined are 5 and 7. Of these only 7 works (and thus also 9).

Let $\xi : \mathbf{N} \to \{-1, 0, 1\}$ be such that

$$\xi(n) = \begin{cases} 1 & \text{if } 8|n \\ -1 & \text{if } 2|n \text{ and } 4 \not| n \\ 0 & \text{otherwise} \end{cases}$$

Lemma 5.15
If $n = p^m$ for some prime p and $m \in \mathbf{N}$ then (5.3) has exactly $2^{1+\xi(n)}$ solutions.
Proof:
If (5.3) holds, then $p^m|(k-1)(k+1)$, which means that $p^i|(k-1)$, $p^j|(k+1)$ with $i + j = m$ and $i, j \geq 0$ (note that never $i = j = 0$).

If $j = 0$ (or $i = 0$) then $p^m|k-1$ (or $p^m|k+1$) yields

$$k \equiv \pm 1 \mod p^m \ . \tag{5.4}$$

We note that if $p = 2$ and $m = 1$, then the solutions for $+1$ and -1 in (5.4) are the same.

Suppose then that $i, j > 0$. First let $i < j$. This implies that $p^i|k-1$ and $p^i|k+1$, so that $p^i|(k+1)-(k-1) = 2$. Now $i > 0$ implies $i = 1$ and $p = 2$, and consequently $j = m-1$ and substitution gives $2^{m-1}|k+1$. Applying the same reasoning to $j < i$ we get

$$k \equiv \pm 1 \mod 2^{m-1} \ . \tag{5.5}$$

If $m = 1$ or $m = 2$, (5.5) yields solutions that are equal to the ones of (5.4).

Summarizing, we get one solution if $p = 2$ and $m = 1$, four solutions if $p = 2$ and $m \geq 3$ and two if $p \neq 2$ or $p = 2$ and $m = 2$.

In the above we have only proven half of what we need to prove, namely indicating possible solutions. The fact that these solutions do indeed always exist can be verified easily from (5.3).

Hence for every n the number of solutions equals $2^{1+\xi(n)}$. □

If we have a factor $p = 2$, then not only do we have solutions $k = 1$ and $k = p^m - 1$, but also two solutions that are 'halfway'. In some cases a number of these solutions collapse together yielding a smaller amount of solutions. For instance the group $\mathbf{Z}_2$ has only one anti-involution.

Theorem 5.16
Let $n = p_1^{m_1} \cdot \ldots \cdot p_r^{m_r}$, $r \geq 1$, for prime numbers $p_j \geq 2$, $m_j > 0$ for $1 \leq j \leq r$ with $p_i < p_{i+1}$, and for $1 \leq i \leq r-1$. Then $|\mathrm{INV}(\mathbf{Z}_n)| = 2^{r+\xi(n)}$.
Proof:
A cyclic group $\mathbf{Z}_n$ can be written

$$\mathbf{Z}_n \cong \mathbf{Z}_{p_1^{m_1}} \oplus \mathbf{Z}_{p_2^{m_2}} \oplus \cdots \oplus \mathbf{Z}_{p_r^{m_r}}$$

where $r > 0$, $p_j \geq 2$, $m_j > 0$ for $1 \leq j \leq r$ and $p_i < p_{i+1}$, for $1 \leq i \leq r-1$.

Note first of all that we are constructing the involutions for arbitrary cyclic groups from the involutions of the primary groups $\mathbf{Z}_{p_i^m}$ for primes p_i. From Example 5.12 we know that we have to be careful, because it may be that not every involution of $\mathbf{Z}_n$ can be constructed in this way. For this, note that the order of the elements in any primary group divides the order of the primary group. Because all primes in the decomposition are different, we need not fear the swapping of Example 5.12, because an involution always maps an element to an element of the same order. In other words: involutions on $\mathbf{Z}_n$ are such that they only map an element to an element of the same primary group. Hence to every $p_j^{m_j}$ $(1 \leq j \leq r)$ Lemma 5.15 can be applied yielding the given number of solutions.

We can then use Theorem 122 of [29] to conclude that the total number of solutions equals the product of the numbers of solutions to the separate equations for $p_j^{m_j}$ for $1 \leq j \leq r$.

If $p_1 \neq 2$ then every prime number gives two solutions yielding a total of 2^r solutions, and indeed in this case $\xi(n) = 0$. If $p_1 = 2$ then we have three possibilities: $m_1 = 1$ and hence p_1 yields only one solution and we get a total of 2^{r-1} solutions. The other two cases, $m_1 = 2$ and $m_1 > 2$, follow similarly. □

Note that the anti-involutions themselves can be found by solving two sets of equations using the Chinese Remainder Theorem, see [29].

Example 5.17
Let $\Gamma = \mathbf{Z}_{60}$. The factoring of 60 into prime powers is $2^2 \cdot 3 \cdot 5$. According to Theorem 5.16 we should get $2^{3+0} = 8$ anti-involutions. The anti-involutions are determined by the values that the generator 1 is mapped to: these values are $1, 11, 19, 29, 31, 41, 49, 59$. This can be verified along the lines of the example at the beginning of this section.

For the group $\mathbf{Z}_{81}$ we obtain two anti-involutions: the identity and the group inversion. ◇

5.4 Spanning acyclic skew gain subgraphs

A vertex $u \in V$ is called a *horizon* of a (Γ, δ)-gain graph g on $(V, E_2(V))$, if for all $v \in V - \{u\}$, $g(uv) = 1_\Gamma$. A horizon was called an isolated vertex in the context of undirected graphs.

As will become clear later, (Γ, δ)-gain graphs that have a horizon are very useful. First we prove that for a fixed vertex u, every switching class contains an element in which u is a horizon. Note however that here we do not have uniqueness, that is, unlike in Lemma 3.9.

Let g be a (Γ, δ)-gain graph on $G = (V, E_2(V))$. For a vertex $u \in V$ and an element $a \in \Gamma$, define the selector $\sigma_{u,a}$ such that

$$\sigma_{u,a}(u) = a \text{ and } \sigma_{u,a}(v) = \delta(ag(uv))^{-1} \text{ for } v \in V - \{u\} \ . \tag{5.6}$$

It is easy to verify that u is a horizon in $g^{\sigma_{u,a}}$: for $v \in V - \{u\}$,

$$\begin{aligned} g^{\sigma_{u,a}}(uv) &= ag(uv)\delta^2(ag(uv))^{-1} = ag(uv)(ag(uv))^{-1} \\ &= ag(uv)g(uv)^{-1}a^{-1} = 1_\Gamma. \end{aligned}$$

We shall now generalize this to the largest extent possible starting with the connected case. Recall that a *rooted tree* is a tree T with an indicated vertex $u = \mathrm{root}(T)$. We say that a vertex v is *odd* (*even*) with respect to u if the path in T between u and v is of odd (even) length. Vertices that are both even (with respect to u), or that are both odd (with respect to u) are said to be of the same *parity*. We shall use $\mathrm{odd}(T)$ and $\mathrm{even}(T)$ to refer to the sets of odd and even vertices with respect to $\mathrm{root}(T)$ respectively.

In the following we say that T is a tree in G if T is a subgraph of G that is a tree, and similarly for rooted trees and forests.

Let $g \in \mathbf{L}_G(\Gamma, \delta)$ and let $t \in \mathbf{L}_T$ where T is a rooted tree of G. Define the selector $\sigma_{g,t}$ recursively,

$$\sigma_{g,t}(u) = \begin{cases} 1_\Gamma & \text{if } u = \mathrm{root}(T), \text{ or } u \notin V(T) \\ \delta(g(vu)^{-1}\sigma_{g,t}(v)^{-1}t(vu)) & \text{otherwise, where } v \text{ is the father of } u \text{ in } T \end{cases} \tag{5.7}$$

Define then g_t to be equal to $g^{\sigma_{g,t}}$.

The following can be now proved (see, for instance, Zaslavsky [49]).

Lemma 5.18
Let $g \in \mathbf{L}_G$ and let $t \in \mathbf{L}_T$ where T is a (rooted) tree in G. For all $e \in E(T)$,

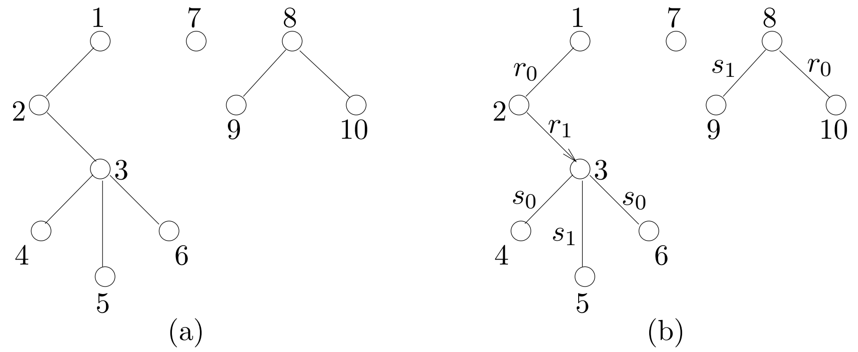


Figure 5.4: The forest F and $f \in \mathbf{L}_F(S_3, {}^{-1})$

$g_t(e) = t(e)$.
Proof:
Let $vu \in E(T)$, where v is the father of u. We compute

$$\begin{aligned} g_t(vu) &= g^{\sigma_{g,t}}(vu) = \sigma_{g,t}(v)g(vu)\delta(\sigma_{g,t}(u)) \\ &= \sigma_{g,t}(v)g(vu)\delta(\delta(g(vu)^{-1}\sigma_{g,t}(v)^{-1}t(vu))) \\ &= \sigma_{g,t}(v)g(vu)g(vu)^{-1}\sigma_{g,t}(v)^{-1}t(vu) \\ &= \sigma_{g,t}(v)\sigma_{g,t}(v)^{-1}t(vu) = t(vu)\ . \end{aligned}$$

This completes the proof. □

If t has 1_Γ as its only gain, then we write g_T instead of g_t. If T is a spanning tree of G, then we call g_T the T*-canonical* (Γ, δ)*-gain graph* of g. We sometimes say that T is 1_Γ*-labelled* in g_T. In the following, we will simply write g_T^σ for $(g_T)^\sigma$.

Lemma 5.18 can be generalized to acyclic graphs straightforwardly: simply apply the above to the rooted trees the forest consists of. Of course, a rooted forest will have a set of roots: one for each component.

Applying the method to each tree separately will yield a number of independent selectors that can be combined into one single selector by selector composition. Note that the selectors do not interfere, because no two of them select a nonidentity in the same vertex.

Corollary 5.19
Let $g \in \mathbf{L}_G$ and let $f \in \mathbf{L}_F$ where F is an acyclic subgraph of G. There exists a $g_f \in \mathbf{L}_F$ such that $g_f(e) = f(e)$ for all $e \in E(F)$.

Example 5.20
The rooted forest F of Figure 5.4(a) consists of trees T_1, T_7 and T_8, where for each T_i, the vertex i is the designated root. Using F as underlying graph, we have depicted $f \in \mathbf{L}_F(S_3, {}^{-1})$ in Figure 5.4(b). To prevent cluttering, we have included only one edge between any pair of vertices if the labels of the edges between them differ. Hence, we have not included the edge from 3 to 2, but since we know the anti-involution we know its label should be ${r_1}^{-1} = r_2$. The $(S_3, {}^{-1})$-gain graphs t_i are the components of f corresponding to T_i, where $i = 1, 7, 8$.

Let G be the graph of Figure 5.5(a) and let $g \in \mathbf{L}_G(S_3, {}^{-1})$ be the gain graph of Figure 5.5(b). Note that F is *not* a spanning forest of G since it has more components than G does.

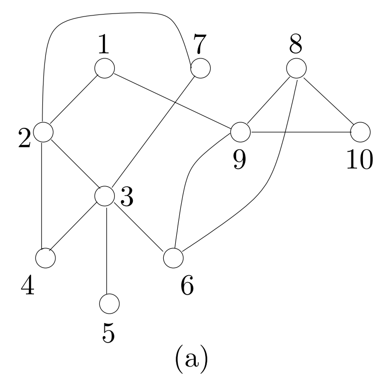


Figure 5.5: The graph G and $g \in \mathbf{L}_G(S_3, {}^{-1})$

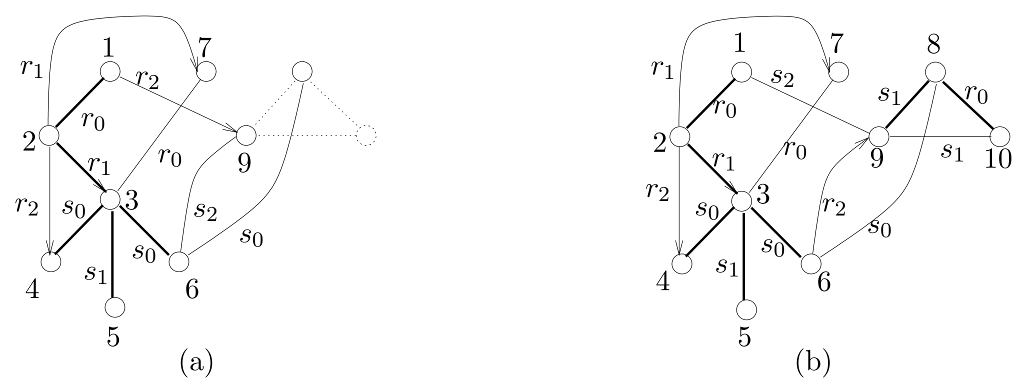


Figure 5.6: g_{t_1} and g_f respectively

In Figure 5.6(a) we find g_{t_1}. Note that the parts that are not incident with t_1 have been slightly obscured. The selector σ_{g,t_1} selects everywhere r_0 with these exceptions: $\sigma_{g,t_1}(2) = s_0$, $\sigma_{g,t_1}(3) = r_1 = \sigma_{g,t_1}(5)$, $\sigma_{g,t_1}(4) = s_2$ and $\sigma_{g,t_1}(6) = r_2$.

The selector σ_{g,t_8} selects $1_\Gamma = r_0$ in every vertex except 9 where it selects s_1. The selector σ_{g,t_7} is simply the identity selector.

In Figure 5.6(b) we present the (Γ, δ)-gain graph g_f. Note that it can be found by applying the selectors σ_{g,t_1}, σ_{g,t_7} and σ_{g,t_8} in any order; that we can do this in any order follows from the fact that no two selectors select a nonidentity in the same vertex. $\diamond$

Chapter 6

The Sizes of Switching Classes

This chapter is devoted to an investigation of the sizes of the switching classes of the graphs with skew gains. The results in this chapter are from Hage and Harju [26] unless otherwise indicated.

We shall first study this problem for complete graphs $G = (V, E_2(V))$. In the last section we reduce the general case where G is not bipartite to the complete case – the bipartite case is solved differently.

If G is complete, it turns out that, when (Γ, δ) is abelian, the cardinalities are the same for each switching class $\mathcal{G}$ with a given domain V, while in the nonabelian case the cardinality of each $\mathcal{G}$ depends (only) on the content number $c(\mathcal{G})$, which is counted using the set of elements of Γ that appear in an $h \in \mathcal{G}$ that has a horizon. Such an h exists by Lemma 5.18, and $c(\mathcal{G})$ turns out be independent of the choice of h and horizon u. The formula will then read

$$|\mathcal{G}| = k^{n-1} \cdot k/c(\mathcal{G}) \ ,$$

where k is the order of the (finite) group Γ and n is the number of elements of the domain V. Obviously if Γ is infinite then the switching classes have infinite size as well. In the above, the content number $c(\mathcal{G})$ always divides the order k of the group Γ.

6.1 Complete graphs with skew gains

It is technically convenient to assume that the domains of our switching classes contain at least three vertices. We therefore first treat the case in which the order of the underlying graph is either one or two. The following simple result holds for all groups.

Lemma 6.1
Let a set V be such that $1 \leq |V| \leq 2$ and let G be any graph on V. For all groups Γ, anti-involutions δ on Γ, and $g \in \mathbf{L}_G(\Gamma, \delta)$, we have $[g] = \mathbf{L}_G(\Gamma, \delta)$.

In the remainder of this chapter assume that $|V| \geq 3$ and Γ is finite.

Recall from Chapter 5 that since $\mathbf{S} = \mathbf{S}(V, \Gamma)$ acts on $\mathbf{L}_G$, then for all $g \in \mathbf{L}_G$,

$$|\mathbf{S}| = |\mathrm{Stab}(g)| \cdot |[g]| \ , \tag{6.1}$$

where $\mathrm{Stab}(g) = \{\sigma \in \mathbf{S} \mid g^\sigma = g\}$ is the stabilizer of $g \in \mathbf{S}$, and $[g]$ is the orbit containing g.

If $k = |\Gamma|$, (6.1) can be rewritten as

$$k^n = |\{\sigma \mid g = g^\sigma\}| \cdot |[g]| \ . \tag{6.2}$$

Note that (6.2) is independent of the chosen representative g of the switching class.

We are interested in determining the size of the stabilizer $\mathrm{Stab}(g)$, because this determines by (6.2) the remaining unknown, the size of the switching class.

Example 6.2
In this example we consider $g \in \mathbf{L}_G(S_3, f_1)$ of Figure 6.1(b). As in Example 5.6 we consider only selectors that leave the edges on the path $(1, 2, 3, 4)$ intact. This happens if for vertices i and $i+1$, $1 \leq i \leq 3$, $\sigma(i) = \delta(\sigma(i+1))^{-1}$. In other words, the selector is determined by the value selected in, say 1. As in Example 5.6 we denote with σ_a the selectors that select $a \in \Gamma$ in 1 and are of the type just described.

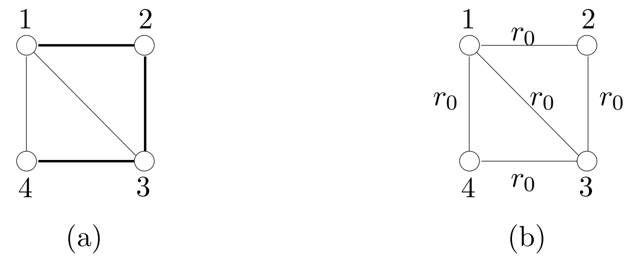


Figure 6.1: The graph G and $g \in \mathbf{L}_G(S_3, f_1)$

The selectors σ_{r_0} – the identity selector – and σ_{s_2} map g into itself. The theory in this chapter will help us to determine that these are the only two selectors that leave g unchanged. In other words $\mathrm{Stab}(g) = \{\sigma_{r_0}, \sigma_{s_2}\}$.

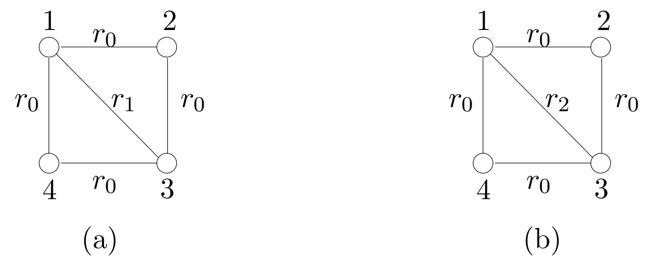


Figure 6.2: The skew gain graphs $g^{\sigma_{r_2}} = g^{\sigma_{s_0}}$ and $g^{\sigma_{r_1}} = g^{\sigma_{s_1}}$

The selectors σ_{r_2} and σ_{s_0} map g into the skew gain graph of Figure 6.2(a) and the remaining two σ_{r_1} and σ_{s_1} transform g into Figure 6.2(b). $\diamond$

For the rest of this and the following section the underlying graph will be complete, meaning that $E = E_2(V)$. (In the last section of this chapter we give a general formula for determining the sizes of the switching classes $\mathcal{G}$ of arbitrary (Γ, δ)-gain graphs.) Let $\mathbf{L}_K = \mathbf{L}_K(\Gamma, \delta)$ where K is the complete graph on V.

To determine directly the size of the stabilizer of an arbitrary g seems to be a rather complicated task. Fortunately we can, by Lemma 5.18, reduce this problem to determining the size of $\mathrm{Stab}(g)$ for a representative g of the switching class that has a horizon u. The horizons allow us also to reduce the problem further to the

δ-centralizers of $A(g-u)$, and in this way the problem is simply to determine the order of a certain subgroup of the group Γ of gains.

We adopt first some notations. Let $g \in \mathbf{L}_K$ be such that it has a horizon u. Recall that by $A(g-u)$ we denote the set of gains that occur in the subgraph of g that results when the horizon u is removed.

For any given $a \in \Gamma$, the δ-*centralizer* of a is defined as the set

$$C_a^\delta = \{x \in \Gamma \mid \delta(x)a = ax^{-1}\} \ .$$

Define

$$C_u^\delta(g) = \{x \in \Gamma \mid \delta(x)a = ax^{-1} \text{ for all } a \in A(g-u)\} = \bigcap_{a \in A(g-u)} C_a^\delta \ .$$

Lemma 6.3 [Hage and Harju [23]]

i. For $a \in \Gamma$, C_a^δ is a subgroup of Γ.

ii. For each $a \in \Gamma$, $C_a^\delta = C_{\delta(a)}^\delta$ and $C_{a^{-1}}^\delta = \{\delta(x) \mid x \in C_a^\delta\}$.

iii. For all $g \in \mathbf{L}_K$ with a horizon u, $C_u^\delta(g)$ is a subgroup of Γ.

iv. If Γ has a nontrivial centre and $\delta \neq id$, then $C_u^\delta(g)$ is a nontrivial subgroup of Γ.

Proof:
For (i), let $x, y \in C_a^\delta$. Then $\delta(xy)a = \delta(y)\delta(x)a = \delta(y)ax^{-1} = ay^{-1}x^{-1} = a(xy)^{-1}$. Clearly, $1_\Gamma \in C_a^\delta$ (so it is not empty), and if $x \in C_a^\delta$, then $x^{-1} \in C_a^\delta$, because

$$\begin{aligned} \delta(x)a = ax^{-1} \quad &\iff (\delta(x)a)^{-1} = (ax^{-1})^{-1} \\ &\iff a^{-1}\delta(x^{-1}) = xa^{-1} \iff \delta(x^{-1})a = ax \ . \end{aligned}$$

For (ii), if $x \in C_a^\delta$, then also $x^{-1} \in C_a^\delta$, and thus $\delta(x)^{-1}a = ax$. Taking anti-involutions on both sides, we obtain $\delta(a)x^{-1} = \delta(x)\delta(a)$, which shows that $x \in C_{\delta(a)}^\delta$. By symmetry, we get $C_a^\delta = C_{\delta(a)}^\delta$.

If $x \in C_a^\delta$, then $xa^{-1} = (ax^{-1})^{-1} = (\delta(x)a)^{-1} = a^{-1}\delta(x)^{-1}$, and therefore $\delta(x) \in C_{a^{-1}}^\delta$. Again, the claim follows by symmetry. Claim (iii) follows from the observation that $C_u^\delta(g)$ is the intersection of all C_a^δ for $a \in A(g-u)$.

The last claim follows from Theorem 5.11, since if δ is not the identity on the elements of $\mathrm{Z}(\Gamma)$, then there exists an element $x \in \mathrm{Z}(\Gamma)$ such that $\delta(x) = x^{-1}$ with $x \neq 1_\Gamma$, so clearly $x \in C_u^\delta(g)$. □

Lemma 6.4
For $a \in \Gamma$, C_a^δ is trivial if and only if Γ is an abelian group of odd order and δ is its identity function.
Proof:
In order to show the claim, let $\alpha : \Gamma \to \Gamma$ be the function defined by

$$\alpha(x) = \delta(x)ax \ .$$

If $\alpha(x) = \alpha(y)$ for some $x, y \in \Gamma$, then $\delta(x)ax = \delta(y)ay$ and so $\delta(y)^{-1}\delta(x)a = a(xy^{-1})^{-1}$. It follows that $xy^{-1} \in C_a^\delta$. Now let $x \in C_a^\delta, x \neq 1_\Gamma$. Then

$$\delta(x)a = ax^{-1} \iff \delta(x)ax = a \iff \alpha(x) = a \ ,$$

but also $\alpha(1_\Gamma) = a$ so α is not injective.

Therefore C_a^δ is trivial if and only if α is an injective function, that is, a bijection on Γ. However,

$$\alpha(a^{-1}) = \delta(a^{-1}) = \alpha(\delta(a^{-1})) \ ,$$

and thus, if α is a bijection, then $\delta(a) = a$, which, in turn, implies that for all $x \in \Gamma$,

$$\delta(\alpha(x)) = \delta(\delta(x)ax) = \delta(x)\delta(a)x = \delta(x)ax = \alpha(x) \ .$$

Therefore, if α is a bijection, then $\delta(y) = y$ for all $y \in \Gamma$, and Γ is abelian and of odd order by Lemma 5.10. □

We say that a selector $\sigma : V \to \Gamma$ is *constant below* u, if

$$\sigma(v) = \delta(\sigma(u))^{-1} \qquad \text{for all } v \in V - \{u\} \ . \tag{6.3}$$

Note that the selectors $\sigma_{u,a}$ for $a \in \Gamma$ defined in Section 5.4 generalize the selectors that are constant below u, take $g(uv) = 1_\Gamma$.

Lemma 6.5
Let g be a (Γ, δ)-gain graph with a horizon u. Then $g = g^\sigma$ if and only if σ is constant below u and $\sigma(u) \in C_u^\delta(g)$.
Proof:
Assume that $g^\sigma = g$ and let $a \in A(g - u)$. Let $vw \in E_2(V - \{u\})$ be such that $g(vw) = a$. We have $g^\sigma(uv) = \sigma(u)\delta(\sigma(v)) = 1_\Gamma$, hence σ is a constant below u. Also, $g(vw) = g^\sigma(vw)$, and thus $a = \sigma(v)a\delta(\sigma(w))$, yielding, by the first part of the claim, that $a = \delta(\sigma(u))^{-1}a\sigma(u)^{-1}$. The second claim follows from this.

For the converse, let σ be constant below u and $\sigma(u) \in C_u^\delta(g)$. For $v \in V - u$, $g^\sigma(uv) = \sigma(u)g(uv)\delta(\delta(\sigma(u))^{-1}) = \sigma(u)g(uv)\sigma(u)^{-1} = \sigma(u)\sigma(u)^{-1} = 1_\Gamma = g(uv)$. For $v, w \in V - u$ with $v \neq w$ and setting $a = g(vw)$, we have $g^\sigma(vw) = \sigma(v)a\delta(\sigma(w)) = \delta(\sigma(u))^{-1}a\delta(\delta(\sigma(u))^{-1}) = \delta(\sigma(u))^{-1}a\sigma(u)^{-1} = a$ because $\sigma(u) \in C_u^\delta(g)$. □

We show then that the set $C_u^\delta(g)$ is independent up to conjugacy of the choice of the horizon u.

Theorem 6.6 [Hage and Harju [23]]
If $h = g^\sigma$, where g has horizon u and h has horizon v such that $u \neq v$, then

$$C_v^\delta(h) = \{\sigma(v)^{-1}\delta(x)^{-1}\sigma(v) \mid x \in C_u^\delta(g)\} \ .$$

In particular, the automorphism ψ of Γ, defined by $\psi(z) = \sigma(v)^{-1}\delta(z)^{-1}\sigma(v)$, maps $C_u^\delta(g)$ onto $C_v^\delta(h)$.
Proof:
For each (Γ, δ)-gain graph g with a horizon u define $\alpha_g : \mathrm{Stab}(g) \to \Gamma$ by

$$\alpha_g(\tau) = \tau(u) \quad \text{ for all } \tau \in \mathrm{Stab}(g) \ .$$

By Lemma 6.5, we have $\alpha_g(\tau) \in C_u^\delta(g)$. By the same lemma, $\tau \in \mathrm{Stab}(g)$ implies that τ is constant below u, and thus τ is uniquely determined by the value $\tau(u)$. This shows that α_g is injective. That α_g maps $\mathrm{Stab}(g)$ onto $C_u^\delta(g)$ follows again from Lemma 6.5. We have also $\alpha_g(\tau_1\tau_2) = (\tau_1\tau_2)(u) = \tau_1(u)\tau_2(u) = \alpha_g(\tau_1)\alpha_g(\tau_2)$, and thus $\alpha_g : \mathrm{Stab}(g) \to C_u^\delta(g)$ is an isomorphism.

Suppose then that $h = g^\sigma$, and thus also that $g = h^{\sigma^{-1}}$. By the above, $C_u^\delta(g)$ and $C_v^\delta(h)$ are isomorphic.

We have

$$\mathrm{Stab}(h) = \{\sigma^{-1}\tau\sigma \mid \tau \in \mathrm{Stab}(g)\} \enspace ,$$

since

$$g^\tau = g \iff h^{\sigma^{-1}\tau} = h^{\sigma^{-1}} \iff h^{\sigma^{-1}\tau\sigma} = h \enspace .$$

In particular, the mapping $\mathrm{Stab}(g) \to \mathrm{Stab}(h)$ with $\tau \mapsto \sigma^{-1}\tau\sigma$ is an isomorphism, since
it is an (inner) automorphism of the group of selectors.

Let $x \in C_u^\delta(g)$, and let τ be a selector which is constant below the horizon u of g such that $\tau(u) = x$. Because $\tau(v) = \delta(x)^{-1}$,

$$\psi(x) = \sigma^{-1}(v)\tau(v)\sigma(v) = \sigma^{-1}\tau\sigma(v) \in \mathrm{Stab}(h) \enspace .$$

Clearly, ψ is an automorphism of Γ, and therefore the claim follows. □

The above proof uses the assumption $u \neq v$ only in determining the value $\tau(v) = \delta(x)^{-1}$. If here $u = v$, then, by the assumption, $\tau(v) = x$ and we obtain the following result.

Theorem 6.7 [Hage and Harju [23]]
If $h = g^\sigma$, where both g and h have the same horizon u, then

$$C_u^\delta(h) = \{\sigma(u)^{-1}x\sigma(u) \mid x \in C_u^\delta(g)\} \enspace .$$

In particular, the automorphism ψ of Γ, defined by $\psi(z) = \sigma(u)^{-1}z\sigma(u)$, maps $C_u^\delta(g)$ onto $C_u^\delta(h)$.

It follows from these results that if g and h are two complete (Γ,δ)-gain graphs with horizons u and v respectively, then $[g] = [h]$ implies that $C_u^\delta(g) \cong C_v^\delta(h)$, since the stabilizers $\mathrm{Stab}(g)$ and $\mathrm{Stab}(h)$ are conjugate. Therefore, for a switching class $\mathcal{G}$, the integer

$$c(\mathcal{G}) = |C_u^\delta(g)| \enspace ,$$

called the *content number* of $\mathcal{G}$, is independent of the choice of $u \in V$ and of $g \in \mathcal{G}$ having a horizon u.

We have by (6.2) and Theorem 6.6 that

Theorem 6.8
For each switching class $\mathcal{G}$ of (Γ,δ)-gain graphs with $|\Gamma| = k$ and $|V| = n$,

$$|\mathcal{G}| = k^n/c(\mathcal{G})$$

and $c(\mathcal{G})$ divides k.

In particular, the size of every switching class is a multiple of the order of the group.

Also, the content number $c(\mathcal{G})$ is partly independent of $\mathcal{G}$ in the sense that it becomes determined by the *set* $A(g - u)$ of gains (where $g \in \mathcal{G}$ has a horizon u). Moreover, since $c(\mathcal{G})$ is independent of the choice of g and u, we get the following result.

Corollary 6.9
Let $\mathcal{G}$ and $\mathcal{H}$ be switching classes of (Γ, δ)-gain graphs (having possibly domains of different sizes) and let $\mathcal{G} = [g]$ and $\mathcal{H} = [h]$, where g and h have horizons u and v, respectively. If $g - u$ and $h - v$ have the same sets of gains, $A(g-u) = A(h-v)$, then $c(\mathcal{G}) = c(\mathcal{H})$.

We conclude this section with some details of the switching classes. For $u \in V$, the *u-refinement* generated by a (Γ, δ)-gain graph h is

$$\langle h \rangle_u = \{h^\sigma \mid \sigma(u) = 1_\Gamma\} \ .$$

Note that the selectors whose value in a fixed vertex u is 1_Γ form a subgroup of $\mathbf{S}$. Hence, they induce a partitioning of the switching classes into u-refinements, which explains the name refinement. The selectors $\sigma_{u,a}$, defined in Section 5.4, then take us from one u-refinement to another as we shall show in the following theorem; we shall exploit this again in Chapter 7.

Theorem 6.10
Let $g \in \mathbf{L}_K$ have a horizon u and let $\mathcal{T}$ be a transversal of the left cosets of $C_u^\delta(g)$. Then $[g] = \bigcup_{a \in \mathcal{T}} \langle g^{\sigma_{u,a}} \rangle_u$. Moreover, all $g^{\sigma_{u,a}}$ have a horizon u, and the refinements $\langle g^{\sigma_{u,a}} \rangle_u$ for $a \in \mathcal{T}$ are disjoint.
Proof:
Indeed, by Lemma 6.5 and (5.6), $g^{\sigma_{u,a}} = g^{\sigma_{u,b}}$ if and only if $b^{-1}a \in C_u^\delta(g)$, from which it follows that all $g^{\sigma_{u,a}}$ are different. By (5.6) it is clear that u is a horizon of each $g^{\sigma_{u,a}}$. Now, $g^{\sigma_{u,a}}$ is the only element of $\langle g^{\sigma_{u,a}} \rangle_u$ that has u as its horizon: the only selector that selects 1_Γ in u and keeps the edges between u and the other vertices 1_Γ-labelled is the identity selector. Hence the u-refinements $\langle g^{\sigma_{u,a}} \rangle_u$ for $a \in \mathcal{T}$ are disjoint.

For $h = g^\tau$, let $b = \tau(u)$. Now $\tau = (\tau {\sigma_{u,b}}^{-1})\sigma_{u,b}$, where, again by (5.6), $\tau {\sigma_{u,b}}^{-1}(u) = 1_\Gamma$. This shows that $h \in \langle g^{\sigma_{u,b}} \rangle_u$. The claim follows, since there exists an $a \in \mathcal{T}$ such that $g^{\sigma_{u,a}} = g^{\sigma_{u,b}}$ by the first part of the proof. □

Recall that g^{-1} is the (Γ, δ)-gain graph defined by $g^{-1}(uv) = g(uv)^{-1}$ for all $uv \in E_2(V)$. Note that it can be true that $[g] \neq [g^{-1}]$. However, we show next that the switching classes generated by g and g^{-1} are of the same size.

Theorem 6.11
For any (Γ, δ)-gain graph g with a horizon u, $C_u^\delta(g) = \delta(C_u^\delta(g^{-1}))$. Consequently, $|[g]| = |[g^{-1}]|$.
Proof:
Because $C_a^\delta = \delta(C_{a^{-1}}^\delta)$, and δ is a bijection, the claim follows from the definition of $C_u^\delta(g)$ and the facts that g^{-1} has u as its horizon and $A^u(g^{-1}) = \{a^{-1} \mid a \in A^u(g)\}$.
□

6.2 Improvements in some special cases

Throughout this section we let Γ be a finite group of order k and $[g]$ a switching class where g is a (Γ, δ)-gain graph on a complete graph, which has a horizon u and a domain V of size n.

We observe that, by the definition of $C_u^\delta(g)$, if $1_\Gamma \in A(g-u)$, then necessarily $\delta(x) = x^{-1}$ and $xa = ax$ for all $x \in C_u^\delta(g)$ and $a \in A(g-u)$. As the next theorem states, this generalizes to a wider family of skew gain graphs.

Recall that the centralizer of a subset $A \subseteq \Gamma$ is

$$C(A) = \{x \in \Gamma \mid ax = xa \text{ for all } a \in A\} .$$

Denote also

$$I^\delta(\Gamma) = \{x \in \Gamma \mid \delta(x) = x^{-1}\} .$$

We observe that always

$$I^\delta(\Gamma) \cap C(A(g-u)) \subseteq C_u^\delta(g) .$$

Theorem 6.12

i. If there exist $a, b, ab \in A(g-u)$, then $C_u^\delta(g) = I^\delta(\Gamma) \cap C(A(g-u))$.

ii. For the (Γ, δ)-gain graph g_1 that has only identity gains, $k^n = |I^\delta(\Gamma)| \cdot |[g_1]|$.

Proof:
For (i), assume that $a, b, ab \in A(g-u)$.

Thus for any $x \in C_u^\delta(g)$, $\delta(x)b = bx^{-1}$ and $\delta(x)ab = abx^{-1}$. Therefore $\delta(x)ab = a\delta(x)b$, whence $\delta(x)a = a\delta(x)$. But also $\delta(x)a = ax^{-1}$, so $\delta(x) = x^{-1}$. The claim follows.

Since g_1 has the identity as its only gain and $C(\{1_\Gamma\}) = \Gamma$, $C_u^\delta(g_1) = I^\delta(\Gamma)$ and thus $k^n = |I^\delta(\Gamma)| \cdot |[g_1]|$ and (ii) has been proven to hold. □

If a (Γ, δ)-gain graph g with a horizon u has a gain $a \in A(g-u)$ such that $a \in Z(\Gamma)$, then $\delta(x) = x^{-1}$ for all $x \in C_u^\delta(g)$; thus

Theorem 6.13
If $Z(\Gamma) \cap A(g-u) \neq \emptyset$, then $C_u^\delta(g) = I^\delta(\Gamma) \cap C(A(g-u))$.

In particular, for an abelian group Γ, $C_u^\delta(g) = I^\delta(\Gamma)$ and thus the size of the switching class does not depend on the gains of g.

For the case when δ is the inversion of Γ, we can be more specific.

Corollary 6.14
If the anti-involution is the group inversion, then $k^n = |C(A(g-u))| \cdot |[g]|$. If, moreover, $A(g-u) \subseteq \mathrm{Z}(\Gamma)$ then $|[g]| = k^{n-1}$.

Example 6.15
(1) Suppose first that the anti-involution of S_3 is the group inversion.

We can determine that

$$C_{r_0}^\delta = S_3, \quad C_{r_1}^\delta = \{r_0, r_1, r_2\} = C_{r_2}^\delta,$$
$$C_{s_0}^\delta = \{r_0, s_0\},\ C_{s_1}^\delta = \{r_0, s_1\}, \quad C_{s_2}^\delta = \{r_0, s_2\} .$$

In particular, if the domain V has three elements and g has a horizon u, then

$$|[g]| = \begin{cases} 36, & \text{if } g \text{ has only } r_0\text{-labels. ,} \\ 72, & \text{if } A(g-u) \subseteq \{r_1, r_2\}, \\ 108, & \text{otherwise} \quad . \end{cases}$$

(2) Let δ be the anti-involution of f_1 from the table of Example 5.1. In particular, $I^\delta = \{r_0, s_2\}$. We can now compute

$$C^\delta_{r_0} = \{r_0, s_2\}, \quad C^\delta_{r_1} = \{r_0, s_0\}, \quad C^\delta_{r_2} = \{r_0, s_1\},$$
$$C^\delta_{s_0} = \{r_0, r_1, r_2\} = C^\delta_{s_1}, \quad C^\delta_{s_2} = S_3 \ .$$

Therefore, *e.g.*, if a complete (S_3, δ)-gain graph g with a horizon u yields a nontrivial subgroup $C^\delta_u(g)$, then $A(g-u)$ can contain at most one of the gains r_0, r_1, r_2, s_2. (Note that $s_0 \in A(g-u)$ if and only if $s_1 \in A(g-u)$, since $\delta(s_1) = s_0$.) If $C^\delta_u(g)$ is the trivial subgroup of S_3, then $|[g]| = 6^n$, where $n \geq 3$ is the size of the domain. This can only happen when $n \geq 4$. ◇

6.3 The general case

In this section we consider an arbitrary (Γ, δ)-gain graph g on $G = (V, E)$. As before we denote $n = |V|$.

If G is disconnected and the connected components of g are $g_1, \ldots, g_c$, then clearly

$$|[g]| = |[g_1]| \cdot |[g_2]| \cdots |[g_c]|.$$

Thus the size of the switching class is reduced to the sizes of switching subclasses generated by the connected components.

Suppose now that G is connected, and let T be one of its spanning trees. By Lemma 5.18 there exists a $g_T \in [g]$ such that $g_T(e) = 1_\Gamma$ for all $e \in T$.

A selector is *alternating* if it is of the form

$$\sigma_{T,a}(v) = \begin{cases} a & \text{if } v \text{ is even with respect to } u, \\ \delta(a)^{-1} & \text{if } v \text{ is odd with respect to } u. \end{cases} \tag{6.4}$$

The alternating selectors generalize selectors constant below u to arbitrary trees; in the case of selectors constant below u the trees are restricted to star graphs. We generalize Lemma 6.5.

Lemma 6.16
Let T be a spanning tree of g. If $g^\sigma_T = g_T$ for a selector σ, then $\sigma = \sigma_{T,a}$ for some $a \in \Gamma$.
Proof:
Let $vw \in E(T)$. Clearly, v and w have different parity and we have

$$g^\sigma_T(vw) = \sigma(v) g_T(vw) \delta(\sigma(w)) = \sigma(v)\delta(\sigma(w)) = 1_\Gamma \iff \sigma(v) = \delta(\sigma(w))^{-1} \ .$$

The result now follows, because T is connected. □

We divide our further considerations into two parts according to whether g is bipartite or not.

Theorem 6.17
Let g be a (Γ, δ)-gain graph on a connected bipartite graph $G = (V, E)$, and let T be a spanning tree of G. Then $\mathrm{Stab}(g) \cong C(A(g_T))$. As a consequence,

$$|[g]| = k^n / |C(A(g_T))| \ .$$

Proof:
Let $u \in V$ be a fixed vertex. The sets of even vertices and odd vertices with respect to u form the unique bipartition of G, since G is connected.

Let $e = (vw) \in E$, where v is even and w is odd. For each selector σ of the form in (6.4), $g_T(e) = g_T^{\sigma}(e)$ holds if and only if $g_T(e) = \sigma(v)g_T(e)\delta(\sigma(w)) = \sigma(u)g_T(e)\sigma(u)^{-1}$. Therefore, $g_T = g_T^{\sigma}$ if and only if $\sigma(u) \in C(A(g_T))$.

The mapping $\alpha : \mathrm{Stab}(g_T) \to C(A(g_T))$ defined by $\alpha(\sigma) = \sigma(u)$ is easily seen to be an isomorphism. This proves the claim, since it is always the case that $\mathrm{Stab}(g) \cong \mathrm{Stab}(g_T)$. □

Example 6.18
In Example 5.6 we saw an example of a (Γ, δ)-gain graph g with an underlying graph that was bipartite. Because the group, $\mathbf{Z}_4$, is abelian we should have $|[g]| = 4^n/4$ and indeed the switching class generated by g has 64 elements: they are listed in Figure 5.3. ◇

For the non-bipartite case we prove the following

Theorem 6.19
Let g be a (Γ, δ)-gain graph on a connected graph $G = (V, E)$ that is not bipartite. Then there exists a complete (Γ, δ)-gain graph $\hat{g}$ on domain V such that $\hat{g}|_E \sim g$ and $\mathrm{Stab}(g) \cong \mathrm{Stab}(\hat{g})$; therefore $|[g]| = |[\hat{g}]|$.

Proof:
Let T be a spanning tree of G, and $u \in V$. Since G is not bipartite, there is an edge $e_0 = u_0v_0 \in E$ that connects either two odd vertices or two even vertices with respect to u. We can assume that the former case holds; otherwise we change the fixed vertex u to one of its neighbours in T. Such a change changes the parity of the vertices.

Denote $a = g_T(e_0)$ and let $e \in E_2(V)$. We define $\hat{g}$ as follows. If $e \in E$, then $\hat{g}(e) = g_T(e)$. For $e \notin E$, there are three cases: if the endpoints of e are

- of different parity, then put $\hat{g}(e) = 1_\Gamma$,
- both odd, then put $\hat{g}(e) = a$ and $\hat{g}(e^{-1}) = \delta(a)$,
- both even, then put $\hat{g}(e) = a^{-1}$ and $\hat{g}(e^{-1}) = \delta(a)^{-1}$.

(In the last two cases the choice of priority between e and e^{-1} is arbitrary.)

Since g_T is a subgraph of $\hat{g}$ (that is, G is a subgraph of $\hat{G} = (V, E_2(V))$ and g respects the gains of $\hat{g}$), it follows that $\mathrm{Stab}(\hat{g}) \subseteq \mathrm{Stab}(g_T)$. In words: for the selectors σ such that $\hat{g}^{\sigma} = \hat{g}$, also $g_T^{\sigma} = g_T$.

In the other direction, assume that $\sigma \in \mathrm{Stab}(g_T)$, and let $e = vw \in E_2(V)$ be any edge of $\hat{g}$. If $e \in E$ then $\hat{g}(e) = g_T(e)$, and therefore also $\hat{g}^{\sigma}(e) = \hat{g}(e)$. Suppose that $e \notin E$.

If v and w are of opposite parity (with respect to u), say v is even and w is odd, then $\hat{g}(e) = 1_\Gamma$, and $\sigma(v) = \sigma(u)$, $\sigma(w) = \delta(\sigma(u))^{-1}$ by Lemma 6.16. Now also

$$\hat{g}^{\sigma}(e) = \sigma(v)\hat{g}(e)\delta(\sigma(w)) = \sigma(u)\hat{g}(e)\delta(\delta(\sigma(u)))^{-1} = \sigma(u)\hat{g}(e)\sigma(u)^{-1} = 1_\Gamma \ .$$

If both v and w are odd, then $\hat{g}(e) = a$ or $\hat{g}(e) = \delta(a)$. In this case, $\sigma(v) = \sigma(u_0) = \sigma(v_0) = \sigma(w)$. Now, if $\hat{g}(e) = a$ $(= g_T(e_0))$, then

$$\hat{g}^{\sigma}(e) = \sigma(v)\hat{g}(e)\delta(\sigma(w)) = \sigma(u_0)g_T(e_0)\delta(\sigma(v_0)) = g_T^{\sigma}(e_0) = g_T(e_0) = \hat{g}(e).$$

If $\hat{g}(e) = \delta(a)$, then

$$\begin{aligned}\hat{g}^\sigma(e) &= \sigma(v)\hat{g}(e)\delta(\sigma(w)) = \sigma(v_0)\hat{g}(e)\delta(\sigma(u_0)) \\ &= \delta(\sigma(u_0)g_T(e_0)\delta(\sigma(v_0))) = \delta(g_T^\sigma(e_0)) = \delta(g_T(e_0)) = \hat{g}(e)\end{aligned}$$

as required.

The final case is when both v and w are even, in which case $\hat{g}(e) = a^{-1}$ or $\hat{g}(e) = \delta(a)^{-1}$. In this case, $\sigma(v) = \sigma(w) = \delta(\sigma(v_0))^{-1} = \delta(\sigma(u_0))^{-1}$. As above

$$\begin{aligned}\hat{g}^\sigma(e) &= \sigma(v)\hat{g}(e)\delta(\sigma(w)) = \delta(\sigma(v_0))^{-1}\hat{g}(e)\delta(\delta(\sigma(u_0))^{-1}) \\ &= \delta(\sigma(v_0))^{-1}\hat{g}(e)\sigma(u_0)^{-1} = \left(\sigma(u_0)\hat{g}(e)^{-1}\delta(\sigma(v_0))\right)^{-1} \\ &= (\sigma(u_0)g_T(e_0)\delta(\sigma(v_0)))^{-1} = g_T^\sigma(e_0)^{-1} = g_T(e_0)^{-1} = \hat{g}(e).\end{aligned}$$

Finally, if $\hat{g}(e) = \delta(a)^{-1}$, then

$$\begin{aligned}\hat{g}^\sigma(e) &= \sigma(v)\hat{g}(e)\delta(\sigma(w)) = \delta(\sigma(u_0))^{-1}\hat{g}(e)\delta(\delta(\sigma(v_0))^{-1}) = \delta(\sigma(u_0))^{-1}\hat{g}(e)\sigma(v_0)^{-1} \\ &= \left(\sigma(v_0)\hat{g}(e)^{-1}\delta(\sigma(u_0))\right)^{-1} = \delta\left(\sigma(u_0)\delta(\hat{g}(e))^{-1}\delta(\sigma(v_0))\right)^{-1} \\ &= \delta(\sigma(u_0)g_T(e_0)\delta(\sigma(v_0)))^{-1} = \delta(g_T^\sigma(e_0))^{-1} = \delta(g_T(e_0))^{-1} = \hat{g}(e).\end{aligned}$$

This shows that $\sigma \in \mathrm{Stab}(\hat{g})$. Thus, $\mathrm{Stab}(\hat{g}) = \mathrm{Stab}(g_T) \cong \mathrm{Stab}(g)$, and hence the proof is completed. □

We conclude this chapter with an extensive example.

Example 6.20

In this example, $\Gamma = S_3$ and $\delta = f_1$ of Example 5.1. Let g be the (Γ, δ)-gain graph of Figure 6.1(b) on the graph G of Figure 6.1(a). The spanning tree T is indicated in G by the bold edges. Note that T is r_0-labelled in g. We choose 2 for the root of T to make sure that the – in this case only – edge between vertices of the same parity is between odd vertices, 1 and 3, as in the theorem.

The sets C_a^δ for $a \in S_3$ were given in Example 6.15. The (Γ, δ)-gain graph $\hat{g}$ is obtained from g by adding the missing edges $(2, 4)$ and $(4, 2)$ and labelling it with the element that labels the edge between the odd vertices (with respect to 2), 1 and 3. The edge $(2, 4)$ obtains the label ${r_0}^{-1} = r_0$ and the edge $(4, 2)$ is labelled with $\delta(r_0)^{-1} = r_0$.

In this way we obtain $\hat{g}$ of Figure 6.3(a). Obviously $A(\hat{g}) = \{r_0\}$ and as a consequence $C_u^\delta(\hat{g}) = C_{r_0}^\delta = \{r_0, s_2\}$. By Theorem 6.8, $|[\hat{g}]| = k^n/2 = 3 \cdot 6^3$. By Theorem 6.19, also $|[g]| = 3 \cdot 6^3$. Note that this corresponds exactly to Example 6.2 where we found that $[g]$ contained three (Γ, δ)-gain graphs in which T was 1_Γ-labelled. (They can be found in Figure 6.1(b) and Figure 6.2(a) and (b).)

Another way of looking at it is to say that the alternating selectors σ_{T,r_0} and σ_{T,s_2} are the only ones that map g into itself, a fact that was already demonstrated in Example 6.2.

If we take g' to be the (Γ, δ)-gain graph of Figure 6.3(b) we again have to add the edges $(2, 4)$ and $(4, 2)$. Now the label a of Theorem 6.19 is arbitrarily chosen to be that of $(1, 3)$ (the edge $(3, 1)$ would be the alternative), which is s_0. Again arbitrarily, we choose to label $(2, 4)$ with ${s_0}^{-1} = s_0$ (the element $\delta(s_0)^{-1} = s_1$ would be the alternative) and consequently $(4, 2)$ gets the label $s_1 = \delta(s_0)^{-1}$ as in Figure 6.3(c). Now $A(\hat{g}') = \{s_2, s_0, s_1\}$ and $C_u^\delta(g) = C_{s_0}^\delta \cap C_{s_1}^\delta \cap C_{s_2}^\delta = \{r_0, r_1, r_2\}$ and consequently $|[\hat{g}']| = 2 \cdot 6^3 = |[g']|$. ◇

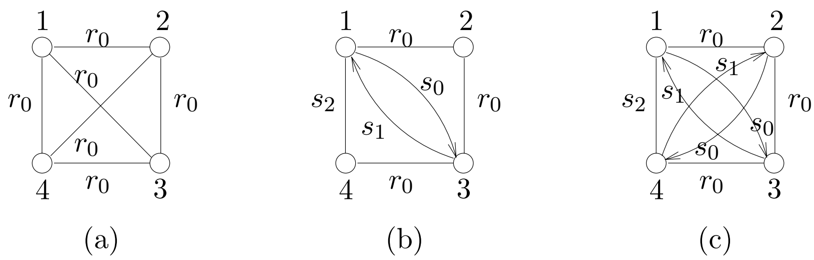


Figure 6.3:

Chapter 7

The Membership Problem

This chapter considers the membership problem or equivalence problem for switching classes of skew gain graphs, i.e., given two skew gain graphs $g, h \in \mathbf{L}_G(\Gamma, \delta)$, decide whether $h \in [g]$. This problem is very well motivated from the point of view of dynamic labelled 2-structures (see [16]). From the point of view of the networks of the introduction the question whether or not a network can end up in a specific configuration is certainly a central question. The material in this chapter is from Hage [21] unless otherwise indicated.

We start with a general treatment of the membership problem that reduces its complexity. Based on this we construct an efficient algorithm of time complexity $\mathcal{O}(k|V(G)|^2)$ for finite groups of order k. Note however that this complexity estimate is not always optimal since if G is acyclic, then Corollary 5.19 implies that the answer is always *yes* and the complexity is then, of course, much lower.

We also look at various optimizations that can be made for particular kinds of groups, anti-involutions and underlying graphs. In this way we get more efficient algorithms for abelian groups when the anti-involution is the group inversion or the underlying graph is bipartite. Also, if the switching class contains a skew gain graph with only gains from the centre of the group, then we can reduce the membership problem to an elegant problem on anti-involutions, which makes this problem tractable for some infinite groups. This reduction takes $\mathcal{O}(|E(G)|)$ time.

Finally, we prove that in general there are groups that yield an undecidable membership problem already for very simple underlying graphs. In particular, these groups have an undecidable conjugacy problem.

7.1 The general problem of membership

Let $g, h \in \mathbf{L}_G(\Gamma, \delta)$. If G consists of connected components G_i, for $1 \leq i \leq c$ (inducing in this way components g_i and h_i of g and h respectively), then we can reduce the problem $g \in [h]$ to the connected case: $g \in [h]$ if and only if $g_i \in [h_i]$ for $1 \leq i \leq c$. Therefore we may concentrate on connected graphs G.

Let $G = (V, E)$ be connected and let T be a spanning tree of G. By Lemma 5.18 there exists a T-canonical (Γ, δ)-gain graph $g_T \in [g]$.

We recall some material essential for this chapter from Chapter 5 and 6. For a tree T we defined in (6.4) the selector $\sigma = \sigma_{T,a}$ alternating in T: for every edge uv in the tree, $\sigma(u) = \delta(\sigma(v))^{-1}$ and $\sigma(\mathrm{root}(T)) = a$. The essential property of alternating selectors is that switching a T-canonical (Γ, δ)-gain graph by an alternating selector yields a T-canonical (Γ, δ)-gain graph (Lemma 6.16).

For $u \in V$, a selector σ with $\sigma(u) = 1_\Gamma$ is a *u-selector*. Let $u \in V$ be fixed. The u-selectors form a subgroup of the group of selectors and, as a consequence, they induce a partitioning of the switching classes into (possibly) smaller classes, see Chapter 6. Note that $\sigma_{g,t}$, defined by (5.7), is a root(T)-selector.

The u-refinement $\langle g \rangle_u$ generated by g contains exactly one (Γ, δ)-gain graph with T 1_Γ-labelled, which is g_T. This follows from the fact that the only selector that is both alternating and a u-selector is the trivial selector.

A property of a u-refinement is that it contains as many elements as there are u-selectors, i.e., each u-selector maps to a different (Γ, δ)-gain graph.

Lemma 7.1
For each $g \in \mathbf{L}_G$ and vertex u of G, $|\langle g \rangle_u| = |\{\sigma \mid \sigma \text{ is a } u\text{-selector}\}|$.
Proof:
Let σ and τ be different u-selectors. Because they differ on at least one vertex, say z, and they correspond on the value selected in u, there must be an edge vw on the path between u and z such that they equal on v and differ on w. Consequently $g^\sigma(vw) \neq g^\tau(vw)$. □

Corollary 7.2
If Γ is finite, of order k and G is of order n, then $|\langle g \rangle_u| = k^{n-1}$ for $u \in V(G)$.

Summarizing, we have found that every switching class on a connected graph consists of a number of u-refinements that all have the same size. We also know that each such equivalence class is generated by a T-canonical (Γ, δ)-gain graph. The only remaining question is to decide which u-refinements constitute a switching class or, equivalently, which T-canonical (Γ, δ)-gain graphs belong to the same switching class.

In the following we will try to formulate an answer to the following question: given two different T-canonical (Γ, δ)-gain graphs, is there a selector that maps the one to the other? This question is simpler than the original, because we need not consider u-selectors. In fact, we need only consider alternating selectors, because of Lemma 6.16.

We introduce some definitions based on Chapter 6. We denote by

$$\mathrm{EO}_T(g) = \{a \in \Gamma \mid g(uv) = a, uv \in E(G), u \in \mathrm{even}(T), v \in \mathrm{odd}(T), uv \notin E(T)\}$$

the gains of the edges of g that are not in T and that start in an even and end in an odd vertex with respect to root(T). In a similar way we define

$$\mathrm{EE}_T(g) = \{a \in \Gamma \mid g(uv) = a, uv \in E(G), u, v \in \mathrm{even}(T)\} \text{ and}$$

$$\mathrm{OO}_T(g) = \{a \in \Gamma \mid g(uv) = a, uv \in E(G), u, v \in \mathrm{odd}(T)\} \ .$$

Further, let

$$C_T^\delta(g) = C(\mathrm{EO}_T(g_T)) \cap \mathrm{OO}_T^\delta(g_T) \cap \mathrm{EE}_T^\delta(g_T)$$

where

$$\mathrm{OO}_T^\delta(g) = \{x \in \Gamma \mid \delta(x)a = ax^{-1} \text{ for all } a \in \mathrm{OO}_T(g)\}, \text{ and}$$

$$\mathrm{EE}_T^\delta(g) = \{x \in \Gamma \mid a\delta(x) = x^{-1}a \text{ for all } a \in \mathrm{EE}_T(g)\} \ .$$

Note that $\mathrm{OO}_T^\delta(g)$ and $\mathrm{EE}_T^\delta(g)$ are subgroups of Γ. Hence, $C_T^\delta(g)$ is also a subgroup of Γ. Note also that if the anti-involution is the group inversion, then the definitions of all three sets that determine $C_T^\delta(g)$ coincide and $C_T^\delta(g)$ can be defined as $C(A(g_T - T))$; $g_T - T$ denotes the (Γ, δ)-gain graph that can be constructed from g_T by omitting the edges in T. The sets also coincide if G is bipartite, because then $\mathrm{EE}_T(g)$ and $\mathrm{OO}_T(g)$ are empty, hence $\mathrm{OO}^{\delta}_T(g) = \mathrm{EE}^{\delta}_T(g) = \Gamma$. We return to this in Section 7.3.

The following lemma generalizes Lemma 6.5.

Lemma 7.3
Let $g \in \mathbf{L}_G$ and let T be a spanning tree of G. Then $g_T = g^{\sigma}_T$ if and only if σ is alternating in T and $\sigma(\mathrm{root}(T)) \in C^{\delta}_T(g)$.
Proof:
Assume that $g_T = g^{\sigma}_T$. By Lemma 6.16, σ is alternating.

Let $vw \in E(g_T) - E(T)$ and let $a = g_T(vw)$. We must consider three cases: both v and w are even, both are odd, or one is even and the other is odd (all with respect to $r = \mathrm{root}(T)$).

We start with the case that both v and w are even, in which case $\sigma(r) = \sigma(v) = \sigma(w)$. Because $g_T(vw) = g^{\sigma}_T(vw)$, $a = \sigma(v)a\delta(\sigma(w))$. This together with the fact that σ is alternating, yields $a = \sigma(r)a\delta(\sigma(r))$. Hence, $\sigma(r)^{-1}a = a\delta(\sigma(r))$ or equivalently, $\sigma(r) \in \mathrm{EE}^{\delta}_T(g_T)$. The claim now follows.

For v and w odd, now $\sigma(v) = \sigma(w) = \delta(\sigma(r))^{-1}$, we obtain in a similar fashion $a = \delta(\sigma(r))^{-1}a\delta(\delta(\sigma(r)))^{-1} = \delta(\sigma(r))^{-1}a\sigma(r)^{-1}$ or equivalently, $\delta(\sigma(r))a = a\sigma(r)^{-1}$ and indeed $\sigma(r) \in \mathrm{OO}^{\delta}_T(g_T)$.

Finally, for v even and w odd we have $\sigma(r) = \sigma(v) = \delta(\sigma(w))^{-1}$. We obtain $a = \sigma(r)a\delta(\delta(\sigma(r)))^{-1} = \sigma(r)a\sigma(r)^{-1}$ or equivalently, $a\sigma(r) = \sigma(r)a$ and indeed $\sigma(r) \in C(\mathrm{EO}_T(g_T))$.

The converse is proved similarly in all cases. □

The following result characterizes the u-refinements that constitute a switching class, straightforwardly generalizing Theorem 6.10 from Chapter 6.

Theorem 7.4
Let T be a spanning tree of G with root $r = \mathrm{root}(T)$, and $g \in \mathbf{L}_G$ such that T is 1_Γ-labelled in g. Also, let $\mathcal{T}$ be a transversal of the left cosets of $C^{\delta}_T(g)$. Then $[g] = \bigcup_{a \in \mathcal{T}} \langle g^{\sigma_{T,a}} \rangle_r$. Moreover, all $\langle g^{\sigma_{T,a}} \rangle_r$ are disjoint.
Proof:
Indeed, $\langle g^{\sigma_{T,a}} \rangle_r = \langle g^{\sigma_{T,b}} \rangle_r$ if and only if $(g^{\sigma_{T,a}})^{\sigma_{T,b^{-1}}} = g$ if and only if $\sigma(r) = b^{-1}a \in C^{\delta}_T(g)$, by Lemma 7.3. Then $\langle g^{\sigma_{T,a}} \rangle_r$ $(a \in \mathcal{T})$ are different, because $\mathcal{T}$ is a transversal of the left cosets of $C^{\delta}_T(g)$.

For $h = g^{\tau}$, let $b = \tau(r)$. Now, $\tau = (\tau{\sigma_{T,b}}^{-1})\sigma_{T,b}$, where $\tau{\sigma_{T,b}}^{-1}(r) = 1_\Gamma$. Hence, $h \in \langle g^{\sigma_{T,b}} \rangle_r$. The claim follows, because there exists an $a \in \mathcal{T}$ such that $g^{\sigma_{T,a}} = g^{\sigma_{T,b}}$ by the first part of the proof. □

For the membership problem, the following corollary will be useful.

Corollary 7.5
Let $g, h \in \mathbf{L}_G$ and let T be a spanning tree of G. Also, let $\mathcal{T}$ be a transversal of the left cosets of $C^{\delta}_T(g)$. Then $g \in [h]$ if and only if $h_T = g^{\sigma_{T,a}}_T$ for some $a \in \mathcal{T}$.

Summarizing, given a graph G of order n and $g, h \in \mathbf{L}_G$ we can answer the question whether or not $g \in [h]$. For a finite group of order k, this question can be answered by simply applying all k^n selectors σ to g and checking whether $h = g^{\sigma}$.

Applying the theory developed above we need only apply k selectors, which are alternating. It is important to realize here that the number of selectors is independent of the order n of G. For this to work, we should first compute g_T and h_T (for some tree T) from g and h. A further saving can be made by applying only those alternating selectors that select in the root of T an element of a transversal of the left cosets of $C_T^{\delta}(g)$.

7.2 Algorithms

Now we describe an algorithm for finite groups (and, if necessary, it can be modified to work for infinite groups as long as the transversal $\mathcal{T}$, see Corollary 7.5, is finite). We only give the algorithm for connected graphs; for disconnected graphs it should be applied to each component.

Algorithm 7.6

```
SameSwitchingClass?  (g,h)
(* Here, g,h ∈ L_G(Γ,δ). *)                                        begin
    T = a spanning tree of G;
    Compute g_T and h_T;
    for all a ∈ Γ do
        if h_T = g_T^{σ_{T,a}} then return true;
    od;
    return false;
end;
```

Please note that we do not use the theory to the fullest, in the sense that the transversal $\mathcal{T}$ of Corollary 7.5 is not used. The reason is that constructing this transversal in a straightforward way takes as much time as the entire loop, so it is better to just switch by selectors $\sigma_{T,a}$ for all values $a \in \Gamma$ and not just $a \in \mathcal{T}$. Note that if the group is infinite we might find that we must use $\mathcal{T}$ if we can find it, since it might be finite.

Example 7.7
Let h and h' be given in Figure 7.1(a) and (b) respectively, and let g be the skew gain graph of Figure 6.1(b). We would like to determine whether $h \in [g]$ and whether $h' \in [g]$. For the first we observe that if we take the spanning tree T indicated in Figure 6.1(a) by the bold edges and choose root$(T) = 2$, then h_T is the skew gain graph of Figure 6.2. Hence the algorithm will find that σ_{T,r_2} maps h_T into g and the answer will be `true`.

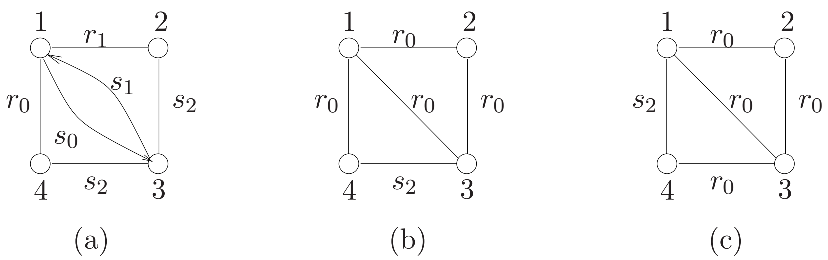


Figure 7.1: Three skew gain graphs

For h', we find that h'_T is the skew gain graph given in Figure 7.1(c). From Example 6.2 we know then that we can not find an alternating selector mapping h'_T

to g, because it is not equal to any of the skew gain graphs in Figure 6.1 or those of Figure 6.2.

The build-up of a switching class (for a fixed but arbitrary spanning tree T with $r = \text{root}(T)$) and the way the algorithm works is illustrated in Figure 7.2.

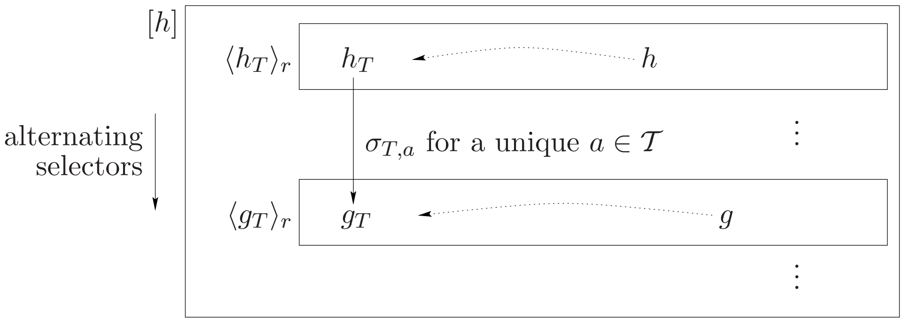


Figure 7.2: The build-up of a switching class and the question $g \in [h]$?

Theorem 7.8
Let Γ be a finite group of order k and G a graph on n vertices. Then the membership problem for (Γ, δ)-gain graphs on G is in $\mathcal{O}(\max(n^2, k \cdot \max(\xi, n)))$, where ξ is the cyclomatic number of G.
Proof:
We determine the complexity of the algorithm by counting the number of edge comparisons. We switch by at most k selectors and must change and compare at most ξ edges every single time. If $\xi > n$ we get a complexity of $\mathcal{O}(k\xi)$, but if $\xi < n$, then constructing the selectors $\sigma_{T,a}$ dominates and we get a time complexity of $\mathcal{O}(kn)$. So, depending on their respective sizes we express the complexity in either the number of vertices, or the number of edges outside the chosen spanning tree.

The n^2 in the formula is because we always have to compute g_T and h_T and in the worst case this takes $\mathcal{O}(n^2)$ time. □

The theory in this chapter, most notably the structure of a switching class for graphs with skew gains as depicted in Figure 7.2, together with the theory of Appendix A.1 allow for an "efficient" way to enumerate all the switches of a skew gain graph g without repetitions (comparable to that of graphs as described in Appendix A.3).

Algorithm 7.9
```
GenerateSwitchingClass (g)
(* Here, g ∈ L_G(Γ, δ) where Γ is a finite group. *)                begin
    T = a spanning tree of G;
    Compute g_T;
    for all different g_T^{σ_{T,a}}
        list the skew gain graphs in ⟨g_T^{σ_{T,a}}⟩root(T)
    od;
end;
```

The algorithm still contains two omissions: first of all, we should only list a refinement if we have not listed it yet. There are two possible ways to make sure this happens: either we have the transversal $\mathcal{T}$ of Theorem 7.4, so that we only apply $\sigma_{T,a}$ with $a \in \mathcal{T}$, or we simply remember which T-canonical skew gain graphs we have already encountered. Fortunately there are at most $|\Gamma|$ of those.

The other "omission" concerns how to list the elements of a root(T)-refinement. This problem is addressed in detail in Appendix A.4.

In the introduction we mentioned the problem of determining whether a particular configuration can occur in a network. We now address this problem, which is in fact equivalent to determining whether a skew gain graph can be embedded into some skew gain graph in a switching class.

In the embedding problem we are given two (Γ, δ)-gain graphs g and h on, possibly different, graphs G and H. The question is whether there exists a $j \in [g]$ such that h can be *embedded* in j, that is, whether there exists an injective function $\psi : V(H) \to V(G)$ such that

$$h(uv) = j(\psi(u)\psi(v))$$

for all edges $uv \in E(h)$. This version of embedding is the labelled version of the full embedding as defined in Chapter 2.

Once we have fixed an injection ψ from h into g, we can restrict g to g' by removing edges that are not in h', where h' is the image of h under ψ. (Note that if h' contains an edge that is not present in g', then we know that ψ does not embed h in g.) With the algorithm described in this section we can answer the question whether $h' \in [g']$. If the answer is affirmative, then we have our embedding ψ; if the answer is negative, then we should try the next embedding. Note that although the membership check can be efficient, there may be many possible injections. In fact, the embedding problem is a hard one, even if restricted to $\mathbf{Z}_2$ (see Section 3.4).

7.3 Improvements in the abelian case

In this section we first look at (Γ, δ)-gain graphs with gains from the centre of Γ. Initially, the anti-involution is arbitrary, but later on we give a further optimization when the anti-involution is the group inversion. The theory in this section differs from the treatment in the previous section, because here we shall *construct* the selector that maps g_T to h_T if it exists, instead of *trying* a number of different selectors. We could have chosen to just improve the algorithm of the previous section for abelian groups, but the treatment here results in a more widely applicable algorithm. We shall demonstrate this in Example 7.12. For simplicity we assume that the underlying graphs are *connected*.

7.3.1 Improvements when g_T has abelian gains

Let G be a connected graph, let T be a spanning tree of G and let $g, h \in \mathbf{L}_G$.

To improve on our algorithm for checking that $h \in [g]$, we assume that at least one of g_T and h_T is abelian. We can assume, without loss of generality, that this is the case for g_T.

Let σ be a selector. Define $G_\sigma \in \mathbf{L}_G$ such that

$$G_\sigma(uv) = \sigma(u)\delta(\sigma(v)), \text{ for all } uv \in E(G) \ .$$

It is easy to prove that if $j \in \mathbf{L}_G$ is abelian, $jG_\sigma = j^\sigma$. In other words: switching by a selector is in this case equivalent to applying the group operation edgewise.

Now, assume that $h_T \in [g_T]$ and let σ be the alternating selector such that $h_T = g_T^\sigma$.

Let $vw \in E(G)$ and $r = \mathrm{root}(T)$. If v and w are of different parity, say v is even and w is odd, then $G_\sigma(vw) = \sigma(v)\delta(\sigma(w)) = \sigma(r)\delta(\delta(\sigma(r)^{-1})) = 1_\Gamma$. If $v, w \in \mathrm{even}(T)$ then $h_T(vw) = g_T^\sigma(vw) = \sigma(r)\delta(\sigma(r))g_T(vw)$, or equivalently, $G_\sigma(vw) = a$, where a can be written as $b\delta(b)$ for some $b \in \Gamma$. For two odd vertices we obtain $G_\sigma(vw) = \delta(\sigma(r)^{-1})\delta(\delta(\sigma(r)^{-1})) = \delta(\sigma(r)^{-1})\sigma(r)^{-1} = (\sigma(r)\delta(\sigma(r)))^{-1} = a^{-1}$.

Summarizing, for gains of the edges of $G_\sigma = {g_T}^{-1}h_T$ we have the following situation for some $a \in \Gamma$.

parity	odd	even
odd	a^{-1}	1_Γ
even	1_Γ	a

The previous leads to the following definition: the set of *skewed squares* of (Γ, δ) is

$$\Delta^2(\Gamma, \delta) = \{a\delta(a) \mid a \in \Gamma\}.$$

In Section 7.3.2 we shall give some properties of this set. First, we shall continue with the problem at hand.

In light of the fact that the above construction works in two directions the previous part of this section can be summarized by the following result.

Theorem 7.10
For g such that g_T is abelian, $h \in [g]$ if and only if the edges between vertices of different parity in ${g_T}^{-1}h_T$ are labelled with 1_Γ and there exists a $b \in \Gamma$ such that the edges between two even vertices (with respect to $\mathrm{root}(T)$) are labelled with $b\delta(b) \in \Delta^2(\Gamma, \delta)$ and the edges between two odd vertices (again, with respect to $\mathrm{root}(T)$) are labelled with $(b\delta(b))^{-1}$.

Theorem 7.11
For $g, h \in \mathbf{L}_G$ with g_T abelian, $h \in [g]$ reduces in time $\mathcal{O}(|E(G)|)$ to the characteristic function of $\Delta^2(\Gamma, \delta)$.

Example 7.12
If $\Gamma = \mathbf{Z}$, the anti-involution δ is the identity, and the underlying graph G is not bipartite, then $C_T^\delta(g)$ equals $\{1_\Gamma\}$, where g is any element of $\mathbf{L}_G$. Consequently, the transversal $\mathcal{T}$ as defined in Theorem 7.4 equals Γ and thus is infinite; hence, the switching class consists of infinitely many u-refinements. On the other hand, the switching class is not equal to $\mathbf{L}_G(\mathbf{Z}, \delta)$. This means that an algorithm like that in Section 7.2, even if it uses the information about the transversal, is not able to solve the membership problem in finite time. On the other hand, the theory developed in this section can be applied, since we shall see later that $\Delta^2(\mathbf{Z}, id)$ contains exactly the even numbers. In other words, $g_T \in [h_T]$ if and only if for ${g_T}^{-1}h_T$ the value of a in the previous discussion is an even number. ◇

7.3.2 The set of skewed squares

In this subsection we give some properties of the set of skewed squares. We begin with an example.

Example 7.13
If δ is the group inversion, then clearly $\Delta^2(\Gamma, \delta) = \{1_\Gamma\}$. It is easy to determine that $\Delta^2(\mathbf{Z}, id)$ is the set of even numbers, $\Delta^2(\mathbf{R}, id)$ where the operation is addition equals $\mathbf{R}$, and $\Delta^2(\mathbf{R}^+, id)$ where the operation is multiplication equals $\mathbf{R}^+$. $\diamond$

The *kernel* of a homomorphism $f : \Gamma \to \Gamma'$ is the set $\ker(f) = \{a \mid f(a) = 1_{\Gamma'}\}$. The *image* of f is $\mathrm{image}(f) = \{b \in \Gamma' \mid b = f(a), a \in \Gamma\}$.

Given a fixed group Γ and anti-involution δ we define the function $s_{\Gamma,\delta}$ by

$$s_{\Gamma,\delta}(a) = a\delta(a) \text{ for } a \in \Gamma \ .$$

Note that $\mathrm{image}(s_{\Gamma,\delta}) = \Delta^2(\Gamma, \delta)$. When Γ and δ are obvious from the context we write s instead of $s_{\Gamma,\delta}$.

Lemma 7.14
For any group Γ, $\Delta^2(\Gamma, \delta)$ is closed under the group inversion, and $\Delta^2(\Gamma, \delta) \subseteq \mathrm{Fix}(\delta)$.
Proof:
Let $s(a) \in \Delta^2(\Gamma, \delta)$. Then $s(a)^{-1} = \delta(a)^{-1}a^{-1} = s(\delta(a)^{-1}) \in \Delta^2(\Gamma, \delta)$.

For the second part, $\delta(s(a)) = \delta(a\delta(a)) = a\delta(a) = s(a)$. □

Note that our example of $(\mathbf{Z}, id)$ under addition tells us that equality of $\Delta^2(\Gamma, \delta)$ and $\mathrm{Fix}(\delta)$ does not hold in general.

We repeat here one of the Isomorphism Theorems (see, for instance, Theorem 2.12 of [41]).

Theorem 7.15 [Isomorphism Theorem]
Let $f : \Gamma \to \Gamma'$ be a homomorphism with kernel Δ. Then Δ is a normal subgroup of Γ and $\Gamma/\Delta \cong \mathrm{image}(f)$.

Lemma 7.16 [new]
If Γ is abelian, then $s : \Gamma \to \Gamma$ is a homomorphism. Moreover, $\ker(s)$ is a normal subgroup of Γ and $\Delta^2(\Gamma, \delta) \cong \Gamma/\ker(s)$.
Proof:
For the first $s(ab) = ab\delta(ab) = ab\delta(b)\delta(a) = as(b)\delta(a) = a\delta(a)s(b) = s(a)s(b)$, because Γ is abelian.

By the Isomorphism Theorem $\ker(s)$ is a normal subgroup of Γ and $\Delta^2(\Gamma, \delta) \cong \Gamma/\ker(s)$.

Example 7.17
Unfortunately, the previous result does not tell us enough to find $\Delta^2(\Gamma, \delta)$. If $\Gamma = \mathbf{Z}$ and δ the identity on $\mathbf{Z}$, then $\ker(s) = \{1_\Gamma\}$. The set $\Delta^2(\mathbf{Z}, id)$ is the subgroup of $\mathbf{Z}$ of the even numbers and it is isomorphic to $\mathbf{Z}/\{1_\Gamma\} \cong \mathbf{Z}$ (the isomorphism maps $2a \in \Delta^2(\mathbf{Z}, id)$ to $a \in \mathbf{Z}$). $\diamond$

Because anti-involutions of a direct product do not always project onto the factors it is unlikely that we can determine the skewed squares of a group Γ with anti-involution δ from the skewed squares of the groups in a factorization of Γ, see Example 5.12. However we do have that involutions δ_1 and δ_2 of groups Γ_1 and Γ_2 respectively can be used to construct an involution (δ_1, δ_2) for $\Gamma_1 \times \Gamma_2$ by applying them componentwise. For involutions thus constructed the set of skewed squares can be constructed from the sets of skewed squares of the factors as proved by the following lemma.

Lemma 7.18 [new]
Let $\Gamma = \Gamma_1 \times \Gamma_2$ and $\delta_i \in \mathrm{INV}(\Gamma_i)$ for $i = 1, 2$. Then $\Delta^2(\Gamma, (\delta_1, \delta_2)) = \Delta^2(\Gamma_1, \delta_1) \times \Delta^2(\Gamma_2, \delta_2)$.
Proof:
It holds that

$$(a, b)(\delta_1, \delta_2)(a, b) = (a, b)(\delta_1(a), \delta_2(b)) = (a\delta_1(a), b\delta_2(b)).$$

The result follows from the fact that $a\delta_1(a) \in \Delta^2(\Gamma_1, \delta_1)$ and $b\delta_2(b) \in \Delta^2(\Gamma_2, \delta_2)$. □

Example 7.19
In this example the involution δ is the identity function and the groups are the cyclic groups. Note that in any decomposition δ does project onto the subgroups.

In this case the definition of $\Delta^2(\Gamma, \delta)$ reduces to

$$\{a \in \Gamma \mid a = b + b \text{ for some } b \in \Gamma\}$$

where, as usual, we use addition to denote the (abelian) operation of Γ.

It is not difficult to see that $\Delta^2(\mathbf{Z}_{2^k}, id) = \{0, 2, 4, 6, \ldots, 2^{k-1}\}$, for $k \geq 1$ and $\Delta^2(\mathbf{Z}_{p^k}, id) = \mathbf{Z}_{p^k}$ where $p > 2$ is prime and $k \geq 1$. We already knew that $\Delta^2(\mathbf{Z}, id)$ is the subgroup of even numbers.

By the fundamental theorem of finitely generated abelian groups, Theorem 2.10 and Lemma 7.18 we can now construct the subgroups $\Delta^2(\Gamma, \delta)$ based on the decomposition of Γ into primary groups when $\delta = id$. ◇

7.3.3 Unrefinable switching classes

In the following we will investigate in which cases a switching class consists of only one u-refinement, hence is itself such a u-refinement. The advantage is that in these cases we need not try any selectors, because $h \in [g]$ if and only if $g_T = h_T$.

Theorem 7.20
Let $g \in \mathbf{L}_G(\Gamma, \delta)$. If g_T is abelian, and G is bipartite or g is inversive, then $\langle g \rangle_{\mathrm{root}(T)} = [g]$.
Proof:
If g is inversive, then the result holds by the fact that $\Delta^2(\Gamma, ^{-1}) = \{1_\Gamma\}$.

For the case that G is bipartite, Theorem 6.17 states that $|[g]| = k^n / |C(A(g_T))|$, where k is the order of Γ and $n = |V(G)|$. The result now follows from Corollary 7.2 since obviously $|C(A(g_T))| = k$. □

Lemma 7.21
Let $g \in \mathbf{L}_G(\Gamma, \delta)$. If g_T is abelian and $\langle g \rangle_{\mathrm{root}(T)} = [g]$ then G is bipartite or g is inversive.
Proof:
Assume that G is not bipartite; we prove that g is inversive.

Let vw be an edge in g_T between two vertices of the same parity. Because g_T is the only (Γ, δ)-gain graph in $[g]$ in which T is 1_Γ-labelled, it holds for every alternating selector σ that $g_T(vw) = g_T^\sigma(vw) = \sigma(v) g_T(vw) \delta(\sigma(w)) = \sigma(v)\delta(\sigma(v)) g_T(vw)$ and this holds if and only if $\sigma(v)^{-1} = \delta(\sigma(v))$. □

Lemma 7.22
Let $g \in \mathbf{L}_G(\Gamma, \delta)$. If $\langle g \rangle_{\mathrm{root}(T)} = [g]$, then h is abelian, where h equals g_T, but with

all edges between vertices of the same parity deleted.

Proof:

Let $vw \in E(h)$, with h defined as above. Because for all alternating selectors σ, $h(vw) = h^\sigma(vw) = \sigma(v)h(vw)\delta(\delta(\sigma(v)^{-1})) = \sigma(v)h(vw)\sigma(v)^{-1}$, it follows that $h(vw)$ commutes with each element of Γ. Hence, h is abelian. □

Lemma 7.23

Let $g \in \mathbf{L}_G(\Gamma, \delta)$. If $\langle g\rangle_{\mathrm{root}(T)} = [g]$, and g is inversive or G is bipartite then g_T is abelian.

Proof:

If G is bipartite then we apply Lemma 7.22 to find that g_T is abelian.

Let g be inversive and G not bipartite. First of all, if g is not abelian, this is because of an edge between vertices of the same parity, by Lemma 7.22. So let vw be an edge between vertices of the same parity. Then, because $\langle g\rangle_{\mathrm{root}(T)} = [g]$, for an alternating selector σ, $g_T(vw) = g_T^\sigma(vw) = \sigma(v)g_T(vw)\delta(\sigma(v)) = \sigma(v)g_T(vw)\sigma(v)^{-1}$. For this to hold, $g_T(vw)$ must commute with each $\sigma(v)$. Hence g_T is abelian. □

Corollary 7.24

Let $g \in \mathbf{L}_G(\Gamma, \delta)$. If $\langle g\rangle_{\mathrm{root}(T)} = [g]$, and g is inversive or G is bipartite, then for all $h \in [g]$ and spanning trees T' of G, $h_{T'}$ is abelian.

Proof:

Let $h \in [g]$. First of all, the bipartiteness of G and the inversiveness of g is independent of the labels of g. Because $\langle g\rangle_{\mathrm{root}(T)} = [g] = [h]$, it holds for all spanning trees T' of G that $\langle h\rangle_{\mathrm{root}(T')} = [h]$. The result now follows from Lemma 7.23. □

In the above we have considered three predicates: g is inversive or G is bipartite (P1), $\langle g\rangle_{\mathrm{root}(T)} = [g]$ (P2) and g_T is abelian (P3). Through manipulation of the previous lemmas we get the following result.

Corollary 7.25

Each pair of predicates P_i, P_j is equivalent under the condition that the remaining predicate, P_ℓ, holds, where $\{i, j, \ell\} = \{1, 2, 3\}$.

Example 7.26

If $\Gamma = \mathbf{Z}_3$, $\delta = id$ and G is complete on three vertices, then for any $g \in \mathbf{L}_G(\Gamma, \delta)$, g_T is abelian, but it is possible that $\langle g\rangle_{\mathrm{root}(T)} \neq [g]$. Hence, if G is not bipartite we really need that g is inversive.

For the same G, but with group S_3 and δ equal to the group inversion, it is possible that $\langle g\rangle_{\mathrm{root}(T)} \neq [g]$, see Example 6.15(1). Hence we really need that the (Γ, δ)-gain graphs are abelian. ◇

Example 7.27

In Example 5.6 we saw an example of a bipartite underlying graph and an abelian group, $\mathbf{Z}_4$. We expect that the switching class of Figure 5.3 contains $4^4/4 = 4^3$ graphs, and this is indeed the case (see also Example 6.18). The reader may verify that indeed the switching class contains only one (Γ, δ)-gain graph in which the edges of the tree T, consisting of the edges $\{1, 2\}$, $\{2, 3\}$ and $\{3, 4\}$, are labelled with 0. It is the (Γ, δ)-gain graph in the upperleft corner. ◇

7.4 Undecidability for arbitrary groups

In the previous sections we omitted the task of specifying the group and the anti-involution so that it can be passed to an algorithm. We assumed it was given and that we could compute with it. One of the problems is to represent a, possibly infinite, group in some way.

Usually a group is specified by means of a *presentation*

$$\Gamma = \langle x_1, x_2, \ldots \mid w_1, w_2, \ldots \rangle \ ,$$

where the x_i are the *generators* and the w_j (words over $\gamma = \{x_1, {x_1}^{-1}, x_2, {x_2}^{-1}, \ldots\}$), are the *relations*. The idea is that by the relations we define a number of sequences of generators that equal the identity of the group (but in general there may be more sequences that equal the group identity; they follow from the words w_i given in the presentation). We shall use α to denote the, in general non-injective, mapping of the strings over γ to an element of the group.

Every presentation determines a group, but a group can have a number of presentations. Because a presentation of a group is used as a parameter to an algorithm, we assume the number of generators and relations to be finite. We shall restrict ourselves to groups with a finite presentation, the *finitely presented* groups. For more information on this subject consult [41] and [38].

Example 7.28
(1) Let us consider the symmetric group S_3. This group has the following presentation

$$\langle r, s \mid r^3,\ s^2,\ r^{-1}srs\rangle \ ,$$

where r is the rotation over 120 degrees and s corresponds to one of the reflections of the triangle, analogous to Figure 2.2.
(2) Although we shall not prove this, the multiplicative group of reals $\mathbf{R}$ does not have a finite presentation. ◇

The *word problem* for presentations of groups is the following: given a word w over γ, does it define the identity of the group? Novikov and Boone have independently proven that there are finitely presented groups that have no presentation for which the word problem is decidable. In fact, if the word problem is undecidable for a given presentation of a group, it is undecidable for all presentations of that group.

A slightly more general problem is the conjugacy problem. Recall that $x, y \in \Gamma$ are conjugates if there is an $a \in \Gamma$ such that $x = aya^{-1}$.

The *conjugacy problem* is now to decide, given two words w_1 and w_2 over γ, whether $\alpha(w_1)$ and $\alpha(w_2)$ are conjugates. In other words, whether there exists a word x over γ so that $\alpha(w_1) = \alpha(x)\alpha(w_2)\alpha(x)^{-1}$.

Because the conjugacy problem generalizes the word problem, which is the special case where w_2 is the empty word, the following result, by Novikov [40], holds.

Theorem 7.29
There exist groups with a finite presentation having an undecidable conjugacy problem.

With this background we shall continue now by proving that the conjugacy problem can be reduced to the membership problem, showing that in general the membership problem for groups specified by means of a presentation is undecidable.

Theorem 7.30 [generalized from Hage [21]]
There exist pairs of groups and anti-involutions for which the membership problem is undecidable.
Proof:
Let w_1 and w_2 be the words for which we would like to decide whether the corresponding elements $\alpha(w_1)$ and $\alpha(w_2)$ are conjugates. Define g_w for a word w over γ to be the (Γ, δ)-gain graph of Figure 7.3, where δ is the group inversion.

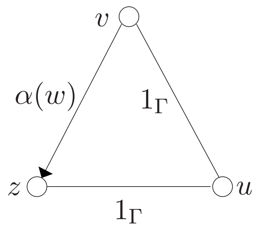


Figure 7.3: The (Γ, δ)-gain graph g_w

Let the tree T consist of the edges uv and uz. If we can decide whether $g_{w_1} \in [g_{w_2}]$, we can solve the conjugacy problem for Γ, because it is solved if we have

$$\alpha(x)\alpha(w_1)\alpha(x)^{-1} = \alpha(w_2)$$

for some x over γ.

Note that by selecting the same value $\alpha(x)$ in all vertices we guarantee that the selector is alternating (remember that the anti-involution is the group inversion). □

Miller III [33] notes that Fridman [17] exhibited a group with a solvable word problem, but of which the conjugacy problem is unsolvable. In that sense the proof of Theorem 7.30 is more general than the corresponding theorem in Hage [21].

Chapter 8

Future Directions

In this short chapter we indicate some problems we have run into (and not completely solved yet) and models we feel are interesting to investigate.

8.1 Problems for switching classes of graphs

The chromatic aspect of switching is studied in Herz [30] in the context of perfect graphs. For colourings, interesting problems are the following

Problem 8.1
What conditions does a graph possess in order for $[G]$ to have a graph H with $\chi(H) = 2\chi(G)$?

Problem 8.2
Characterize the G with $\chi(G) \leq k$ with a set of forbidden graphs. Especially for $k = 2$.

As an aside we mention

Problem 8.3
Characterize the G with $H \in [G]$ a planar graph by means of a set of forbidden graphs.

Because the cycles often figure in our lists of forbidden graphs a nice complexity problem is the following:

Problem 8.4
Given a graph G and a number $k \leq |V(G)|$, can a switch of C_k be embedded into G (in an induced way).

It is quite likely this problem is NP-complete, but that does not follow from our generalization of Yannakakis' result to switching classes.

Something that could help enormously during research is a solution to the following problems:

Problem 8.5
Characterize the maximum (or minimum) size graphs in switching classes.

Problem 8.6
Characterize those switching classes that have a unique maximum (or minimum) size graph in it.

For complexity theory the central problem is to give a general condition that relates problems of graphs and switching classes in terms of complexity.

We would also like to get more insight into those problems which are polynomial for graphs, but become NP-complete for switching classes. As an example, in [35] it is stated that the problem whether a switching class contains a k-regular graph for some k is NP-complete. It remains to be determined whether bipartiteness of switching classes is NP-complete or polynomial. We are also looking for characterizations of switching classes that contain a bipartite graph, and similarly for triangle-free graphs.

From the results in Chapter 4 one might conclude that "acyclic" problems (like hamiltonian path) become easy for switching classes, but it would also be interesting to find some more general statement. For instance hamiltonian circuit is not "acyclic", but in Section 4.1 we have proved that determining whether a switching class has a hamiltonian cycle can be done in polynomial time.

8.2 Problems for switching classes of skew gain graphs

For group theory, it would be interesting to investigate the properties of the skewed versions of group theoretic concepts such as centralizers, but also subgroups. For instance, a subgroup of a group need not be closed under anti-involutions of the group, but the image of a subgroup under an anti-involution is itself a subgroup (of the same order). In this way we might end up with a notion of skewed normal subgroup and a corresponding notion of skewed quotient.

There is still a lot not yet done in the area of skew gain graphs itself. The questions fall into two categories. The first of these is that some of the material just presented still leaves some questions unanswered, the second that we might look for more powerful models based on the one treated here. We start with the former.

For instance, what is the nature of the set $\Delta^2(\Gamma, \delta)$? Can we prove anything about the existence of these sets?

Although we know now that the general membership problem is undecidable, it is clear that by certain restrictions we might find classes of groups and anti-involutions that have a decidable membership problem. To give but an example, if we restrict ourselves to the anti-involution being the identity function (and thus restricting ourselves to abelian groups), we found that the problem was much easier. We might therefore look at other types of anti-involutions for which good results can be obtained.

Another problem to investigate is the influence the choice of tree has on the results obtained in Chapter 7. More specifically, the question is whether we can easily find T such that g_T is abelian (if such a T exists). It seems however, that the choice of T is irrevelant.

A different and, in our opinion, interesting avenue is opened with an extension of the model treated in this thesis.

The framework is essentially quite easy: let Γ be a group and let δ be one of its anti-involutions (maybe sensible to restrict first to inversion). Then we take again an underlying graph and a mapping of the edges into Γ, as always. The difference is now that we can restrict in a vertex u the possible selected values to $\Gamma(u)$, which must be a subgroup of Γ (closed under δ).

The intuition is that in a network of processors, not all processors necessarily have the same capability for actions. With regards to [16], it does mean that at least one of the axioms proposed there should be dropped and there turns out to be only one: if we fix an edge and a value $a \in \Gamma$, it is now not anymore possible to always find a selector so that the edge is after application of the selector labelled by a. We feel this not to be a problem and in fact to be a logical consequence. What we gain is a more diversified model that may be better suited to model the more diverse nature of networks in which printer servers, fileservers, webservers and the like may have different capabilities. Questions that we can put to this model are the same as we have put to the model of skew gain graphs: how large are the classes (they are still equivalence classes; restricting to subgroups is a generalization of the u-refinements in this thesis) and to what extent can we decide the membership problem efficiently?

In the graph case, what becomes of the results on hamiltonian and pancyclic graphs when these restrictions are possible. Refinements are in general smaller than switching classes, so it might be interesting to know to what extent we can uphold the complexity results for polynomial problems for switching classes that are NP-complete for graphs. These may then be of use in approximation algorithms, or they may be used to gain an understanding of exactly *when* a problem becomes hard.

Tyydyttynyt nyt, przyszłym.

Appendix A

Algorithms and Programming Techniques

In this appendix we shall consider a number of problems that came up during the investigation of the material written down in both parts of the thesis. These results are not always directly related to switching classes, but were in their way very useful during our research.

To pave the way for later sections we first examine the problem of enumerating the submultisets of a certain multiset. The theory developed here enables us to develop an algorithm that can efficiently enumerate u-refinements. In Section A.2 we illustrate the general results of Section A.1 by the simplest possible example and show links to the Game of Hanoi and Gray Codes; these links are well known. For the background on the Game of Hanoi I acknowledge the help of A.M. Hinz.

We continue then with a method for generating all the graphs in a switching class in an efficient way. Instead of applying selectors of varying sizes, it is shown how to obtain all switches of a graph (exactly once) by applying only singleton selectors. In a following section we repeat this for finite groups with arbitrary anti-involutions. The split up is both a consequence of the split of the thesis into two parts, and the fact that the algorithm for the case of undirected graphs is easier to understand. It can in this way serve as an introduction to the more general algorithm, in which case some knowledge of group theory is useful.

A.1 Enumerating submultisets

In this section we shall devise a way to sequence all submultisets of a certain multiset.

A *multiset* S over V is a function $S : V \to \mathbf{N}_0$. The value $S(v)$ for $v \in V$ is the *multiplicity* of v in S. A multiset S' over V is a *submultiset* of S if for all $v \in V$, $S'(v) \leq S(v)$. The *cardinality* or *size* of a multiset includes multiplicity: $|S| = \sum_{v \in V} S(v)$.

For $m > 1$, let $S(m, n)$ be the multiset over $V = \{v_0, \ldots, v_{n-1}\}$ so that $S(v) = m - 1$ for all $v \in V$. If we linearly order the elements of V from v_{n-1} to v_0, then we can code any submultiset by the multiplicities, $S(v_{n-1}) \ldots S(v_0)$, in other words, a number with base m. We can also consider these numbers to be strings of length n over $M = \{0, \ldots, m-1\}$. Because of the obvious bijections between submultisets, numbers with base m and these strings, we shall use them interchangeably. We note that there are exactly m^n submultisets of $S(m, n)$.

Example A.1
For $m = 3$ and $n = 4$, $S = S(m,n) = \{v_0, v_0, v_1, v_1, v_2, v_2, v_3, v_3\}$. Some examples of submultisets are $\{v_0, v_0, v_1, v_2, v_3, v_3\}$, $\{v_0, v_0, v_1, v_2\}$ and $\{v_1, v_1, v_2, v_3, v_3\}$; the corresponding numbers with base 3 are 2112, 0112 and 2120 and these can also be understood as strings over $\{0, 1, 2\}$. It is important to remember that our strings are zero-indexed and position zero is at the extreme right. Hence 0 has position zero in 2120, and 1 has position two.

Sometimes we shall want to refer to the decimal equivalents of the numbers with base k. For example, converting the numbers 2112, 0112 and 2120 into the decimal system we obtain $2 \cdot 27 + 1 \cdot 9 + 1 \cdot 3 + 2 \cdot 1 = 68$, $1 \cdot 9 + 1 \cdot 3 + 2 \cdot 1 = 14$ and $2 \cdot 27 + 1 \cdot 9 + 2 \cdot 3 = 69$. ◇

Given the fact that similar bitstrings yield widely different decimal numbers led Frank Gray to define and patent the Gray Code [19]: he devised a coding of decimal numbers into bitstrings (the case $m = 2$) such that two decimal numbers k and $k+1$ are coded by bitstrings that differ in exactly one position. This code was used to reduce the importance of transmission errors. We shall now generalize this method to strings over M.

First we introduce some notation for rooted edge-labelled trees. The trees we consider have arity m and are complete. In other words, every internal vertex has exactly m children and the leaves are all at the same level. The children are ordered from left to right. We shall refer to them as child i for $i \in M$. The labels we shall use for the edges are the elements of M and we demand that for each internal vertex, the edge to every child is labelled with a unique element of M. Hence the labels on the edges to the children of a certain internal vertex are a permutation of M.

Recall that the *level* of a vertex v in a tree is the number of vertices on the path from the root to v. Hence the level of the root is 1. The *height* of a tree is the level of the lowest leaf in the tree minus one. Hence the height of the trivial tree is 0.

Let T be a tree of height n and let $\pi = (a_1, \ldots, a_{n'})$ be a sequence over M with $n' \leq n$. Then π determines a vertex $C(\pi, T)$, as follows: $C(\lambda, T) = \mathrm{root}(T)$, and $C(a\!:\!\pi, T) = C(\pi, T')$ where T' is the subtree rooted at child a. Similarly we define $L(\pi, T)$ as follows: $L(\lambda, T) = \mathrm{root}(T)$ and $L(a\!:\!\pi, T) = L(\pi, T')$ where T' is the subtree reachable from $\mathrm{root}(T)$ by an edge labelled with a. As a mnemonic, C stands for Child directed and L for Label directed.

It should be clear that in both cases there is a bijection between sequences $\pi = a_1, \ldots, a_{n'}$ over M for $n' \leq n$ and vertices in the tree. Hence we may define for a vertex v in T, $L(v,T) = \pi$ if $L(\pi, T) = v$ and $C(v, T) = \pi$ if $C(\pi, T) = v$. Remember that we can interpret the value of $C(v,T)$ and $L(v,T)$ as a string, as a number with base k, and as the corresponding natural number.

The *least common ancestor* of two different vertices $v_1, v_2 \in V(T)$ is the unique vertex $v = \mathrm{lca}(v_1, v_2)$, so that the paths from the root to v_1 and v_2 split up in v.

If v_1 and v_2 are on the same level of the tree, we say that v_2 *follows* v_1 in T if $C(v_2, T) = C(v_1, T) + 1$. This means that v_2 is the first vertex to be encountered starting at v_1 and going to the right on the same level. Note that in this case, the labels to the subtrees of $\mathrm{lca}(v_1, v_2)$ containing v_1 and v_2 respectively differ at most one in their labels.

For natural numbers m and k, define $\rho(m,k) = \max_i(m^i | k)$ where $q|k$ means that q divides k. In words, $\rho(m,k)$ yields the number of powers of m dividing k. A basic property of this function is the following:

Lemma A.2
For $p \geq 0$ and $m^p < k < m^{p+1}$, $\rho(m,k) = \rho(m, k-m^p)$.
Proof:
Let $j = \rho(m,k)$, that is, $k = m^j q$, where $m \nmid q$. Then $j \leq p$ and $k - m^p = m^j(q - m^{p-j})$. In particular, $\rho(m, k-m^p) \geq \rho(m,k)$. For inequality we should have $m|(q-m^{p-j})$, that is, $p = j$ and $m|q-1$, because $m \nmid q$. But now, remember in this case $p = j$, $m < q$ contradicts the assumption $k < m^{p+1}$. □

Lemma A.3
Let v_1 and v_2 be leaves so that v_2 follows v_1. Then the leftmost position in which $c_1 = C(v_1, T)$ and $c_2 = C(v_2, T)$ differ is $\rho(m, C(v_2, T))$.
Proof:
Let v_1 and v_2 be leaves so that v_2 follows v_1. Let $c_1 = C(v_1, T)$ and $c_2 = C(v_2, T)$ and let $v = \mathrm{lca}(v_1, v_2)$ and $w = C(v, T)$. Now

$$c_1 = wi\,\overbrace{(m-1)\ldots(m-1)}^{p} \text{ and } c_2 = w(i+1)\overbrace{0\ldots0}^{p}$$

for some p and i. This follows because for v_2 to follow v_1, v_2 is the leftmost child in its subtree (of r) and v_1 is the rightmost child in its subtree (of r).

Note that the first position in which c_1 and c_2 differ is the pth. Because c_2 ends in p zeroes, $m^p | c_2$ and, because $i+1>0$, also $m^{p+1} \nmid c_2$. Hence $p = \rho(m, C(v_2, T))$.
□

Let T be a tree of the kind described above. The *mirror* of T, denoted by $\overleftarrow{T}$, is the tree where, for each internal vertex, the label of child i is exchanged with the label of child $m-1-i$, for $i = 0, \ldots, \lfloor m/2 \rfloor$, where $\lfloor a \rfloor$ is the largest integer less than or equal to a. Note that only the labels are changed and for the rest the tree stays intact. Also note that $\overleftarrow{\overleftarrow{T}} = T$.

We shall now define recursively the type of trees we are interested in. The tree T_0^m is equal to the trivial tree; the tree T_n^m is the tree consisting of a vertex, say v, with m subtrees $T_{n-1}^m, \overleftarrow{T_{n-1}^m}, T_{n-1}^m, \overleftarrow{T_{n-1}^m}, \cdots$ ordered from left to right. The edge to child i of v, for $i \in M$, is labelled with i. In Figure A.1 this construction is pictorially represented, where $T = T_{n-1}^m$ if m is odd and $T = \overleftarrow{T_{n-1}^m}$ if m is even.

Given a complete m-ary tree of height n there is also a non-recursive way to obtain the tree T_n^m making it easy to recognize whether the tree is correctly constructed: between each two levels the edges are labelled from left to right

$$0, 1, \ldots, m-1, m-1, m-2, \ldots, 0, 0, 1, \ldots$$

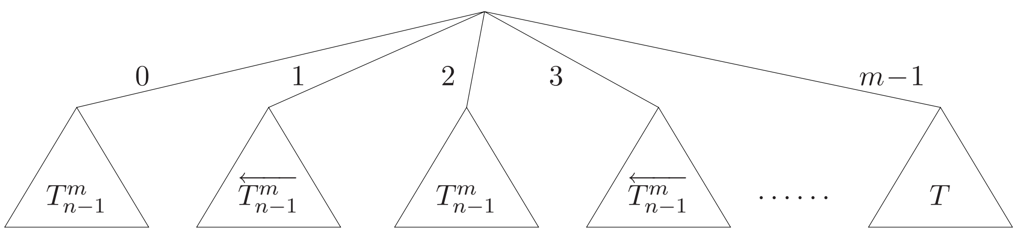


Figure A.1: The tree T_n^m schematically

Lemma A.4
Let $T = T_n^m$ for natural numbers m, n and let $v_1, v_2 \in V(T)$ be such that v_2 follows v_1. Then $\ell_1 = L(v_1, T)$ and $\ell_2 = L(v_2, T)$ differ only in position $p = \rho(m, C(v_2, T))$. Moreover, $\ell_2(p) = \ell_1(p) + 1$ if the number of odd numbers occuring in ℓ_1 to the left of p is even, and $\ell_2(p) = \ell_1(p) - 1$ otherwise.
Proof:
Let $v = \text{lca}(v_1, v_2)$. From the root to v there are obviously no differences between $L(v_1, T)$ and $L(v_2, T)$. Because the path splits up at v and the edges to its children are labelled with different elements of M, the sequences differ in this position. From then on the paths are the same: because v_2 follows v_1 the subtrees of v to which they belong are mirrors of each other and in these subtrees, v_1 and v_2 are rightmost and leftmost vertex of their respective subtrees (see Figure A.2).

The first position at which $C(v_1, T)$ and $C(v_2, T)$, and $L(v_1, T)$ and $L(v_2, T)$ differ obviously coincide and so $p = \rho(m, C(v_2, T))$ follows from Lemma A.3.

The last claim follows from the fact that every odd number on the path to the least common ancestor implies a mirroring of the subtree. If this number is even, then an even number of mirror operations yields a tree where the labels to the children are in the original ascending order; otherwise they are in descending order.
□

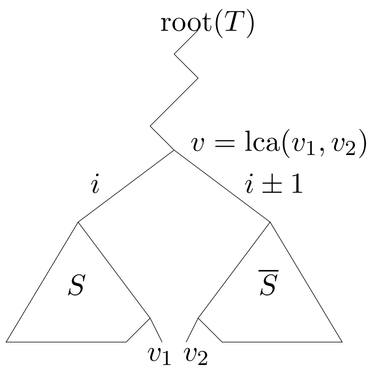


Figure A.2: Illustrating Lemma A.4

The result of this section can be summed up as follows

Corollary A.5
Let m, n be integers. A list of all submultisets of $S(m, n)$, say $S_0 = \emptyset, S_1, \ldots, S_{m^n-1}$, can be constructed so that S_i and S_{i+1} differ only in the multiplicity of v_p where $p = \rho(m, i+1)$. Also, $S_{i+1}(v_p) = S_i(v_p) + 1$ if $\sum_{k=p+1}^{n-1} S_i(k)$ is even and $S_{i+1}(v_p) = S_i(v_p) - 1$ otherwise.

For completeness we give an algorithm that given $s = L(v, T)$ constructs $t = C(v, T)$. We remember how many "mirrorings" are encountered during the virtual traversal of the tree T_n^m. We keep this in the variable `complemented`. We negate this boolean if and only if we would traverse an odd-labelled edge in T_n^m. In the algorithm $m > 1$ is an integer and s is a string of length n over M.

```
complemented = false;
for i = n-1 downto 0 do
```

```
    if complemented
      t[i] = m-1-s[i];
    else
      t[i] = s[i];
    if;
    if odd(s[i])
      complemented = not(complemented);
    if;
  od
```

A.2 An example: the case $m = 2$

In this section we illustrate the previous section by the simplest possible example. The more general application will be treated in Section A.4 and involves the second part of the thesis up to Chapter 7. We first specialize Corollary A.5 for $m = 2$.

Corollary A.6
For an integer n, a list of all subsets of $\{v_0, \ldots, v_{n-1}\}$, say $S_0 = \emptyset, S_1, \ldots, S_{2^n-1}$, can be constructed so that $S_i \ominus S_{i+1} = \{v_p\}$ for $0 \leq i \leq 2^n - 2$ where $p = \rho(2, i+1) = \max_j(2^j|(i+1))$.

We now give an example for $n = 4$. The corresponding tree T_4^2 is given in Figure A.3. Note that in this special case there is no need to know whether we have to increment or decrement as in Corollary A.5: we can simply negate.

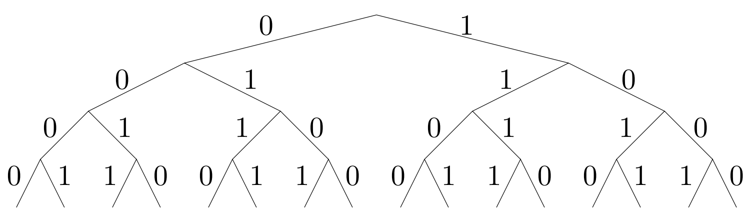


Figure A.3: The tree T_4^2

The Gray Code – or equivalently, the list of subsets of $\{v_0, \ldots, v_3\}$ – can now easily be reconstructed from the tree. Start at the root. The leftmost edge is labelled 0, so we shall first construct the bitstrings starting with 0. Then by taking the leftmost edge every time, we obtain the first bitstring 0000. By proceeding to the next leaf repeatedly we obtain 0001, 0011, 0010, 0110, 0111, ... Recall that the rightmost bit says whether v_0 is in the set and the leftmost bit tells us whether v_3 is in the set. The corresponding subsets are $\emptyset, \{v_0\}, \{v_0, v_1\}, \{v_1\}, \{v_1, v_2\}, \{v_0, v_1, v_2\}, \ldots$.

To see that Corollary A.6 works, we shall show that it correctly determines the change from 0111 to 0101. The value of i is 5 in this case (numbering starts at zero) and so the value for p is 1, which is the number of two-divisors of $6 = i + 1$. And in fact the bit for v_1 is the second from the right, which is indeed the bit that was negated.

The connection with Gray Codes and the Game of Hanoi is well known [31], [46] and [3]. We shall in short repeat the game and its solution: the player is given three pegs (numbered 0, 1 and 2), on the first of which rests a pile of n discs stacked on top of each other in order of decreasing size. The goal of the game is to move the entire pile of discs to one of the other pegs, say peg 2, but the moves are restricted by the following rules:

- You may only move a single disc, coming from the top of a pile.
- You may only place a disc on a larger one or on an empty peg.

The optimal solution is most easily specified by recursion: first move the top $n-1$ discs to peg 1, then the largest disc to peg 2 and again by recursion, move the pile of $n-1$ discs, now on peg 1, to peg 2. The base case is the moving of a single disc.

In our case, the discs are numbered from v_0 up to v_{n-1}, in order of increasing size. The link of Theorem A.6 with Hanoi is that the kth move in the optimal solution of the Game of Hanoi, involves the moving of the disc $\rho(2,k) = \max_j(2^j|k)$ (in this case it is natural to start counting with $k = 1$). Hence the order of the discs using four discs should be $v_0, v_1, v_0, v_2, v_0, v_1, v_0, v_3, \ldots$

The link is established by the following result.

Theorem A.7
In move k of the optimal solution to the Hanoi game, disc j is moved where $j = \rho(2,k)$.
Proof:
The proof is by induction on the number of discs. For the base case, $n = 1$, there is only one move, so we need consider only $k = 1$. The only disc that can be moved is v_0 and indeed $0 = \rho(2,1)$.

Let $n > 1$ and $1 \leq k \leq 2^n - 1$. The first $2^{n-1} - 1$ of these involve the moving of the first $n-1$ discs to the second peg. By induction the disc to be moved for $k < 2^{n-1}$ is indeed the disc v_j where $j = \rho(2,k)$.

If $k = 2^{n-1}$, then $\rho(2,k) = n-1$ and, indeed, in this move we should move the largest disc v_{n-1}.

For $2^{n-1} < k < 2^n$, we should obtain the same sequence of moves as for $k < 2^{n-1}$, which holds if $\rho(2,k) = \rho(2, k - 2^{n-1})$. This follows from Lemma A.2. □

If we want to use the previous algorithm to play the Game of Hanoi it is not sufficient to know which disc to move: we also need to know where to move it. This can be determined by examining the non-recursive way of playing the Game of Hanoi: in the first step we can only move disc v_0. The second move should involve disc v_1, because moving disc v_0 again yields a less than optimal solution. So we move disc v_1 and there is only one possible destination – the peg having disc v_0 on top and the peg on which v_1 itself rested being disqualified. If disc v_0 was moved to the right, to peg 1, then disc v_1 has to move to the left (modulo pegs), to peg 2. In the third move we must move disc v_0 again: v_1 has just been moved and disc v_2 can not be moved. This holds in general: if we have just moved disc v_0, then another disc must be moved to ensure optimality. If there is only one peg with a disc on top other than disc v_0, then its topmost disc is moved to a, by necessity, empty peg. If there are two pegs with a disc other than v_0 on top, then we can only move the smallest of the discs on top of these unto to the top of the other. Hence Hanoi seems to be governed by the following two rules:

- every odd move disc v_0 is moved to the right, and

- every even move the smallest of the two other discs, i.e., that is not v_0 is moved. Of course, if there is only one "other disc" we move that one. There is only one possible destination peg.

Note that the algorithm implied here is deterministic except for the initial choice to move v_0 to either the left or the right. This will determine – with the fact whether n is even or odd – where the stack finally ends up (on the second or the third peg). One can check that if n is odd, then the stack will be moved to the peg where disc v_0 is placed the first time, and if n is even, then the stack will end up on the other peg. Without proof we state that all even numbered discs go the same direction as v_0 and the odd ones go the opposite direction.

As an example, consider Figure A.4 where is depicted the first part of the Game of Hanoi for four discs. For the remaining moves, it suffices to stop at configuration 8 and continue while looking at the page from the other side of the paper (peg 0 and 2 are then interchanged).

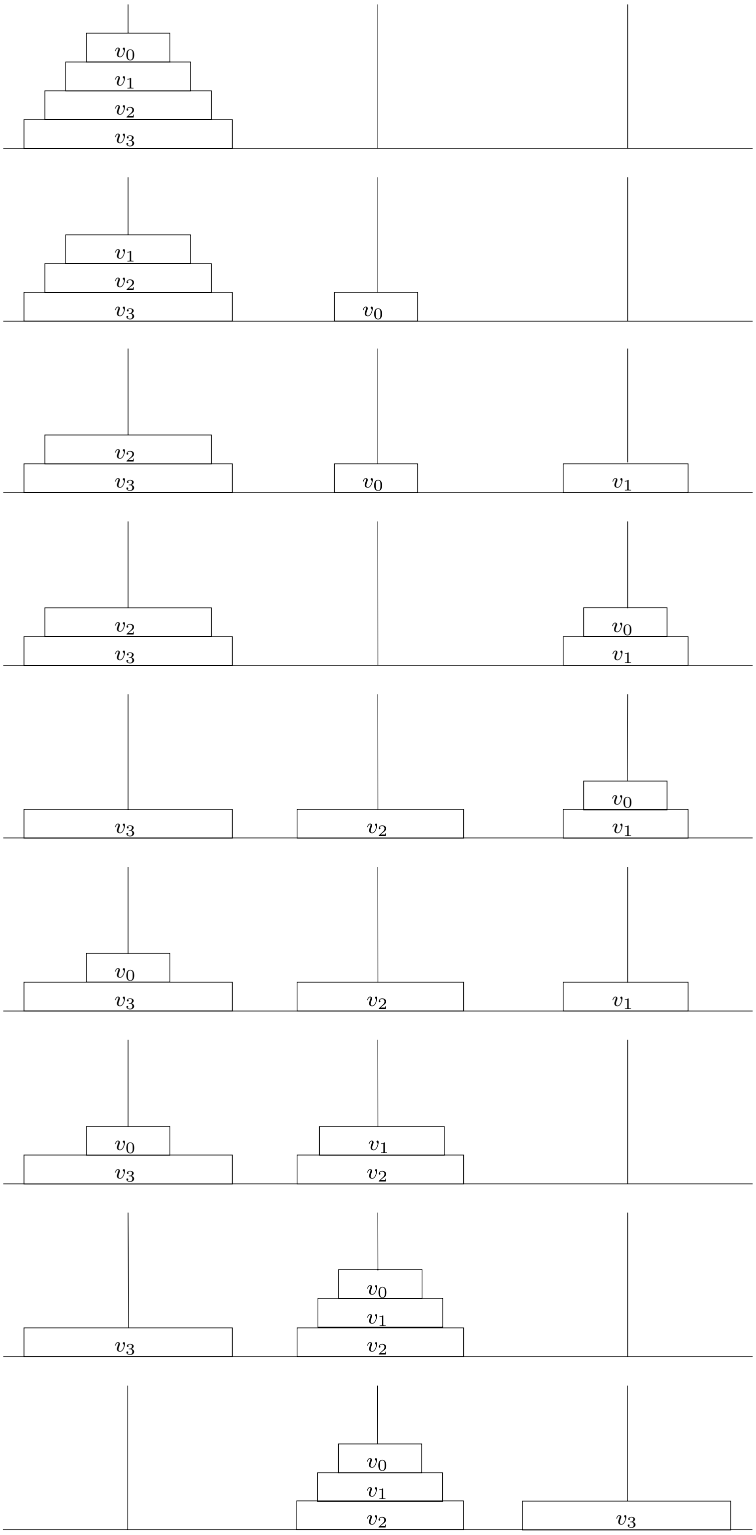
Although it has little to do with the Gray Codes of the above, we shall take some time here to link the number k of the configuration with the configuration itself.

Any configuration of the Game of Hanoi on n discs can be coded as a bitstring of length n over $\{0, 1, 2\}$, where these numbers code the peg numbers. Of course, not every configuration is used in the optimal game of Hanoi, but which are? Assume that we are to move the pile from peg 0 to peg 2. First of all, the leftmost bit, bit $n - 1$, is either 0 or 2, because moving the largest disc to peg 1 is not necessary and hence never done. Before it becomes 2, half the game has been played and in that first half of the game bit $n - 2$ has only been 0 or 1. This holds in general: of the three possible values, only two are used. These two are given by the source and destination peg at that point in the recursive algorithm. If we remember from which peg to which peg we are moving (the other one is then by definition the "via" peg), then we know what values to put there: if bit i is set, then disc v_i has already been moved to the destination peg, and otherwise the move has not been made yet and it is still on the source peg. Based on these two cases we exchange the free peg with the source respectively destination peg, according to the optimal solution.

The algorithm maps the configuration number k into c, the latter is a string over $\{0, 1, 2\}$. The free peg has number $3 - dest - src$.

```
src  = 0;
dest = 2; // change to 1 if you want to end up on peg 1
for i = n-1 downto 0 do
    if k[i] then
        c[i] = dest;
        src = 3-dest-src;
    else
        c[i] = src;
        dest = 3-dest-src;
    fi;
od;
```

A variation of this algorithm can also be used to determine from the configuration the index of this configuration in the Game of Hanoi, in other words, the value k. The rules are more or less the same, but instead of using k to determine c, we use c to determine k. For instance, if `src` $= 1$ and `dest` $= 0$ and $c[i] = 1$, then $k[i] = 0$ and



k	$\max_j 2^j \| k$	$v_3v_2v_1v_0$
1		0000
	0	
2		0001
	1	
3		0011
	0	
4		0010
	2	
5		0110
	0	
6		0111
	1	
7		0101
	0	
8		0100
	3	
9		1100

Figure A.4: Part of the Game of Hanoi for 4 discs `dest` becomes the free peg, because `src` $= c[i]$. If, on the other hand, `dest` $= c[i]$, then we would get $k[i] = 1$.

A.3 The computation of the switches of a graph

The most obvious way to generate all graphs in the switching class of a graph G on $V = \{v_0, \ldots, v_{n-1}\}$ is to switch with respect to all selectors $\sigma \subseteq V$ that do not contain a fixed vertex, say v_{n-1}. The need for omitting v_{n-1} comes from the fact that $G^\sigma = G^{V-\sigma}$. We get an improvement if we apply $V - \sigma$ if $|\sigma| > n/2$. Notwithstanding this improvement, there are still $\mathcal{O}(n^2)$ edges to be changed in both the worst and the average case. To generate the entire switching class we need time $\mathcal{O}(n^2 2^{n-1})$.

The results of the previous sections allow us to apply a singleton selector every time and still obtain all possible switches of G exactly once. This method is graphically depicted in Figure A.5. The original method of switching is also graphically present in this picture: take the dotted edges and the leftmost edge from G to G_1.

The index of the vertex to be switched can be determined using Corollary A.6 for the set $\{v_0, \ldots, v_{n-2}\}$. Notice that we can in fact return to the original graph by switching with respect to v_{n-2} at the end. This implies that we do not need to make a copy of G before starting to switch. This holds in general if m (in T_n^m) is even: the leftmost and rightmost path in the tree T_n^m for any $n > 0$ differ only in the edges from the root.

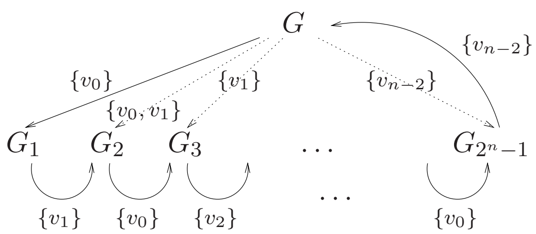


Figure A.5: Cumulative switching

Another view on the result is as follows: define a graph where the vertices are the graphs in a certain switching class, where two graphs have a edge between them if they can be switched to each other by a singleton selector. The results obtained so far imply that this graph has a hamiltonian path – if m is even, a hamiltonian cycle.

The overall effect of the improvement of switching singleton selectors instead of arbitrary ones is to reduce the number of edges to be changed in each switch to $n-1$, yielding an average and worst-case complexity of $\mathcal{O}(n2^{n-1})$ instead of $\mathcal{O}(n^2 2^{n-1})$. It should be clear that this is optimal: every switch modifies at least $n-1$ edges. However, we still need an efficient way to compute $\rho(2, k)$.

The following pseudo code computes the index p of the vertex v_p to be switched, given the rank number of the subset/selector $k > 0$ in the list of selectors.

```
int p (int k)
  for i=0 to n do
    if k[i] is set then
      return i;
```

```
    fi;
  od;
```

The program simply finds the first position in k, coming from the right, where the bit is set.

Clearly, this program works in time $\mathcal{O}(\log(k)) = \mathcal{O}(n)$ in the worst case since $k \leq 2^n - 1$. Executing this function for $1 \leq i \leq 2^{n-1}$ results in a first approximation for the number of loops in the function `p` of $n(2^n - 1)$.

This is not a very good estimate, because half of the parameters to the function will be odd numbers and in this case the for loop is evaluated only once.

In fact, a more detailed examination finds the following number of loops:

$$\sum_{i=0}^{n-1} 2^i(n-i) \ . \tag{A.1}$$

For a given value i, 2^i gives the number of integers that take $n - i$ loops. Splitting up this summation into two we obtain

$$\sum_{i=0}^{n-1} n2^i - \sum_{i=0}^{n-1} i2^i \ .$$

The first part is simply $n(2^n - 1)$. For the second part we define

$$f_n(x) = \sum_{i=0}^{n} ix^i \text{ and } F_n(x) = \sum_{i=0}^{n} x^i = \frac{x^{n+1} - 1}{x - 1} \ .$$

The latter equality is standard from calculus. Now note that the derivative F'_n (as a summation) multiplied by x equals exactly f_n. By taking the derivative of the closed formula we obtain

$$F'_n(x) = \frac{(n+1)x^n(x-1) - (x^{n+1} - 1)}{(x-1)^2} = \frac{nx^{n+1} - (n+1)x^n + 1}{(x-1)^2}$$

Hence

$$f_n(x) = xF'_n(x) = \frac{nx^{n+2} - (n+1)x^{n+1} + x}{(x-1)^2}$$

Substituting 2 for x and $n - 1$ for n we obtain $(n-1)2^{n+1} - n2^n + 2$ for the second part of the summation of (A.1). Subtracting it from $n(2^n - 1)$ yields the final number of loop condition evaluations which is

$$2^{n+1} - n - 2$$

Some experimental results were obtained running on a computer with an Intel Pentium I 120Mhz running under Linux. The measurements were obtained using the program `gprof 2.9.1`. It excludes the time spent in `mcount`, which is time spent on profiling. In the row "simple" we give timings for a switching algorithm that simply applies half the selectors to a fixed graph in the switching class, while the optimized version uses the just described algorithm for sequentially generating all switches by applying singleton selectors consecutively. The timings for the optimized version are given in the row "cumulative".

Total running times (in seconds):

n→	8	9	10	11
simple	0.26	4.79	175.26	22097.78
cumulative	0.14	1.96	64.67	18918.60

Times spent in the actual function that does the switching (in seconds):

n→	8	9	10	11
simple	0.16	3.54	136.21	9609.93
cumulative	0.05	0.75	24.41	6739.49

A.4 Switching skew gain graphs

Let Γ be a finite group with carrier $\{x_1, \ldots, x_k\}$ and anti-involution δ. Let $V = \{v_0, \ldots, v_q\}$ be a set of vertices, $G = (V, E)$ be a connected graph and T a rooted spanning tree of G. Let $g \in \mathbf{L}_G(\Gamma, \delta)$ such that T is 1_Γ-labelled in g. We assume $\mathrm{root}(T) = v_q$.

In this section we show how to apply the theory of Section A.1 to list all switches in $\langle g \rangle_{v_q}$. Originally, the switches were obtained by applying all v_q-selectors to g. In other words, we should apply all selectors that select 1_Γ in v_q and arbitrary values in the other vertices $v_0, \ldots, v_{q-1}$. There are exactly k^q of those. Listing all the selectors in the same cumulative way as we did in the previous section for ordinary graphs is not difficult: take $m = k$ and $n = q$ and apply the theory of Section A.1, which enumerates all submultisets of $S(k, q)$ which is equivalent to listing all the selectors $\sigma : \{v_0, \ldots, v_{q-1}\} \to \Gamma$ if we allow the multiplicity j of v_i to indicate the value x_{j+1} selected in v_i.

There are differences with the previous section however: there it was obvious what to do, because a vertex is either selected or it is not. In the case of arbitrary groups it is not sufficient to know that in a vertex we must select a value $a \neq 1_\Gamma$: we must also know what value to select in that vertex.

Instead of giving a formal algorithm we explain the method by an example.

Example A.8
In Figure 6.1 the graph G and $g \in \mathbf{L}_G(S_3, f_1)$ are shown, where f_1 is the anti-involution of Example 5.1 mapping s_0 into s_1, s_1 into s_0 and all other elements to themselves. The spanning tree T is indicated in G by the bold edges, with vertex 1 as its root. To make our example correspond more closely to the theory we set $v_0 = 2, v_1 = 3$ and $v_2 = 4$; the vertex 1 takes the role of v_q.

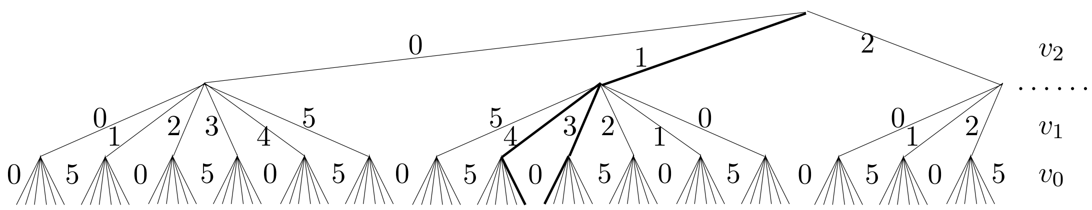


Figure A.6: Part of T_3^6

Part of the tree T_3^6 is given in Figure A.6. Because the carrier of S_3 is the set $\{r_0, r_1, r_2, s_0, s_1, s_2\}$ and not $M = \{0, \ldots, 5\}$, we introduce a bijection $\psi : M \to S_3$ that handles the translation between them by enumerating the pairs: $\{(i, r_i), (3 +$

$i, s_i) \mid i = 0, 1, 2\}$. The effect of ψ is that it orders S_3. Although this linearization is quite arbitrary, we do demand that 0 maps to the group identity.

Applying the theory of Section A.1 we can list all functions $\tau : \{v_0, v_1, v_2\} \to M$, and, by the bijection ψ, all selectors $\sigma = \psi \cdot \tau$, ordered so that the difference between any pair of subsequent selectors in this list is that they select a different value in exactly one vertex.

Two such subsequent selectors are

$$\tau_1 = \{(v_0, r_0), (v_1, s_1), (v_2, r_1)\} \text{ and } \tau_2 = \{(v_0, r_0), (v_1, s_0), (v_2, r_1)\} \; .$$

Coded as bitstrings these are 140 and 130; they are indicated by the bold lines in Figure A.6. Because $1 = \rho(6, 48)$ (note that 130 is the 48th leaf in the tree of Figure A.6), g^{τ_2} can be obtained from g^{τ_1} by applying a selector that maps all vertices except v_1 to 1_Γ.

It may not be evident why 140 precedes 130 in the listing. This can be construed from Corollary A.5 as follows: let us take 140 as the current submultiset and construct the next one using our result. The rule is that if we take the string of values up to but not including the position of change (this is the string $s = 1$), then we can determine whether we are ascending on descending for position 1 (i.e., vertex v_1), by counting the number of odd numbers in s. In this case that number is odd, and we can conclude with Corollary A.5 that the numbering on the level of v_1 is descending.

It remains to be determined what value to select in $v_1 = 3$. For the selector σ, it should hold that $g^{\tau_2} = (g^{\tau_1})^\sigma$. Equivalently $\tau_2 = \sigma\tau_1$ by Lemma 5.5, or, $\sigma(v) = \tau_2(v)\tau_1(v)^{-1}$ for all $v \in \{v_0, v_1, v_2\}$. Because τ_1 and τ_2 differ only on v_1, it holds that $\sigma(v) = 1_\Gamma$ for $v \in \{v_0, v_2\}$ as promised and $\sigma(v_1) = \tau_2(v_1)\tau_1(v_1)^{-1}$. Hence $\sigma(v_1) = s_0 {s_1}^{-1} = s_0 s_1 = r_1$. Given the strings 140 and 130 instead of the selectors themselves, it is simply a question of applying ψ: in v_1 we should select $\psi(3)(\psi(4))^{-1}$.

The (S_3, f_1)-gain graphs g^{τ_1} and g^{τ_2} are given in Figure A.7 and the reader may verify that indeed $g^{\tau_2} = (g^{\tau_1})^\sigma$. ◇

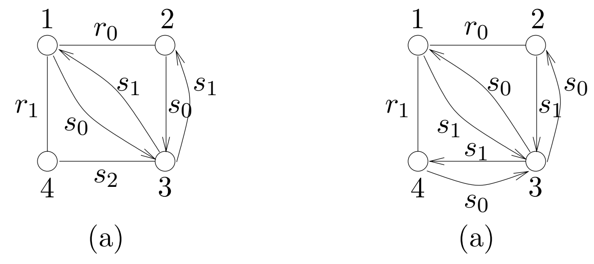


Figure A.7: The skew gain graphs g^{τ_1} and g^{τ_2}

We now summarize our findings. Let $0 \leq \ell < k^q$ and let s be the string over $M = \{0, \ldots, k-1\}$ denoting selector number ℓ. We show how to find selector $\ell + 1$.

First, compute the position p in s where the change occurs: $p = \rho(k, \ell + 1)$. Hence s is of the form wiw' where i is the value at position p. The resulting string over M is of the form wjw' where $i = j - 1$ or $i = j + 1$. To know which of the two is the case, we remember for each level whether we are ascending or descending. Whenever we reach one of the bounds on a certain level, 0 or $k - 1$, then we reverse on that level. The correctness of this method follows from the fact that subsequent subtrees are each others mirror image, and so incrementing and decrementing are alternated; all levels start with incrementing.

Given i and j it is easy to compute the element of the group to be selected in v_p: $\sigma(v_p) = \psi(j)(\psi(i))^{-1}$. Note that, as may be expected, the selector σ^{-1} induces a change from wjw' to wiw'.

Example A.9
We continue Example A.8. In Figure A.8 we have listed all 216 elements of $\langle g \rangle_1$. Looking up the (S_3, f_1)-gain graph with number 130 and 140 – 54 and 60 respectively in decimal, both indicated boldfaced in Figure A.8 – we find the skew gain graphs of Figure A.7. Note that if the selectors are applied in the order implied by the algorithm implied above, these switches should be adjacent (in fact, they would be number 47 and 48 in that sequence). However, the construction used in Figure A.8 is the classical one in which the value selected in 2 changes the fastest and in 4 the slowest.

Of course (cf. Example 6.20) the entire switching class consist of three of these refinements, the other two are generated by the (S_3, f_1)-gain graphs of Figure 6.2.$\diamond$

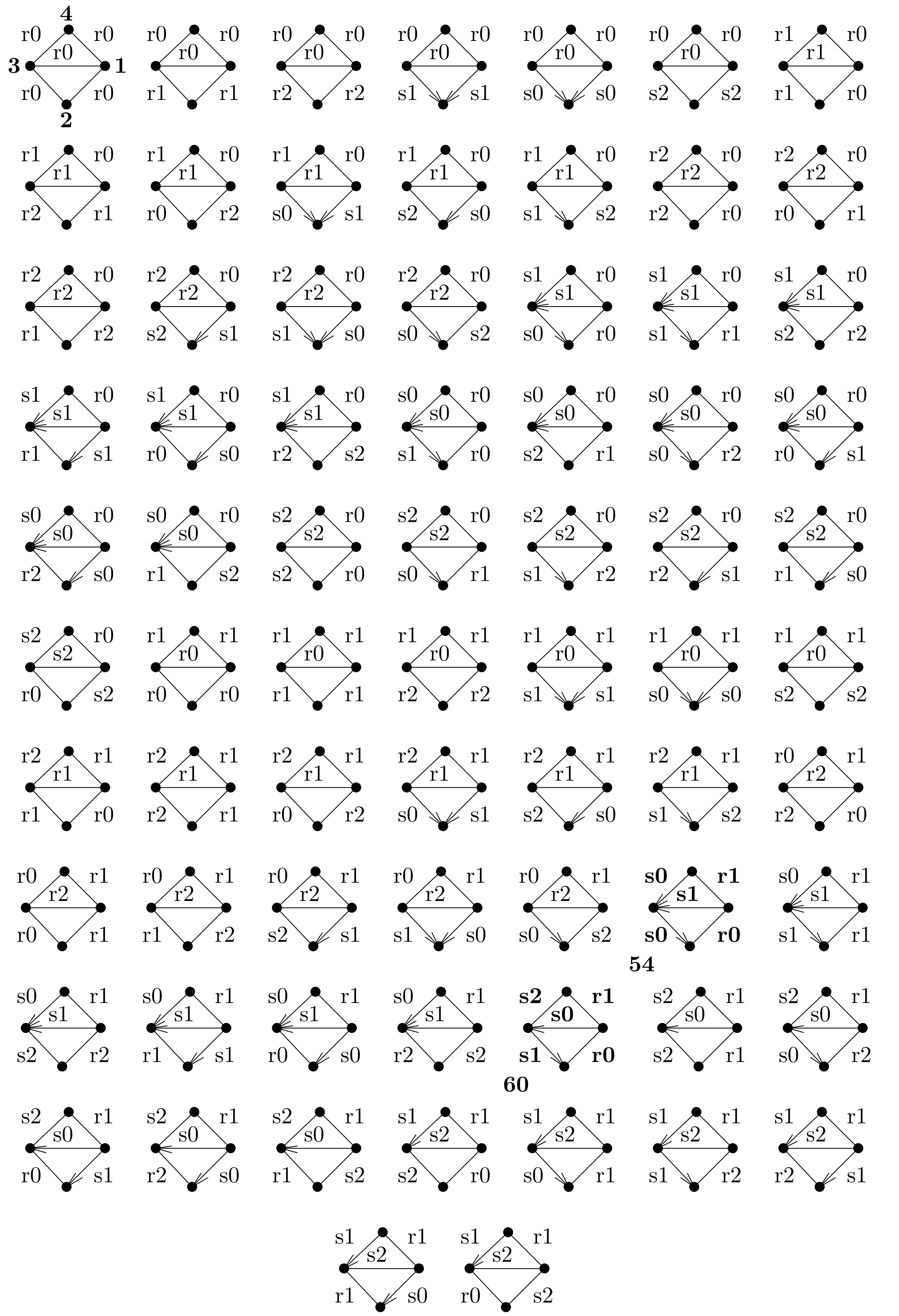


Figure A.8: The first part of $\langle g \rangle_1$

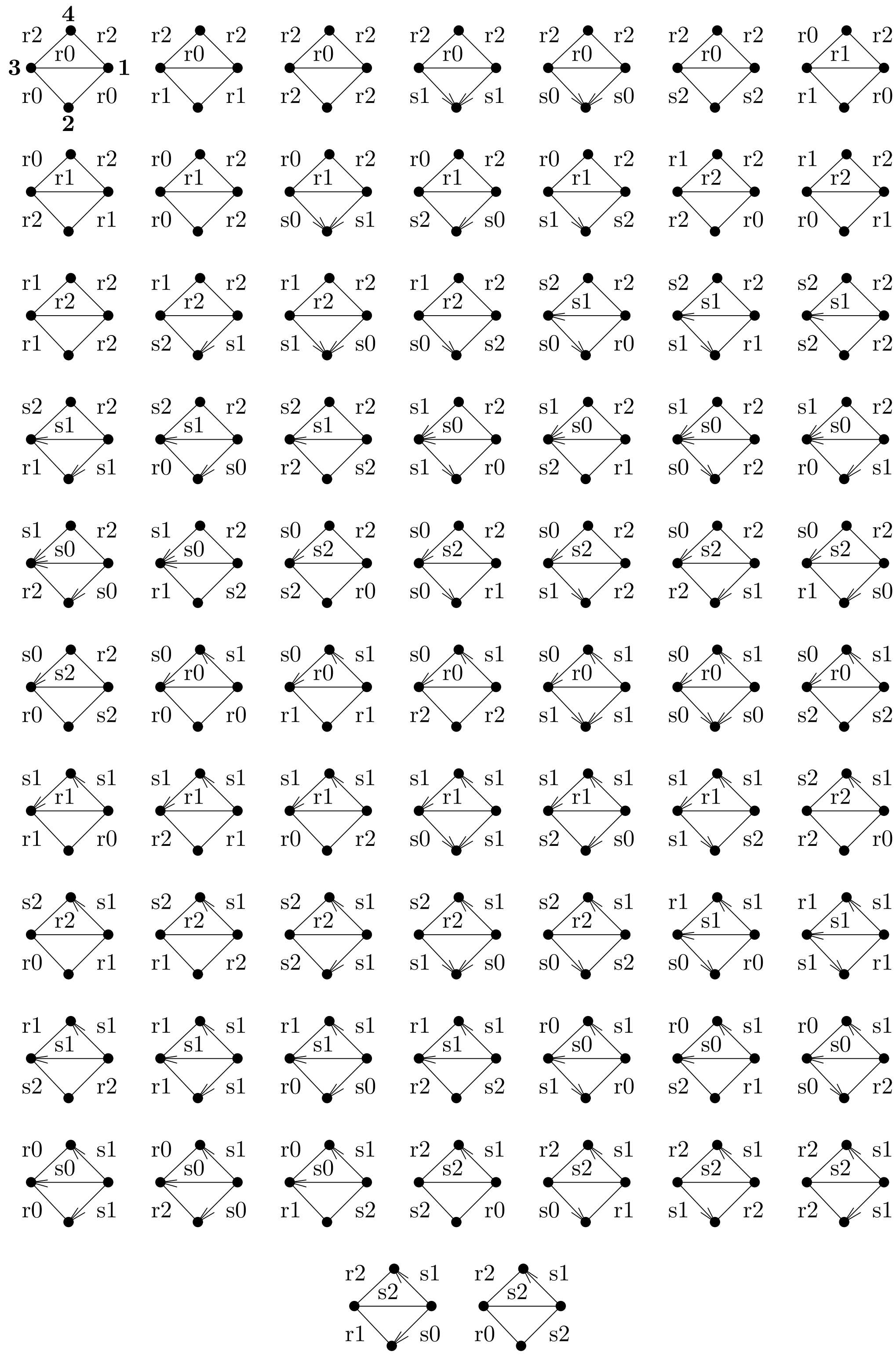


Figure A.8: (Continued)

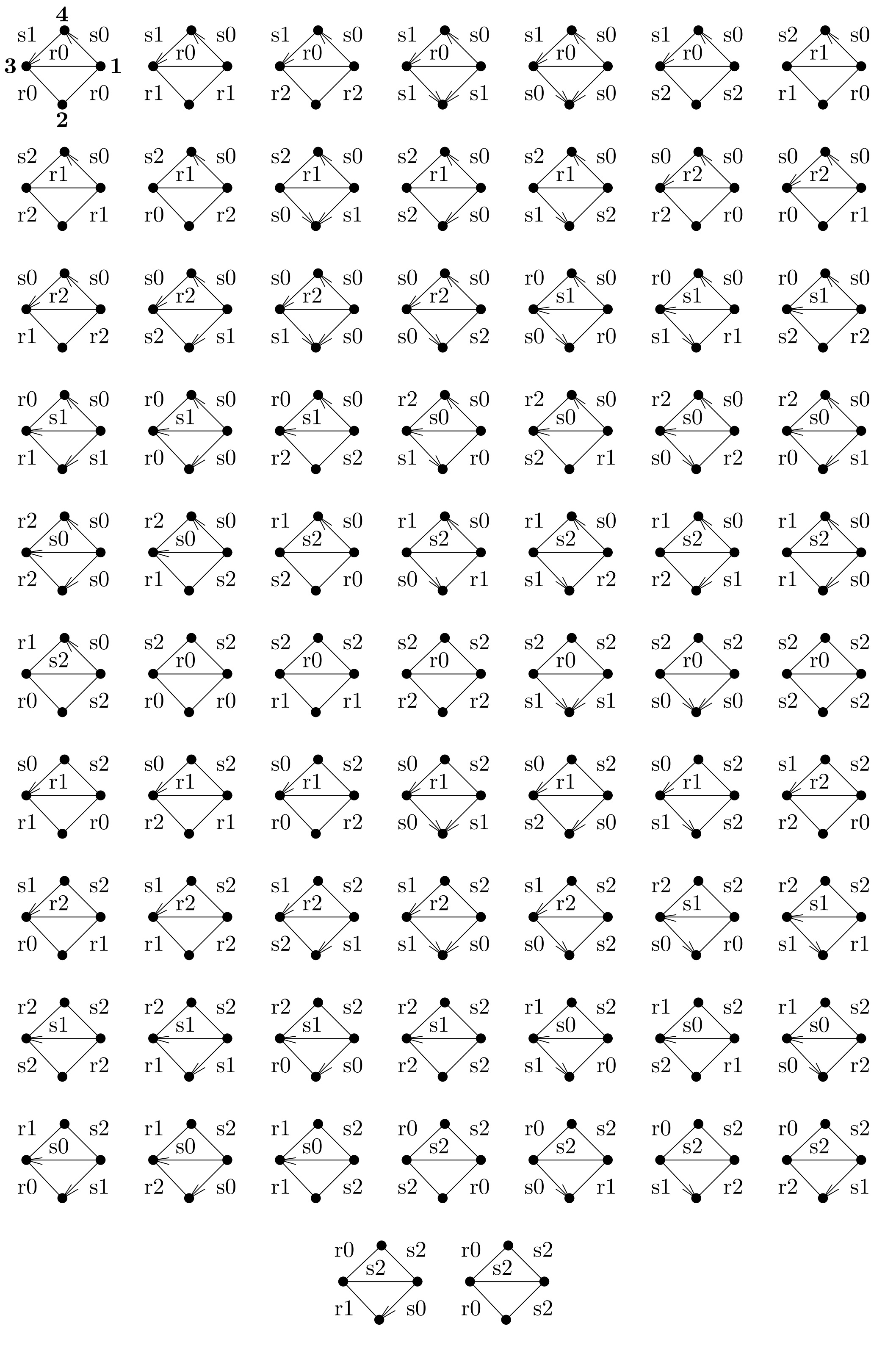


Figure A.8: (Continued)

Appendix B

Researching Switching Classes With Programs

While doing research into switching classes in both the restricted case of undirected graphs and the more general set-up of the second part of this thesis, programs written in `Scheme`, `C` and `Java` were used for various purposes. These were

- helping us to form a hypothesis,
- doing computations that were to be used as part of a result,
- empirically verifying a hypothesis, and
- doing basic computations, that are errorprone when done by hand.

An example of the first concerns the results on the sizes of switching classes with skew gains, where a program helped us to form a hypothesis about them. This was very much in the beginning of the research when the programs were still limited to the language `Scheme`. Another important example is the search we made for critically cyclic switching classes, this time in the language `C`. This is also an example of the second purpose for using a computer program, because one of our theorems uses a result computed by a program.

After forming a hypothesis we sometimes prefer to put some time into writing a program to verify it, especially if a proof is either not vital or would be very cumbersome and detailed. An example in case is the following: in the proof of Theorem 4.14, we did not use a number of the critically cyclic switching classes we had discovered. There can be two reasons for this: we have some more general reasoning that already forbids those switching classes, or we have an omission in our proof. To make as sure as possible, that the former was the case, we hypothesized that some switching classes were unnecessary in the proof, because we proved only that there are no critically cyclic graphs $n \geq 10$ vertices (unless they contain a cycle C_n). It turned out that if we omitted (8-10), (8-13)-(8-15), (9-3), (9-5) from the list of forbidden graphs, and we would run our program to find switching classes on 10 vertices with zero acyclic graphs, then it would not find any. The conclusion was that for each of these forbidden graphs, adding one or two vertices in *any* possible way always results in a graph that has one of the other forbidden graphs, i.e., the ones that were not removed from the list of forbidden graphs.

An example of the fourth case is the construction of the anti-involutions of cyclic groups. It can be done by hand, but it is tedious work and there is room for error. Another example is computation with nonabelian groups.

Also, a hypothesis is often based on a number of examples and if the computation is done by hand, there is a chance that some inadvertent error sneaks in and therefore we may spend a lot of time on a hypothesis that has almost no chance of being correct. Of course, errors in programs are also possible, but they tend to be systematic and therefore easier to spot.

In the following few sections, we shall give some details of the programs, but usually only about *what* they can do, not how.

B.1 The `Scheme` programs

`Scheme` is a functional language of the `Lisp`-type. This means that it is untyped and programs are full of parentheses.
The advantages that made us use `Scheme` were

- familiarity,
- a large set of basic functions that were programmed in the past,
- it is interpreted and therefore flexible,
- high expressive power, and
- easy to work with infinite groups

The strongest disadvantages are

- a large set of functions makes it difficult to know what has been programmed,
- the programs are often quite abstract and therefore hard to read, and
- slow computation

The features of the functions written in `Scheme` that were put to good use during our research were as follows. Programs in `Scheme` are quickly and easily written, although they are generally not very efficient. However, this rapid prototyping makes it possible to easily verify computations done either by hand or by a program written in, for instance, `C`. The programs are efficient to write, but not to run and depending on where the bottleneck was, we decided what to do in `C` and what to do in `Scheme`.

Another big advantage of `Scheme` is that if one has computed something, say a set of graphs, then one can at that point decide what to do with it. First, for instance, one can count how many there are by computing the length of the result. If the number happens to be very large and this is unexpected, then one can inspect the result and decide on the fly what to do: it often happened that a number of elements in the list were "superfluous", e.g., not up to isomorphism. The superfluous ones could then "easily" be removed by utilizing an often small and quickly written function. The great advantage is that because research is essentially open ended, it is good to have programs that are open ended. In fact, if a large number of small things have to be done, and we do not yet know in which way or in which order, a compiler is less useful than an interpreter, especially since most serious compilers take their time compiling.

At some point functions were written that converted graphs (in the chosen `Scheme` format) to LaTeX, which could then be compiled into a paper or simply into a `.dvi` file for inspection. The switching classes in this thesis were made using these functions. The programs were quite flexible, although they did not try to make any pretty graph layouts, which for switching classes is quite unnecessary. Another example of the use of this "pretty-printing" is that for the critically cyclic graphs found by the `C` program we wanted to have them drawn including all graphs with a horizon, a minimum size graph and a maximum size graph in their switching class. This made it easier to discover similarities. Doing it automatically avoided the simple, but troublesome, errors when drawing the graphs by hand.

In the theory of skew gain graphs we want to work with arbitrary groups, arbitrary anti-involutions. In some cases the groups are infinite. We could cope with this in `Scheme` quite easily using characteristic functions instead of sets (for the carrier). In fact, both implementations are possible at this moment and this is almost transparent to the programmer/user. In a functional language it is also easy to construct the direct product of two groups and more such things. This makes it very useful in research into skew gain graphs.

B.2 The `C` programs

When we were searching for the critically cyclic graphs, we decided that the programs in `Scheme` were too slow to be of any help. Speeding up the computations by factors of a few hundreds after we turned to plain `C`, we were able to do an exhaustive search for the critically cyclic graphs up to 12 vertices. The results enabled us to pose the hypothesis, that after 9 vertices only the simple cycles are left.

Hence for our proof we needed only the critically cyclic graphs for $n \leq 9$. To compute the cases for $n = 9$ is, of course, much less time consuming than for $n = 10$. At first we constructed all graphs for a certain number of vertices, but later we found and used Spence's files [45] which list representatives for the switching classes up to isomorphism and up to complementation for up to 10 vertices.[1] In this way for 9 vertices this resulted in 2038 switching classes (up to isomorphism). This is a small number for a computer.

Let $S(n)$ be the number of switching classes on n vertices up to isomorphism. The table of Mallows and Sloane:

n	1	2	3	4	5	6	7	8	9	10	11	12
S(n)	1	1	2	3	7	16	54	243	2.038	33.120	1.182.004	87.723.296

The table (which extends to $n = 21$ in Mallows and Sloane [39]) is counted using Robinson's formula for even graphs. For more computational information on switching classes, see Bussemaker, Mathon and Seidel [6].

To be able to use both the LaTeX conversion functions of `Scheme`, the programs in `C` had four output options: the upper part of the adjacency-matrix (coded with $+$ and $-$), `Scheme` graphs in both a format for further computation in `Scheme` and the "drawing" format for conversion to LaTeX, and a last format which was that of a graph in `C`-code. In the latter format, the code constructs the corresponding graph in `C`. Because often graphs that were found to be critically cyclic were to be forbidden in switching classes of higher order, we had to put the graphs we found into the program itself. It turned out that some errors were made in the process until we included a function that outputted the `C`-code itself, which could be put into the `C`-program directly. Later a program was made that could convert between these formats.

[1]Some care must be taken if you simply complement all the graphs in these files, because there are a few graphs whose complemented switching class is isomorphic to the one they are in.

Using the files of two-graphs we could compute for much higher orders. However, the files were only for switching classes up to 10 vertices (because of size considerations). For more vertices we were forced to construct all possible extensions of the graphs on 10 to graphs of $n > 10$ vertices. This is not so nice, but much better than constructing *all* switching classes on n vertices. However, it has to be taken into account that the results are not anymore up to isomorphism. This is why a separate program was written to remove isomorphic copies of graphs from a file of graphs.

The datastructure chosen in `C` for graphs is the simple adjacency matrix with a maximum on the order of the graph. This turned out not to be a problem: with switching the number of vertices stays the same, we were generally not interested in large switching classes, and the number of edges varies considerably. Hence adjacency lists are not very useful here. Performing a switch on a matrix, simply entails the complementation of all rows and columns belonging to elements in the switching set. With the help of the result of Section A.3 this became a very efficient way of generating all the graphs in the switching class.

Although we have not seriously started yet, it may very well be possible to write code for investigating skew gain graphs in `C++`. Up to this point this has not been done, because in that field, speed of computation has not yet been very important.

B.3 A `Java`-applet

Switching is a simple computation, but it is certainly possible to make errors. If one wants to find out quickly whether a graph switches into an isomorphic copy of another, it is useful to have some interactive computer support.

For this and other reasons a small `Java`-program was written that enables the user to draw and switch graphs interactively. Especially when writing the material for the critically cyclic graphs it helped us a lot.

A disadvantage is still that it only works for switching classes of undirected graphs. For arbitrary groups with anti-involutions, it is not just a matter of programming it: the problem of easily selecting the correct values in the vertices has to be addressed as well.

The `Java`-applet can be seen at work at the following URL:

`http://www.cs.uu.nl/people/jur/2s.html`

Bibliography

[1] R.P. Abelson and M.J. Rosenberg. Symbolic psycho-logic: A model of attitudinal cognition. *Behavioral Sci.*, 3:1 – 13, 1958.

[2] D. Acharya. On characterizing graphs switching equivalent to acyclic graphs. *Indian J. Pure Appl. Math*, 12:1187–1191, 1981.

[3] S.N. Afriat. *The Ring of Linked Rings*. Duckworth, 1982.

[4] J.A. Bondy. Pancyclic graphs I. *J. Combin. Theory, Ser. B*, 11:80 – 84, 1971.

[5] J.A. Bondy and U.S.R. Murty. *Graph Theory with Applications*. North-Holland, 1976.

[6] F.C. Bussemaker, R.A. Mathon, and J.J. Seidel. Tables of two-graphs. In S.B. Rao, editor, *Combinatorics and Graph Theory - Proc. Sympos., Calcutta, 1980)*, volume 885 of *Lecture Notes in Computer Science*, pages 70–112. Springer-Verlag, 1981.

[7] P.J. Cameron. Cohomological aspects of two-graphs. *Math. Z.*, 157:101 – 119, 1977.

[8] P.J. Cameron. Two-graphs and trees. *Discrete Math.*, 127:63 – 74, 1994.

[9] D.G. Corneil and R.A. Mathon, editors. *Geometry and Combinatorics: Selected Works of J.J. Seidel*. Academic Press, Boston, 1991.

[10] G.A. Dirac. Some theorems on abstract graphs. *Proc. London Math. Soc., Ser. 3*, 2:69 – 81, 1952.

[11] A. Ehrenfeucht, J. Hage, T. Harju, and G. Rozenberg. Complexity problems in switching classes of graphs. Technical Report 15, Leiden University, Department of Computer Science, 1997.

[12] A. Ehrenfeucht, J. Hage, T. Harju, and G. Rozenberg. Complexity issues in switching of graphs. In H. Ehrig, G. Engels, H.-J. Kreowski, and G. Rozenberg, editors, *Theory And Application Of Graph Transformations - TAGT '98*, volume 1764 of *Lecture Notes in Computer Science*, pages 59–70, Berlin, 2000. Springer-Verlag.

[13] A. Ehrenfeucht, J. Hage, T. Harju, and G. Rozenberg. Pancyclicity in switching classes. *Inf. Proc. Letters*, 73(5–6):153 – 156, 2000.

[14] A. Ehrenfeucht, T. Harju, and G. Rozenberg. *The Theory of 2-Structures*. World Scientific, 1999.

[15] A. Ehrenfeucht and G. Rozenberg. An introduction to dynamic labeled 2-structures. In A.M. Borzyszkowski and S. Sokolowski, editors, *Mathematical Foundations of Computer Science 1993*, volume 711 of *Lecture Notes in Computer Science*, pages 156–173, Berlin, 1993. Springer-Verlag.

[16] A. Ehrenfeucht and G. Rozenberg. Dynamic labeled 2-structures. *Mathematical Structures in Computer Science*, 4:433–455, 1994.

[17] A. A. Fridman. On the relation between the word problem and the conjugacy problem in finitely defined groups (russian). *Trudy Moskov. Mat. Obsc.*, 9:329–356, 1960.

[18] M.R. Garey and D.S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* Freeman, 1979.

[19] F. Gray. Pulse code communication, Mar. 17 1953. U.S. patent no. 2,632,058.

[20] J.L. Gross and T.W. Tucker. *Topological Graph Theory.* Wiley, New York, 1987.

[21] J. Hage. The membership problem for switching classes with skew gains. *Fundamenta Informaticae*, 39(4):375–387, 1999.

[22] J. Hage and T. Harju. Discussions in a cafeteria, 1996 – 2000.

[23] J. Hage and T. Harju. The size of 2-classes in group labeled 2-structures. Technical Report 17, Leiden University, Department of Computer Science, 1996. Generalized in [26].

[24] J. Hage and T. Harju. Acyclicity of switching classes. *European J. Combin.*, 19:321–327, 1998.

[25] J. Hage and T. Harju. A characterization of acyclic switching classes using forbidden subgraphs. Technical Report 5, Leiden University, Department of Computer Science, 2000. Submitted to Siam J. Disc. Math.

[26] J. Hage and T. Harju. The size of switching classes with skew gains. *Discrete Math.*, 215:81 – 92, 2000.

[27] F. Harary. On the notion of balance of a signed graph. *Michigan Math. J.*, 2:143–146, 1953–1954. Addendum in same journal preceding page 1.

[28] F. Harary. *Graph Theory.* Addison Wesley, 1972.

[29] G.H Hardy and E.M. Wright. *An Introduction to the Theory of Numbers.* Oxford Science Publications, Hong Kong, 5th edition, 1983.

[30] A. Hertz. On perfect switching classes. *Discrete Applied Math.*, 89:263–267, 1998.

[31] A.M. Hinz. The Towers of Hanoi. *Enseign. Math.*, 2:289–321, 1989.

[32] J.E. Hopcroft and R.E. Tarjan. Efficient planarity testing. *J. ACM.*, 21:549 – 568, 1974.

[33] C.F. Miller III. *On Group-Theoretic Decision Problems and Their Classification.* Number 68 in Annals of Mathematics studies. Oxford University Press, 1971.

[34] J. Kratochvíl. Private communication.

[35] J. Kratochvíl, J. Nešetřil, and O. Zýka. On the computational complexity of Seidel's switching, in: Combinatorics, Graphs and Complexity (M.Fiedler and J.Nešetřil eds.) Proceedings 4th Czechoslovak Symposium on Combinatorics, Prachatice 1990. *Annals of Discrete Math.*, 51:161 – 166, 1992.

[36] M.-C. Li, D. Corneil, and E. Mendelsohn. Pancyclicity and NP-completeness in planar graphs. *Discrete Applied Math.*, 98:219 – 225, 2000.

[37] J.H. van Lint and J.J. Seidel. Equilateral points in elliptic geometry. In *Proc. Kon. Nederl. Acad. Wetensch., Ser. A*, volume 69, pages 335 – 348, 1966. Reprinted in [9].

[38] W. Magnus, A. Karras, and D. Solitar. *Combinatorial Group Theory.* Dover, New York, 2nd edition, 1976.

[39] C.L. Mallows and N.J.A. Sloane. Two-graphs, switching classes and Euler graphs are equal in number. *SIAM J. Appl. Math*, 28:876 – 880, 1975.

[40] P.S. Novikov. Unsolvability of the conjugacy problem in the theory of groups (russian). *Izv. Akad. Nauk SSSR, Ser. Mat.*, 18, 1954.

[41] J.J. Rotman. *The Theory of Groups.* Allyn and Bacon, Boston, 2nd edition, 1973.

[42] J.J. Seidel. Graphs and two-graphs. In *Proc. 5th Southeastern Conf. on Combinatorics, Graph Theory, amd Computing*, Winnipeg, Canada, 1974. Utilitas Mathematica Publishing Inc.

[43] J.J. Seidel. A survey of two-graphs. In *Colloquio Internazionale sulle Teorie Combinatorie (Rome,1973)*, volume I, pages 481–511, Rome, 1976. Acc. Naz. Lincei. Reprinted in [9].

[44] J.J. Seidel and D.E. Taylor. Two-graphs, a second survey. In L. Lovasz and V.T. Sós, editors, *Algebraic Methods in Graph Theory (Proc. Internat. Colloq., Szeged, 1978)*, volume II, pages 689–711, Amsterdam, 1981. North-Holland. Reprinted in [9].

[45] E. Spence. Tables of two-graphs. http://gauss.maths.gla.ac.uk/˜ted/.

[46] D. Wood. The Towers of Brahma and Hanoi revisited. *J. Recreational Math.*, 14:17–24, 1981-82.

[47] M. Yannakakis. Node- and edge-deletion NP-complete problems. In *Proc. 10th Ann. ACM Symp. on theory of Computing*, pages 253 – 264, New York, 1978. ACM.

[48] T. Zaslavsky. Characterizations of signed graphs. *J. Graph Theory*, 5:401–406, 1981.

[49] T. Zaslavsky. Signed graphs. *Discrete Applied Math.*, 4:47–74, 1982. Erratum on p. 248 of volume 5.

[50] T. Zaslavsky. Biased graphs. I. Bias, balance, and gains. *J. Combin. Theory, Ser. B*, 47:32–52, 1989.

[51] T. Zaslavsky. *A Mathematical Bibliography of Signed and Gain Graphs and Allied Areas*, volume DS8 of *Dynamic Surveys in Combinatorics*. World Combinatorics Exchange, http://www.combinatorics.org, 1999.

Index

Samenvatting

De motivatie voor het onderzoek dat in dit proefschrift aan de orde komt is het modelleren van het gedrag van sommige netwerken van processoren. Kortgezegd is een netwerk van processoren een verzameling van processoren waartussen verbindingen kunnen bestaan waarbij de toestand van het netwerk bevat is in de waarden op die verbindingen (in dit proefschrift hebben die waarden de structuur van een groep, zoals bekend uit de wiskunde).

In een dergelijk netwerk kan een processor asynchroon besluiten een actie uit te voeren waarbij hij slechts in staat is de waarden op zijn inkomende en uitgaande verbindingen op een zekere vastgelegde wijze te veranderen (deze operatie heet "switchen"). Gegegeven een willekeurige begintoestand van het netwerk is de switching class horende bij dit netwerk, de verzameling van alle toestanden die door middel van het uitvoeren van deze switch-operaties te verkrijgen zijn.

Het model komt voort uit drie eisen die opgelegd zijn aan deze netwerken: elke willekeurige verbinding kan door middel van een actie in een processor in een willekeurige toestand gebracht worden, een tweetal van dergelijke acties kan worden gecombineerd door middel van één enkele, en als twee processoren, verbonden via een zekere connectie, beide een actie uitvoeren, dan zal de volgorde waarin dit gebeurt niet de uitkomst van de combinatie beïnvloeden.

Bovenstaand model werd door Ehrenfeucht en Rozenberg uitgewerkt onder de naam dynamische gelabelde 2-structuren en het onderzoek voor dit proefschrift bestond uit het analyseren van deze structuren (zowel algoritmisch, combinatorisch als algebraïsch). De terminologie in dit proefschrift is aangepast aan die, die te vinden was in de wiskunde sinds eind jaren zestig, toen eenvoudiger versies van het model geïntroduceerd werden.

Opvallend was dat de vragen die men toen stelden en die wij in dit proefschrift aan de orde stellen, door verschil in motivatie sterk uiteen liepen. Het is echter wel zo dat de behaalde resultaten binnen de context van het toenmalige onderzoek geplaatst kan worden.

Het onderzoek op het gebied van het algemene model spitste zich toe op vinden van een efficiënt algoritme dat kon beslissen of vanuit een gegeven toestand van een netwerk, een gegeven andere bereikt kon worden door middel van een switch. Een dergelijk algoritme bleek te bestaan en de benodigde theorie bleek belangrijk te zijn voor het begrip van switching classes.

Voor de simpele versie van switching classes zoals beschreven in het eerste deel van dit proefschrift is vooral gewerkt aan combinatorische problemen, zoals het bewijs dat een switching class slechts één boom kan bevatten (modulo isomorfisme). Ter verkenning van deze combinatorische problemen, is tijdens het onderzoek veelvuldig gebruik gemaakt van programma's geschreven in `C` of `Scheme`.

Curriculum Vitae

Jurriaan Hage werd geboren op 19 september 1969 te Alphen aan den Rijn. Na de openbare lagere school deed hij eerst HAVO en vervolgens VWO aan het Albanianae in diezelfde plaats. Vervolgens begon hij aan een studie Informatica in Leiden die hij na zes jaar cum laude afrondde. Tijdens zijn studie deed hij vier jaar werkgroepassistentie. Zijn afstudeerwerk deed hij bij prof. dr. A. Ollongren op het gebied van vertaling en programmeertaalontwerp.

Na zijn afstuderen deed hij van 1995 tot en met 1998 onderzoek als aio in de groep van prof. dr. G. Rozenberg, alweer aan de Universiteit Leiden (toen nog Rijksuniversiteit). De begeleider bij het onderzoek was dr. Tero Harju uit Finland. Het onderwerp van het onderzoek betrof switching classes, een wiskundig model voor netwerken van processoren; een model dat zoals later bleek ook terug gevonden werd in de wiskunde van enige decennia eerder.

Sinds 1 november 1999 heeft hij een betrekking als docent/onderzoeker in Utrecht in de groep Software Technology van prof. dr. Doaitse Swierstra.

Titles in the IPA Dissertation Series

J.O. Blanco. *The State Operator in Process Algebra.* Faculty of Mathematics and Computing Science, TUE.1996-1

A.M. Geerling. *Transformational Development of Data-Parallel Algorithms.* Faculty of Mathematics and Computer Science, KUN.1996-2

P.M. Achten. *Interactive Functional Programs: Models, Methods, and Implementation.* Faculty of Mathematics and Computer Science, KUN.1996-3

M.G.A. Verhoeven. *Parallel Local Search.* Faculty of Mathematics and Computing Science, TUE.1996-4

M.H.G.K. Kesseler. *The Implementation of Functional Languages on Parallel Machines with Distrib. Memory.* Faculty of Mathematics and Computer Science, KUN.1996-5

D. Alstein. *Distributed Algorithms for Hard Real-Time Systems.* Faculty of Mathematics and Computing Science, TUE.1996-6

J.H. Hoepman. *Communication, Synchronization, and Fault-Tolerance.* Faculty of Mathematics and Computer Science, UvA.1996-7

H. Doornbos. *Reductivity Arguments and Program Construction.* Faculty of Mathematics and Computing Science, TUE.1996-8

D. Turi. *Functorial Operational Semantics and its Denotational Dual.* Faculty of Mathematics and Computer Science, VUA.1996-9

A.M.G. Peeters. *Single-Rail Handshake Circuits.* Faculty of Mathematics and Computing Science, TUE.1996-10

N.W.A. Arends. *A Systems Engineering Specification Formalism.* Faculty of Mechanical Engineering, TUE.1996-11

P. Severi de Santiago. *Normalisation in Lambda Calculus and its Relation to Type Inference.* Faculty of Mathematics and Computing Science, TUE.1996-12

D.R. Dams. *Abstract Interpretation and Partition Refinement for Model Checking.* Faculty of Mathematics and Computing Science, TUE.1996-13

M.M. Bonsangue. *Topological Dualities in Semantics.* Faculty of Mathematics and Computer Science, VUA.1996-14

B.L.E. de Fluiter. *Algorithms for Graphs of Small Treewidth.* Faculty of Mathematics and Computer Science, UU.1997-01

W.T.M. Kars. *Process-algebraic Transformations in Context.* Faculty of Computer Science, UT.1997-02

P.F. Hoogendijk. *A Generic Theory of Data Types.* Faculty of Mathematics and Computing Science, TUE.1997-03

T.D.L. Laan. *The Evolution of Type Theory in Logic and Mathematics.* Faculty of Mathematics and Computing Science, TUE.1997-04

C.J. Bloo. *Preservation of Termination for Explicit Substitution.* Faculty of Mathematics and Computing Science, TUE.1997-05

J.J. Vereijken. *Discrete-Time Process Algebra.* Faculty of Mathematics and Computing Science, TUE.1997-06

F.A.M. van den Beuken. *A Functional Approach to Syntax and Typing.* Faculty of Mathematics and Informatics, KUN.1997-07

A.W. Heerink. *Ins and Outs in Refusal Testing.* Faculty of Computer Science, UT.1998-01

G. Naumoski and W. Alberts. *A Discrete-Event Simulator for Systems Engineering.* Faculty of Mechanical Engineering, TUE.1998-02

J. Verriet. *Scheduling with Communication for Multiprocessor Computation.* Faculty of Mathematics and Computer Science, UU.1998-03

J.S.H. van Gageldonk. *An Asynchronous Low-Power 80C51 Microcontroller.* Faculty of Mathematics and Computing Science, TUE.1998-04

A.A. Basten. *In Terms of Nets: System Design with Petri Nets and Process Algebra.* Faculty of Mathematics and Computing Science, TUE.1998-05

E. Voermans. *Inductive Datatypes with Laws and Subtyping – A Relational Model.* Faculty of Mathematics and Computing Science, TUE.1999-01

H. ter Doest. *Towards Probabilistic Unification-based Parsing.* Faculty of Computer Science, UT.1999-02

J.P.L. Segers. *Algorithms for the Simulation of Surface Processes.* Faculty of Mathematics and Computing Science, TUE.1999-03

C.H.M. van Kemenade. *Recombinative Evolutionary Search.* Faculty of Mathematics and Natural Sciences, Univ. Leiden.1999-04

E.I. Barakova. *Learning Reliability: a Study on Indecisiveness in Sample Selection.* Faculty of Mathematics and Natural Sciences, RUG.1999-05

M.P. Bodlaender. *Schedulere Optimization in Real-Time Distributed Databases*. Faculty of Mathematics and Computing Science, TUE.1999-06

M.A. Reniers. *Message Sequence Chart: Syntax and Semantics*. Faculty of Mathematics and Computing Science, TUE.1999-07

J.P. Warners. *Nonlinear approaches to satisfiability problems*. Faculty of Mathematics and Computing Science, TUE.1999-08

J.M.T. Romijn. *Analysing Industrial Protocols with Formal Methods*. Faculty of Computer Science, UT.1999-09

P.R. D'Argenio. *Algebras and Automata for Timed and Stochastic Systems*. Faculty of Computer Science, UT.1999-10

G. Fábián. *A Language and Simulator for Hybrid Systems*. Faculty of Mechanical Engineering, TUE.1999-11

J. Zwanenburg. *Object-Oriented Concepts and Proof Rules*. Faculty of Mathematics and Computing Science, TUE.1999-12

R.S. Venema. *Aspects of an Integrated Neural Prediction System*. Faculty of Mathematics and Natural Sciences, RUG.1999-13

J. Saraiva. *A Purely Functional Implementation of Attribute Grammars*. Faculty of Mathematics and Computer Science, UU.1999-14

R. Schiefer. *Viper, A Visualisation Tool for Parallel Progam Construction*. Faculty of Mathematics and Computing Science, TUE.1999-15

K.M.M. de Leeuw. *Cryptology and Statecraft in the Dutch Republic*. Faculty of Mathematics and Computer Science, UvA.2000-01

T.E.J. Vos. *UNITY in Diversity. A stratified approach to the verification of distributed algorithms*. Faculty of Mathematics and Computer Science, UU.2000-02

W. Mallon. *Theories and Tools for the Design of Delay-Insensitive Communicating Processes*. Faculty of Mathematics and Natural Sciences, RUG.2000-03

W.O.D. Griffioen. *Studies in Computer Aided Verification of Protocols*. Faculty of Science, KUN.2000-04

P.H.F.M. Verhoeven. *The Design of the MathSpad Editor*. Faculty of Mathematics and Computing Science, TUE.2000-05

J. Fey. *Design of a Fruit Juice Blending and Packaging Plant*. Faculty of Mechanical Engineering, TUE.2000-06

M. Franssen. *Cocktail: A Tool for Deriving Correct Programs*. Faculty of Mathematics and Computing Science, TUE.2000-07

P.A. Olivier. *A Framework for Debugging Heterogeneous Applications*. Faculty of Natural Sciences, Mathematics and Computer Science, UvA.2000-08

E. Saaman. *Another Formal Specification Language*. Faculty of Mathematics and Natural Sciences, RUG.2000-10

M. Jelasity. *The Shape of Evolutionary Search Discovering and Representing Search Space Structure*. Faculty of Mathematics and Natural Sciences, UL.2001-01

R. Ahn. *Agents, Objects and Events a computational approach to knowledge, observation and communication*. Faculty of Mathematics and Computing Science, TU/e.2001-02

M. Huisman. *Reasoning about Java programs in higher order logic using PVS and Isabelle*. Faculty of Science, KUN.2001-03

I.M.M.J. Reymen. *Improving Design Processes through Structured Reflection*. Faculty of Mathematics and Computing Science, TU/e.2001-04

S.C.C. Blom. *Term Graph Rewriting: syntax and semantics*. Faculty of Sciences, Division of Mathematics and Computer Science, VUA.2001-05

R. van Liere. *Studies in Interactive Visualization*. Faculty of Natural Sciences, Mathematics and Computer Science, UvA.2001-06

A.G. Engels. *Languages for Analysis and Testing of Event Sequences*. Faculty of Mathematics and Computing Science, TU/e.2001-07

J. Hage. *Structural Aspects of Switching Classes*. Faculty of Mathematics and Natural Sciences, UL.2001-08