Universitat Autònoma de Barcelona

Set of periods, topological entropy and combinatorial dynamics for tree and graph maps

David Juher Barrot

Set of periods, topological entropy and combinatorial dynamics for tree and graph maps

David Juher Barrot

Memòria presentada per aspirar al grau de doctor en ciències matemàtiques.

Departament de matemàtiques de la Universitat Autònoma De Barcelona.

Bellaterra, abril del 2003.

Els Drs. Lluís Alsedà Soler
i Pere Mumbrú Rodríguez
CERTIFIQUEM que aquesta
memòria ha estat realitzada per
David Juher Barrot sota la nostra
direcció, al departament de
matemàtiques de la Universitat
Autònoma de Barcelona.

Bellaterra, abril del 2003.

Per l'Esteve,
des de la Terra

Contents

Introduction

This memoir deals with one-dimensional discrete dynamical systems, from both a topological and a combinatorial point of view. More precisely, we are interested in the periodic orbits and topological entropy of continuous self-maps defined on trees and graphs.

The central problem of our work is the characterization of the possible set of periods of all periodic orbits exhibited by a *tree map* (any continuous map from a tree into itself). The widely known Sharkovskii's Theorem (1964) concerning interval maps was the first remarkable result in this setting. This beautiful theorem states that the set of periods of any interval map is an initial segment of the following linear ordering $\trianglerighteq$ in the set $\mathbb{N} \cup \{2^\infty\}$ (the so-called *Sharkovskii ordering*):

$$3 \trianglerighteq 5 \trianglerighteq 7 \trianglerighteq \dots \trianglerighteq 2 \cdot 3 \trianglerighteq 2 \cdot 5 \trianglerighteq 2 \cdot 7 \trianglerighteq \dots \trianglerighteq 4 \cdot 3 \trianglerighteq 4 \cdot 5 \trianglerighteq 4 \cdot 7 \trianglerighteq \dots \trianglerighteq \dots \trianglerighteq 2^n \cdot 3 \trianglerighteq 2^n \cdot 5 \trianglerighteq 2^n \cdot 7 \trianglerighteq \dots \trianglerighteq 2^\infty \trianglerighteq \dots \trianglerighteq 2^n \trianglerighteq \dots \trianglerighteq 16 \trianglerighteq 8 \trianglerighteq 4 \trianglerighteq 2 \trianglerighteq 1.$$

Conversely, given any initial segment $\mathcal{I}$ of the ordering $\trianglerighteq$ there exists an interval map whose set of periods coincides with $\mathcal{I}$.

During the last three decades there have been several attempts to find results similar to that of Sharkovskii for one-dimensional spaces other than the interval (the 3-star and the circle, among them). More recently, the case of maps defined on more general trees has been specially treated. Baldwin's Theorem (1991), which solves the problem in the case of n-stars for any $n \geq 1$, has been one of the most significant advances in this direction. This result states that the set of periods of any n-star map is a finite union of initial segments of n-many partial orderings (*Baldwin orderings*). Conversely, given such a union $\mathcal{I}$ there exists an n-star map whose set of periods is $\mathcal{I}$.

A more detailed chronology of related works, as well as citations to other partial results on this matter, can be found in the Introductions to Chapters 1 and 2.

The main purpose of our research is to describe the generic structure of the set of periods of any tree map $g \colon S \longrightarrow S$ in terms of the combinatorial and topological properties of the tree S: amount and arrangement of endpoints, vertices and edges. In Chapter 1 we make a detailed discussion about which is the more natural approach to this problem, and we propose a strategy consisting on three consecutive stages which can be summarized as follows:

1. For each periodic orbit P of g, calculate the set Λ_P of periods of the corresponding *canonical* (or P-minimal) model $f_P : T_P \longrightarrow T_P$.
2. Prove that Λ_P is contained in the set of periods of each tree map exhibiting an orbit with the *pattern* of P. In particular, $\Lambda_P \subset \mathrm{Per}(g)$.
3. Consider each orbit P of g and its associated Λ_P, and then obtain (by purely number-theoretical arguments) a finite structure of the set of periods of g by arranging adequately the (perhaps uncountable) union of all sets Λ_P.

Observe that this approach depends strongly on the notions of *pattern* (of a finite invariant set) and *minimal model* associated to it. These notions were developed in the context of interval maps and widely used in a number of papers during the last two decades. However, equivalent operative definitions for tree maps were not available until 1997, when Alsedà, Guaschi, Los, Mañosas and Mumbrú proposed to define the pattern of a finite invariant set P essentially as a homotopy class of maps relative to the points of P, and proved (constructively) that there always exists a P-minimal model $f_P \colon T_P \longrightarrow T_P$, that is, a representative of the class displaying several dynamic minimality properties. It is important to remark that the trees S and T_P are not necessarily homeomorphic. This complicates considerably the implementation of the second stage of the above programme, since the only features which are preserved when one compares the maps $g \colon S \longrightarrow S$ and $f_P \colon T_P \longrightarrow T_P$ are the relative positions of the points of P and the way g and f_P act on these points.

In Chapter 1 we carry out the first stage of the above programme. That is, given a periodic orbit P and a P-minimal tree map $f \colon T \longrightarrow T$, we calculate (as large as possible) subsets of the set of periods of f. This task, which has been done by studying the loops of the Markov P-graph of f, is relatively simple when P does not exhibit a certain rotational (or *twist*) behavior around a fixed point of f. When P is twist, we perform a reduction process consisting of what we have called a *sequence of partial reductions* leading up to a periodic orbit P' and a P'-minimal tree map $f' \colon T' \longrightarrow T'$ such that $T' \subset T$, $|P| = k|P'|$ for some $k > 1$, the set of periods of f is essentially the set of periods of f' multiplied by k, and P' is non-twist. By means of this strategy we prove Theorem A, which states that the set of periods of f is, up to an explicitly bounded finite set, the initial segment of a Baldwin ordering starting at $|P|$. We also prove a converse result (Theorem B) which states that, given any set $\mathcal{I}$ of that form, there exists a piecewise monotone tree map whose set of periods coincides with $\mathcal{I}$.

The goal of Chapter 2 is to implement in full the above programme by completing stages 2 and 3. In June 2001 we submitted the work of Chapter 1 to be considered for publication as a paper in International Journal of Bifurcation and Chaos ([5]). Later on, while writing a part of Chapter 2 of this memoir, we realized that using a new simple and powerful argument would allow us to shorten considerably the proofs and improve the obtained results. In particular, with this new approach all the lengthy technical work associated to the construction of a *sequence of partial reductions* is unnecessary. This gave rise to a revised version of the above strategy (with a slightly modified stage 1) which we perform completely in Chapter 2. Despite this new approach overcomes a part of the material of Chapter 1, we have chosen to leave intact the published work.

The main result of Chapter 2 is Theorem C, which tells us that for each tree map $g\colon S \longrightarrow S$ there exists a finite set of sequences $\underline{\mathsf{s}} = (p_1, p_2, \ldots, p_m)$ of positive integers such that the set of periods of g is (up to an explicitly bounded finite set) a finite union of sets of the form

$$\{p_1, p_1p_2, \ldots, p_1p_2\cdots p_{m-1}\} \cup (\mathcal{I}_{\underline{\mathsf{s}}} \setminus p_1p_2\cdots p_m\{2, 3, \ldots, \lambda_{\underline{\mathsf{s}}}\}),$$

where $\lambda_{\underline{\mathsf{s}}}$ is a nonnegative integer and $\mathcal{I}_{\underline{\mathsf{s}}}$ is an initial segment of the Baldwin ordering ${}_{p_1p_2\cdots p_m}\!\geq$. The finite set of sequences which characterizes the set of periods of g depends entirely on the combinatorial properties of the tree S. We also prove a converse result (Theorem D) which asserts that given any finite union $\mathcal{I}$ of sets of the above form there exists a tree map whose set of periods is $\mathcal{I}$.

In Chapter 3 we report some computer experiments on the minimality of the dynamics of canonical models. Chronologically, this work is contemporaneous to Chapter 1. While researching about the set of periods of canonical models, we constructed some computer software to explore how the dynamic minimality translates into some forcing properties of patterns and periods. In a spirit of modular programming, we designed lots of self-contained functions which can be used to implement a wide variety of several-purpose software. Among other, we have functions that:

1. Compute the canonical model of a pattern provided by the user.
2. Calculate the Markov transition matrix associated to a piecewise monotone tree map.
3. Extract all the simple loops of a given length from a Markov transition matrix.
4. Calculate the pattern of a periodic orbit associated to a Markov loop.

The efficient programming of a part of this machinery needs an important theoretical background. In Chapter 3 we list and explain the source code (written in language C) of the most important functions. When required, we also state and prove some results which have been used either to construct the algorithms or to optimize the execution time. The code of other minor routines, which are not interesting from a mathematical point of view, has been listed in the Appendix.

Finally, in Chapter 4 we generalize some results of Block & Coven, Misiurewicz & Nitecki and Takahashi, where the topological entropy of an interval map was approximated by the entropies of its periodic orbits (the entropy of a periodic orbit P, denoted by $h(P)$, is the entropy of a P-minimal model). In Theorem E we show that if $f: G \longrightarrow G$ is a graph map then the entropy of f equals $\sup\{h(P) : P \text{ periodic orbit of } f \text{ and } |P| > m\}$, for each non-negative integer m. This chapter has been published as a paper in Proceedings of the American Mathematical Society ([4]).

Agraïments.

En aquest món que els mitjans de comunicació qualifiquen (amb grans dosis d'humor negre o de mala fe) de globalitzat i multicultural, és un autèntic luxe poder escriure en català l'únic fragment d'aquesta tesi que serà llegit per tothom.

Primer de tot he de donar les gràcies als doctors Lluís Alsedà i Pere Mumbrú. D'aquests dos grans generadors d'energia positiva n'he admirat el rigor extrem i la fina ironia, la feina ben feta i el bon humor, la disciplina de treball fèrria i la immensa humanitat. Penseu-hi: aquests parells d'atributs sovint s'exclouen mútuament. Aplegar-los tots alhora és una qualitat només atribuïble als genis.

Totes aquestes persones, sovint sense ser-ne conscients, m'han ajudat a escriure la tesi: Rupert i Carme, Cristina i Jordi, Jaume, Natàlia, Clara, Esteve, Isolda i avi Esteve (Girona), Francesc (Creixell de Mar), Martha Álvarez (México D.F.), Ricard i Georgina (Barcelona), Prat, Luiiiiis, Maria, Enric, Mónica Vásquez, Argi, Narcís, JR, Marta i Mante (Girona), El Exorcista III (Georgetown), Jordi i Anna (Sant Cugat-Santa Coloma-Cerdanyola), Víctor i Anna (Sant Quirze-Cassà-Sabadell), Fina (Torroella-Ullà), Núria Flores (Cerdanyola), Carina (Girona), Pipo i Eli (Canet-Barcelona), Marta Fraiz (Barcelona), Fèlix Gurucharri (Barcelona), Anna Montanyà (Terrassa), Montse Vilardell (Girona), Sergio Crespo (Lleida), Jaume Soler (La Garriga), Joaquim Gelabertó (Girona), Joan Miró (Girona), Pepus, Mei, Glòria, Santi, Martin, Robert, Esther, Jaume Romero, Carles, Vera, Jordi, Raimon, Joan, Marta, Àngel, Marc, Roel, Maria (UdG).

Chapter 1

Sets of periods for piecewise monotone tree maps

1.1 Introduction

In this chapter we deal with the problem of determining which are the possible sizes of the periodic orbits that appear by iterating a continuous map defined on a tree. For some particular cases (interval and star), several well known results establish that if a continuous map exhibits a periodic orbit which verifies some combinatorial properties then we can determine a set which is a lower bound of the set of periods of the map.

The widely known Sharkovskii's Theorem (see [42]) studying the set of periods of any continuous map from an interval of the real line into itself was the first remarkable result in this setting. In order to state it, we introduce the Sharkovskii ordering $\trianglerighteq$ (the symbols $\trianglelefteq$, $\triangleleft$ and $\triangleright$ will be understood in the natural way) in the set $\mathbb{N} \cup \{2^\infty\}$:

$3 \trianglerighteq 5 \trianglerighteq 7 \trianglerighteq \ldots \trianglerighteq 2 \cdot 3 \trianglerighteq 2 \cdot 5 \trianglerighteq 2 \cdot 7 \trianglerighteq \ldots \trianglerighteq 4 \cdot 3 \trianglerighteq 4 \cdot 5 \trianglerighteq 4 \cdot 7 \trianglerighteq \ldots \trianglerighteq \ldots \trianglerighteq$
$2^n \cdot 3 \trianglerighteq 2^n \cdot 5 \trianglerighteq 2^n \cdot 7 \trianglerighteq \ldots \trianglerighteq 2^\infty \trianglerighteq \ldots \trianglerighteq 2^n \trianglerighteq \ldots \trianglerighteq 16 \trianglerighteq 8 \trianglerighteq 4 \trianglerighteq 2 \trianglerighteq 1.$

The Sharkovskii's theorem states that if an interval map f has a periodic orbit of period m then f has periodic orbits of period k for each $m \trianglerighteq k$. As a consequence, it can be shown that for each interval map f there exists some $n \in \mathbb{N} \cup \{2^\infty\}$ verifying that the set of periods of f is exactly the set of integers k such that $n \trianglerighteq k$. Conversely, given any $n \in \mathbb{N} \cup \{2^\infty\}$ there exists an interval map g whose set of periods is the set of all integers k such that $n \trianglerighteq k$.

During the last three decades there have been several attempts to find results similar to that of Sharkovskii for 1-dimensional spaces other than the interval (see for instance [7] about maps on Y or [28], [21], [19] and [37] about circle maps). More recently, the case of maps defined on trees has been specially treated.

In [16] the characterization of the set of periods of any continuous map defined on an r-star (a tree with r edges and r endpoints) is given in terms of finitely many partial orderings. Let us define the Baldwin partial orderings ${}_p\!\geq$ for all $p \in \mathbb{N}$ (the symbols $<_p$, $\leq_p$ and ${}_p\!>$ will be understood in the natural way). If $p = 1$ then ${}_p\!\geq$ is the Sharkovskii ordering. For $p > 1$ and $k, m \in \mathbb{N} \cup \{p2^\infty\}$, we write $m \;{}_p\!\geq k$ if one of the following cases holds:

(i) $k = 1$ or $k = m$

(ii) $k, m \in p\mathbb{N} \cup \{p2^\infty\}$ and $m/p \rhd k/p$

(iii) $k \in p\mathbb{N} \cup \{p2^\infty\}$ and $m \notin \{1\} \cup p\mathbb{N} \cup \{p2^\infty\}$

(iv) $k, m \notin \{1\} \cup p\mathbb{N} \cup \{p2^\infty\}$ and $k = im + jp$ with $i, j \in \mathbb{N}$

where the arithmetic rule $p2^\infty/p = 2^\infty$ is assumed and $p\mathbb{N}$ stands for $\{pn : n \in \mathbb{N}\}$. It is not difficult to see that ${}_2\!\geq$ also coincides with the Sharkovskii ordering.

In Baldwin's paper, a positive integer is associated to each periodic orbit P of an r-star map f. This integer is called the *type* of P and depends only on the combinatorics of $f|_P$ (in Section 1.4 a precise definition is given for a general tree map). Baldwin proves that if f has a periodic orbit of period m and type p then f has periodic orbits of period k for each $m \;{}_p\!\geq k$.

An *initial segment* of the ordering ${}_p\!\geq$ is defined to be any set $\mathcal{S}$ such that if $m \in \mathcal{S}$ and $m \;{}_p\!> k$ then $k \in \mathcal{S}$. Baldwin proves that the set of periods of any r-star map is a union of finitely many initial segments of the orderings ${}_p\!\geq$ for $1 \leq p \leq r$. Conversely, given such a union $\mathcal{A}$ there exists an r-star map whose set of periods is $\mathcal{A}$.

In what follows, any continuous map from a tree into itself will be called a *tree map*.

The characterization of the set of periods for any tree map $f : T \longrightarrow T$ in terms of some constants which depend on the topological structure of T (such as the amount of vertices or endpoints of T) is yet an open problem. However, there are some partial results in this direction (see, for instance, [31], [14], [36] and [23]).

A natural strategy to obtain this kind of characterization for interval and star maps, that already has been used in the proofs of Sharkovskii and Baldwin theorems, is the following one. Assume that f is an interval map or an r-star map and let P be a periodic orbit of f. The first stage of the strategy consists of studying the subset Λ_P of periods of f which are *forced by the pattern of* P. That is, one wants to know which other orbits the map f will necessarily have, depending only on the combinatorics of $f|_P$. To solve this problem one replaces f by another map g such that $g|_P = f|_P$ and g is monotone between any two consecutive points of P. It can be seen that such a map is the dynamically simplest model which exhibits an orbit having the pattern of P. This means that each pattern exhibited by g is also exhibited by f and that the set Λ_P coincides with the set of periods of g. Therefore, the set Λ_P can be computed just by studying the loops of the Markov graph of g. The last step of the proof consists in considering each orbit P of f and its associated Λ_P. Then one gets the structure of the set of periods of f by obtaining the structure of the (uncountable) union of all sets Λ_P. This is done by purely number-theoretical arguments.

As it has been said before, an important intermediate step in getting the periodic structure of interval and star maps is the study of the set of periods of these (piecewise monotone) "dynamically simplest models". Since, in addition, piecewise monotone maps provide all the necessary examples in the "converse part" of the theorems of Sharkovskii and Baldwin, the proofs of these results are strongly based on the study of this class of maps.

To study the set of periods of tree maps we have chosen to follow a strategy similar to the one described above (as we shall see, this is a natural strategy also in the case of tree maps). However, it turns out that the straightforward implementation of this strategy to tree maps does not work. Indeed, let $f : T \longrightarrow T$ be a tree map, let P be a periodic orbit of f and let V denote the set of vertices of T. Then we want to consider a *P-weakly monotone map* g which is defined to coincide with f on $V \cup P$ and is monotone (injective) on the closure of each connected component of $T \setminus (V \cup P)$. The problem is that a P-weakly monotone map can have (even infinitely many) periods which are not periods of f, and thus it cannot be our desired "minimal model". To illustrate this phenomenon consider the following simple example in the case of interval maps.

Example 1.1.1. Let $g : [0,1] \longrightarrow [0,1]$ denote the tent map such that the point $1/2$ is a periodic point of period 3. That is:

$$g(x) = \begin{cases} \mu x & \text{when } x \in [0,1/2], \\ \mu(1-x) & \text{when } x \in [1/2,1], \end{cases}$$

with $\mu = \frac{1+\sqrt{5}}{2}$. This map has periodic points of all periods. Set $p = g(1/2) = \frac{1+\sqrt{5}}{4}$ and let $f : [0,1] \longrightarrow [0,1]$ be the continuous map such that $f(0) = f(1) = 0$, $f(x) = p$ for each $x \in [1-p, p]$ and f is affine on $[0, 1-p]$ and $[p,1]$. Clearly, p is a fixed point of f and 1 is the only period of f. Now consider $T = [0,1]$ as a 2-star with vertices $V = \{0, 1/2, 1\}$ and suppose that we are given the map f with $P = \{0\}$. The map g coincides with f on $V \cup P$ and is monotone (injective) on the closure of each connected component of

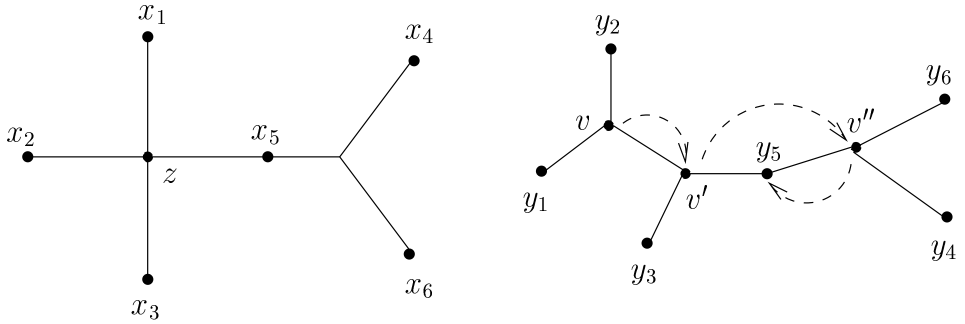

Figure 1.1: *Left figure:* A tree T and a map $f : T \longrightarrow T$ which exhibits an orbit $P = \{x_1, x_2, \ldots, x_6\}$ with $f(x_i) = x_{i+1}$ for $1 \leq i \leq 5$ and $f(x_6) = x_1$. This map can be made P-weakly monotone by setting $f(z) \in P \cup \{z\}$ but it cannot be made P-monotone.
Right figure: A tree S and a map $g : S \longrightarrow S$ having an orbit $Q = \{y_1, y_2, \ldots, y_6\}$ with $g(y_i) = y_{i+1}$ for $1 \leq i \leq 5$ and $g(y_6) = y_1$. If in addition we take $g(v) = v'$, $g(v') = v''$ and $g(v'') = y_5$ then g can be made Q-monotone (and thus Q-weakly monotone).

$T \setminus (V \cup P)$ (so, g is P-weakly monotone). However the map g has periodic points of all periods whereas the map f only has fixed points. $\square$

The above example tells us that it is not straightforward to extend the notion of "minimal model" (or "P-minimal map") to the setting of tree maps. However, in [3] the authors give a definition of *pattern of P* and prove that there always exists a tree S_P and a map $g_P : S_P \longrightarrow S_P$ exhibiting a periodic orbit Q with the same pattern as P and displaying dynamic minimality properties similar to the known ones for the interval case. The crucial point is that the map g_P is *Q-monotone* which means that it is monotone between any two consecutive points of Q (two points a, b of Q are said to be *consecutive* if there are no other points of Q in the convex hull of $\{a, b\}$). We also remark that the tree S_P, which may be different from T, is unique up to homeomorphisms and collapse of invariant forests. The map g_P, which is the crucial tool in our strategy, is called a *P-minimal model*. As an example consider the maps f and g defined in Figure 1.1: It turns out that the orbits P and Q have the same pattern (even living in two different trees) and that the map g is the minimal model corresponding to this pattern. Observe also that the notion of Q-monotonicity is stronger than the notion of Q-weak monotonicity. To see it, consider the map f defined in Figure 1.1 and observe that there does not exist any P-monotone map $\varphi : T \longrightarrow T$ which coincides with f on the set P. Such a map φ would have to satisfy $\varphi([x_1, x_2]) = [x_2, x_3]$ and $\varphi([x_3, x_5]) = [x_4, x_6]$. Thus $\varphi(z) \in [x_2, x_3] \cap [x_4, x_6]$; a contradiction.

Now we are ready to describe the implementation of the strategy we use to study the set of periods of tree maps:

(1) For each periodic orbit P of f calculate Λ_P, the set of periods of the corresponding P-minimal model $g_P : S_P \longrightarrow S_P$ or, if this is not possible, estimate the largest possible subset of Λ_P.

(2) Prove that the set of periods of the P-minimal model g_P is contained in the set of periods of each tree map which exhibits an orbit with the pattern of P. In particular, Λ_P is a subset of the set of periods of f.

(3) Consider each orbit P of f and its associated Λ_P. Then one can obtain the structure of the set of periods of f by describing the structure of the (uncountable) union of all sets Λ_P.

In this chapter we perform step (1) of the above programme by means of the study of the Markov graph of g_P. Indeed, given any tree map $g : S \longrightarrow S$ having a periodic orbit Q and such that g is Q-monotone, we use information from the combinatorics of $g|_Q$ and the topological structure of S in order to study the Markov graph of g and compute as large as possible subsets of the set of periods of g. Moreover, examples are given where the difference between the whole set of periods and these subsets is finite and explicitly bounded.

Since in general T and S_P differ (unless T is an interval or a star) it is not easy to carry out steps (2) and (3) of the above programme. This work is the matter of Chapter 2.

1.2 Basic Definitions and Statement of the Main Results

Let X be a topological space and let $f : X \longrightarrow X$ be a map. As usual, $f^0 = \mathrm{Id}$ and $f^k = f \circ f \circ \cdots \circ f$ (k times) for $k \in \mathbb{N}$. For a finite set A we will denote its cardinality by $|A|$. Given a point $x \in X$ we define its *orbit*, denoted by $\mathrm{Orb}_f(x)$ (or simply by $\mathrm{Orb}(x)$), to be the set $\{f^k(x) : \ k = 0, 1, 2, \ldots\}$. If $|\mathrm{Orb}(x)| = n$, then $f^k(x) \neq x$ for $0 < k < n$ and $f^n(x) = x$. In this case we say that x is a *periodic point of f of period n* (or an *n-periodic point of f*) and that $\mathrm{Orb}_f(x)$ is a *periodic orbit of f of period n* (or an *n-periodic orbit of f*). A point of period 1 is called a *fixed point*, and the set of fixed points of f will be denoted by $\mathrm{Fix}(f)$. The *set of periods of f*, denoted by $\mathrm{Per}(f)$, is the set of periods of all periodic orbits of f. Given a point $x \in X$, we say that x is *eventually periodic* if it is not periodic but $f^n(x)$ is periodic for some $n > 0$. If $A \subset \mathbb{N}$ and $m, n \in \mathbb{N}$, nA stands for $\{nk : k \in A\}$ and $m + nA$ stands for $\{m + nk : k \in A\}$.

A *tree* is a compact uniquely arcwise connected space which is a point or a union of a finite number of intervals (from now on, by an *interval* we mean any space homeomorphic to $[0,1]$). Any continuous map from a tree into itself will be called a *tree map*. If T is a tree and $x \in T$, we define the *valence* of x to be the number of connected components of $T \setminus \{x\}$. Each point of valence 1 will be called an *endpoint* of T and the set of such points will be denoted by $\mathrm{En}(T)$. Each point of valence different from 2 will be called a *vertex* of T and the set of vertices of T will be denoted by $V(T)$. As usual, the closure of each connected component of $T \setminus V(T)$ will be called an *edge* of T. Any tree which is a union of $r > 1$ intervals whose intersection is a unique point x of valence r will be called an *r-star*, and x will be called the *central point*.

If X is a topological space and $f : X \longrightarrow X$ is a map, we will say that a set $A \subset X$ is *f-invariant* if $f(A) \subset A$. A triplet (T, A, f) will be called a *model* if $f : T \longrightarrow T$ is a tree map and A is a finite f-invariant set. In particular, if A is a periodic orbit of f then (T, A, f) will be called a *periodic model*. For a set $B \subset X$ we will denote by $\mathrm{Int}(B)$ and $\mathrm{Cl}(B)$ the interior and the closure of B respectively. Let S be a tree. Given $P \subset S$ we will define the *convex hull* of P, denoted by $\langle P \rangle_S$ or simply by $\langle P \rangle$, as the smallest closed connected subset of S containing P. When $P = \{x, y\}$ we will write $\langle x, y \rangle$ or $[x, y]$ to denote $\langle P \rangle$. The notations (a, b), $(a, b]$ and $[a, b)$ will be understood in the natural way.

Let $g : S \longrightarrow S$ be a tree map. Given $a, b \in S$ we say that $g|_{[a,b]}$ is *monotone* if either $g([a, b])$ is a point or it is an interval and, given two homeomorphisms $\phi : [0,1] \longrightarrow [a, b]$ and $\varphi : g([a, b]) \longrightarrow [0,1]$, then $\varphi \circ g \circ \phi : [0,1] \longrightarrow [0,1]$ is monotone (as a real function). If $P \subset S$ is a finite g-invariant set which contains $\mathrm{En}(S)$, we say that g is *P-monotone* if $g([a, b]) = [g(a), g(b)]$ and $g|_{[a,b]}$ is monotone whenever $[a, b] \cap P = \{a, b\}$. In this case we will say that the model (S, P, g) is *monotone*. If in addition P contains a unique periodic orbit and this orbit consists of a fixed point, then we will say that (S, P, g) is a *trivial* model. Observe that if (S, P, g) is a trivial monotone model and P consists of a fixed point then S reduces to the unique point of P since $\mathrm{En}(S) \subset P$.

Remark 1.2.1. If (S, P, g) is a monotone model, it is shown in Proposition 4.2 of [3] that the image of each vertex z is uniquely determined and is either a vertex or belongs to P. In fact, if $a, b, c \in P$ in such a way that $z \in [a, b] \cap [a, c] \cap [b, c]$ and $\langle \{a, b, c\} \rangle_S \setminus P$ is connected, then it can be easily seen that $g(z)$ is the only point contained in $g([a, b]) \cap g([a, c]) \cap g([b, c])$. $\square$

Let (S, P, g) be a monotone model and let $Q = P \cup V(T)$. Observe that each connected component of $T \setminus Q$ is an interval. By Remark 1.2.1, Q is

g-invariant. It is not difficult to see that g is monotone on each connected component of $T \setminus Q$. In this situation, we can consider the usual notion of the *Markov graph of* g, whose vertices are closures of connected components of $T \setminus Q$ and there is an arrow from K to L if and only if $g(K) \supset L$. It is folk knowledge that there is a certain correspondence between periodic orbits of g and *loops* of its Markov graph (see Section 1.3).

Now we informally sketch the strategy that we use in order to calculate the set of periods of a monotone model. Let (S, P, g) be a non-trivial periodic monotone model. The basic tool we use to obtain periodic points of g is the existence of a special kind of loops in the Markov graph of g, which we call *external loops* (see Section 1.4). The set of external loops in the Markov graph of g which in addition verify certain technical properties will be denoted by $\tilde{\mathcal{E}}(S, P, g)$. If $\tilde{\mathcal{E}}(S, P, g) \neq \emptyset$ then $\operatorname{Per}(g)$ is directly calculable (see Lemma 1.4.6 and Theorem 1.4.7).

If $\tilde{\mathcal{E}}(S, P, g) = \emptyset$ then we proceed as follows. Set $(S_1, P_1, g_1) = (S, P, g)$. We prove that there exist $p_1 \in \mathbb{N}$ and a monotone model (S_2, P_2, g_2) such that $S_2 \subset S_1$, $g_2 = g_1^{p_1}|_{S_2}$ and $\operatorname{Per}(g_1) \supset p_1 \operatorname{Per}(g_2)$. Such a monotone model is called a *partial* p_1*-reduction of* (S_1, P_1, g_1). If we are able to compute $\operatorname{Per}(g_2)$, then the estimation $p_1 \operatorname{Per}(g_2)$ for the set of periods of g_1 is optimal, since we know examples verifying $\operatorname{Per}(g_1) = p_1 \operatorname{Per}(g_2)$. So the problem of estimating $\operatorname{Per}(g_1)$ is reduced to compute $\operatorname{Per}(g_2)$. If $\tilde{\mathcal{E}}(S_2, P_2, g_2) = \emptyset$, we can iterate this procedure. In Section 1.6 it is shown that we can proceed in this way as many times as necessary in order to obtain a finite sequence of monotone models $\{(S_i, P_i, g_i)\}_{i=1}^m$ such that:

(i) $(S_1, P_1, g_1) = (S, P, g)$.

(ii) $(S_{i+1}, P_{i+1}, g_{i+1})$ is a p_i-partial reduction of (S_i, P_i, g_i) for $1 \leq i < m$.

(iii) P_i contains a unique periodic orbit P_i° and $|P_i^\circ| = p_i |P_{i+1}^\circ|$ for $1 \leq i < m$. Moreover, $P_i^\circ \subset P_{i+1} \subsetneq P_i$ when $p_i = 1$.

(iv) $\tilde{\mathcal{E}}(S_i, P_i, g_i) = \emptyset$ for $1 \leq i < m$.

(v) Either (S_m, P_m, g_m) is a trivial model or it verifies $\tilde{\mathcal{E}}(S_m, P_m, g_m) \neq \emptyset$.

Since $\operatorname{Per}(g_i) \supset \{1\} \cup p_i \operatorname{Per}(g_{i+1})$, we easily get that $\operatorname{Per}(g) \supset \{1, p_1, p_1 p_2, \ldots, p_1 p_2 \cdots p_{m-1}\} \cup p_1 p_2 \cdots p_{m-1} \operatorname{Per}(g_m)$. Furthermore, since $P = P_1 = P_1^\circ$, we have that $|P| = p_1 p_2 \cdots p_{m-1} |P_m^\circ|$. We remark that such a sequence of partial reductions of (S, P, g) is not unique.

By means of the above construction, a *complete reduction* of (S, P, g) is defined to be the pair $\{\mathcal{R}, K\}$ where $K = \{1, p_1, p_1 p_2, \ldots, p_1 p_2 \cdots p_{m-1}\}$ and $\mathcal{R} = (S_m, P_m, g_m)$. Note that if $\tilde{\mathcal{E}}(S, P, g) \neq \emptyset$ then $m = 1$ and thus K reduces to $\{1\}$. The model $\mathcal{R}$ will be called a *completely reduced model of* (S, P, g). It satisfies:

(i) $g_m = g^{\max K}|_{S_m}$.

(ii) P_m contains a unique periodic orbit $P_m{}^\circ$ and $|P| = |P_m{}^\circ| \cdot \max K$.

(iii) $\operatorname{Per}(g) \supset K \cup (\max K) \cdot \operatorname{Per}(g_m)$.

Since there exist many sequences of partial reductions, a complete reduction of (S, P, g) is not uniquely determined.

By (iii), the study of the set of periods of a monotone model can be reduced to the study of the set of periods of its completely reduced models. This is the strategy we use and it gives rise to our main result. In order to state it, we need to introduce some more notation.

Let $\mathcal{R} = (\overline{S}, \overline{P}, \overline{g})$ be a non-trivial completely reduced model of a given monotone model (S, P, g). We will prove that $\operatorname{Per}(\overline{g})$ depends on three non-negative constants (besides $|\overline{P}^\circ|$, of course). These constants can be directly calculated from the combinatorics induced by $\overline{g}$ on the $\overline{g}$-invariant set $\overline{P} \cup V(\overline{S})$. Since these numbers strongly depend on the topological structure of the tree $\overline{S}$ and the behavior of $\overline{g}$ on $\overline{P}$, we denote them by $n(\mathcal{R})$, $p(\mathcal{R})$ and $q(\mathcal{R})$ in order to stress their dependence from the model. The constant $n(\mathcal{R})$ is the minimum integer n such that $\overline{g}^n(\overline{P}) = \overline{P}^\circ$. On the other hand, $p(\mathcal{R})$ is called a *type* of the model, and essentially is a generalization of the notion of *type of a periodic orbit* introduced in [16] for star maps. Finally $q(\mathcal{R})$ will be called the *rotation index* of the model. The precise definition of these constants is given in Section 1.4.

Next we introduce a notation to deal with a special type of initial segments of the ${}_p\!\geq$ orderings. If $p \in \mathbb{N}$ and $r \in \mathbb{N} \cup \{p2^\infty\}$, we define $\mathcal{S}_p(r) = \{k \in \mathbb{N} : r \,{}_p\!\geq k\}$. Note that if $r \in p\mathbb{N}$ then $\mathcal{S}_p(r) = \{1\} \cup p\{k \in \mathbb{N} : r/p \trianglerighteq k\}$ and if $r \notin p\mathbb{N}$ then $\mathcal{S}_p(r) = \{1, r\} \cup \{ri + pj : i \geq 0,\, j \geq 1\}$. Given $p, r \in \mathbb{N}$, we define

$$\mathcal{S}_p^*(r) = \begin{cases} \mathcal{S}_p(r) & \text{if } r \notin p\mathbb{N} \\ \mathcal{S}_p(3p) & \text{if } r \in p\mathbb{N} \end{cases}$$

Observe that if $r \in p\mathbb{N}$ then $\mathcal{S}_p^*(r) = p\mathbb{N} \cup \{1\} \supset \mathcal{S}_p(r)$.

Remark 1.2.2. Let k, p, r be natural numbers. Then we have that $\{1\} \cup k\mathcal{S}_p^*(r) = \{k\} \cup \mathcal{S}_{kp}^*(kr)$. Indeed, if $r \notin p\mathbb{N}$ then $\{1\} \cup k\mathcal{S}_p^*(r) = \{1\} \cup k(\{1, r\} \cup \{ri + pj : i \geq 0, j \geq 1\}) = \{1, k, kr\} \cup \{kri + kpj : i \geq 0, j \geq 1\} = \{k\} \cup \mathcal{S}_{kp}^*(kr)$. On the other hand, when $r \in p\mathbb{N}$ we get $\{1\} \cup k\mathcal{S}_p^*(r) = \{1\} \cup k(p\mathbb{N} \cup \{1\}) = \{1, k\} \cup kp\mathbb{N} = \{k\} \cup \mathcal{S}_{kp}^*(kr)$. $\square$

From now on, we take $\{1, 2, \ldots, n\}$ as the representatives of the classes of $\mathbb{Z}/n\mathbb{Z}$.

Now we are ready to state the main results of this chapter.

Theorem A. *Let (S, P, g) be a periodic monotone model. If P consists of a fixed point of g then $\operatorname{Per}(g) = \{1\}$. Otherwise, there exist complete reductions* *of* (S, P, g)*. For any complete reduction* $\{\mathcal{R}, K\}$ *of* (S, P, g)*, we have that* $\mathrm{Per}(g) \supset K$*. If, in addition,* $\mathcal{R}$ *is non-trivial and we denote* $p(\mathcal{R})$*,* $q(\mathcal{R})$*,* $n(\mathcal{R})$ *and* $\max K$ *by* p*,* q*,* n *and* k *respectively, then*

$$\mathrm{Per}(g) \supset K \cup \mathcal{S}^*_{kp}(|P| + lkp) \setminus \{2kp, 3kp, \ldots, \lambda kp\}$$

for some $0 \le \lambda p \le \frac{|P|}{k} + p + q + n + 1$ *and some* $0 \le l \le \frac{|P|}{k} + q + 1$*. Furthermore, if* $n = 0$ *then* $lp \le p + q - (q \bmod p)$*.*

The periods computed in the proof of Theorem A correspond to periodic orbits which do not intersect the set $V(S)$ of vertices of S. We additionally prove (see Corollary 1.6.8) that $\mathrm{Per}(g)$ contains a finite set $\mathcal{V}$ whose elements divide the least common multiple of the periods of all periodic orbits contained in $V(S)$.

Remark 1.2.3. When $|P| \in kp\mathbb{N}$ the upper bound for l in Theorem A is irrelevant, since $\mathcal{S}^*_{kp}(|P| + lkp) = kp\mathbb{N}$ for any l. On the other hand, when $|P| \notin kp\mathbb{N}$ the upper bound for l controls how far $\mathcal{S}^*_{kp}(|P| + lkp)$ is from $\mathcal{S}^*_{kp}(|P|)$. Indeed, one can prove that $\mathcal{S}^*_{kp}(|P|) \setminus \mathcal{S}^*_{kp}(|P| + lkp) = \{|P|\} \cup (\{i|P| + jkp : 1 \le i < p/\mathrm{g.c.d.}(p, n),\ 1 \le j \le il\} \setminus \{|P| + lkp\})$. $\square$

Sometimes a continuous self–map of a compact space is called *chaotic* if it has positive topological entropy (see [27] for a definition). Then it can be derived from Theorem E of [36] and Theorem A that if $\mathcal{R}$ is a non-trivial model then g is chaotic. And conversely, it is not difficult to see that if g is not chaotic then $\mathrm{Per}(g)$ must be finite (this is true only for monotone models). Thus the monotone models with a trivial (respectively non-trivial) completely reduced model correspond to zero entropy (resp. chaotic) maps.

We must stress the fact that there are some known results which describe the set of periods of some kinds of tree maps except for a finite set of periods (see for instance [22] and [14]). Nevertheless, nothing is said usually about this finite set. Theorem A states that the set of periods of a (chaotic) monotone model contains a set $\mathcal{C}$ which is $\mathcal{S}^*_{kp}(|P|)$ except for an *explicitly bounded* finite set of periods. In fact, from Remark 1.2.3 it follows that $\mathcal{S}^*_{kp}(|P|) \setminus \mathcal{C}$ is exactly $\{2kp, 3kp, \ldots, \lambda kp\}$ if $|P| \in kp\mathbb{N}$ and $\{2kp, 3kp, \ldots, \lambda kp\} \cup \{|P|\} \cup (\{i|P| + jkp : 1 \le i < p/\mathrm{g.c.d.}(p, n),\ 1 \le j \le il\} \setminus \{|P| + lkp\})$ otherwise. Thus the difference between $\mathcal{C}$ and $\mathcal{S}^*_{kp}(|P|)$ depends on the constants λ and l, which depend on combinatorial data extracted from the model by means of the constants q and n. The smaller q and n are, the bigger (and closer to $\mathcal{S}^*_{kp}(|P|)$) $\mathcal{C}$ is.

A natural question arises: how accurate is the estimation of $\mathrm{Per}(g)$ given by Theorem A in relation to Sharkovskii and Baldwin theorems when S is

Table 1.1: Some examples of sets of periods given by Theorem A and Theorems of Sharkovskii and Baldwin.

Model	Complete reduction	Sharkovskii's or Baldwin's Theorem	Theorem A
S interval, $\lvert P\rvert = t\cdot 2^s$, t odd, $s>1$, no division	$\mathcal{R}=(S,P,g)$, $K=\{1\}$, $q=n=l=0$, $p\in\{1,2\}$	$\mathrm{Sh}(t\cdot 2^s)$	$p\mathbb{N}\setminus\{2p,3p,\dots,\lambda p\}$
S r-star, $\lvert P\rvert = r\cdot 2^s$, P primary	$\mathcal{R}$ trivial, $K=\{1,r,2r,2^2r,\dots,2^sr\}$	$\mathcal{S}_r(r\cdot 2^s)$	$\mathcal{S}_r(r\cdot 2^s)$
S r-star, $\lvert P\rvert = rt\cdot 2^s$, $t>1$ odd, P primary	$\mathcal{R}$ non-trivial, $K=\{1,r,2r,2^2r,\dots,2^sr\}$, $q=n=l=0$, $p=2$, $\lambda=\frac{t-1}{2}$	$\mathcal{S}_r(rt\cdot 2^s)$	$\mathcal{S}_r(rt\cdot 2^s)\setminus r2^{s+1}\cdot\{2,3,\dots,\frac{t-1}{2}\}$
S r-star, $\lvert P\rvert = s\notin r\mathbb{N}$, (s,r)-spiral map	$\mathcal{R}$ non-trivial, $K=\{1\}$, $q=n=l=0$, $p=r$, $\lambda=\frac{s-(s \bmod r)}{r}$	$\mathcal{S}_r(s)$	$\mathcal{S}_r(s)\setminus\{2r,3r,\dots,s-(s \bmod r)\}$

an interval or a star? Given $r\in\mathbb{N}$, let us write $\mathrm{Sh}(r)$ for the initial segment of Sharkovskii's ordering starting at r. That is, $\mathrm{Sh}(r)=\{s\in\mathbb{N} : r\trianglerighteq s\}$.

Suppose that S is an interval and $|P| = t\cdot 2^s$ with t odd and $s>1$. Assume in addition that P has *no division* (see for instance [35]). Then from the proof of Theorem A one gets that (S,P,g) admits a complete reduction $\{\mathcal{R},K\}$ with $\mathcal{R}=(S,P,g)$, $K=\{1\}$, $q=n=l=0$ and $p\in\{1,2\}$. If $p=1$ then Theorem A states that $\mathrm{Per}(g)\supset\mathbb{N}\setminus\{2,3,\dots,\lambda\}$. When $p=2$, we get

$\mathrm{Per}(g) \supset 2\mathbb{N} \setminus \{4, 6, \ldots, 2\lambda\}$. In both cases, these sets contain infinitely many periods which are not in $\mathrm{Sh}(t \cdot 2^s)$. Theorem A can provide more information than Sharkovskii's theorem, since in our result other combinatorial features of the orbit P, besides its period, are taken into account. This goes in the direction of the main result of [35], Baldwin's theorem and other several results in the same spirit (see [38] or [14]).

Assume that S is an interval and P is a *primary orbit* (see [15] or [7]) of period $t \cdot 2^s$ with t odd and $s \geq 0$. Then $\mathrm{Per}(g) = \mathrm{Sh}(t \cdot 2^s)$, and it is not difficult to see that (S, P, g) admits a complete reduction $\{\mathcal{R}, K\}$ such that $K = \{1, 2, 2^2, \ldots, 2^s\}$ and $\mathcal{R}$ is a trivial model if and only if $t = 1$. If $\mathcal{R}$ is trivial then Theorem A states that $\mathrm{Per}(g) \supset K = \{1, 2, 2^2, \ldots, 2^s\} = \mathrm{Sh}(2^s)$. On the other hand, when $\mathcal{R}$ is not trivial from the proof of Theorem A one gets that $q = n = l = 0$, $k = 2^s$, $p = 2$ and $\lambda = \frac{t-1}{2}$. Hence Theorem A states that

$$\begin{aligned}\mathrm{Per}(g) \supset\ & K \cup \mathcal{S}^*_{2^{s+1}}(t \cdot 2^s) \setminus \\ & \{2 \cdot 2^{s+1}, 3 \cdot 2^{s+1}, \ldots, \frac{t-1}{2} \cdot 2^{s+1}\} \\ =\ & \{1, 2, 2^2, \ldots, 2^s\} \cup \{t \cdot 2^s\} \cup \\ & 2^s\{ti + 2j, i \geq 0, j \geq 1\} \setminus \\ & \{2 \cdot 2^{s+1}, 3 \cdot 2^{s+1}, \ldots, \frac{t-1}{2} \cdot 2^{s+1}\}.\end{aligned}$$

It is not difficult to show that this set is exactly

$$\mathrm{Sh}(t \cdot 2^s) \setminus \{2 \cdot 2^{s+1}, 3 \cdot 2^{s+1}, \ldots, \frac{t-1}{2} \cdot 2^{s+1}\}.$$

A similar calculus can be done when S is an r-star (with $r \geq 3$) and P is a primary orbit. Some of these computations are shown in Table 1.1. When $|P| \notin r\mathbb{N}$ then g is the $(|P|, r)$*-spiral map* (see [16]).

Thus, when S is an interval or a star, in some cases Theorem A misses out the subset of periods $\{2kp, 3kp, \ldots, \lambda kp\}$. Nevertheless, in these cases it can be shown that g^{kp} exhibits a *horseshoe*. Then it easily follows that $\mathrm{Per}(g) \supset kp\mathbb{N}$. In particular, $\mathrm{Per}(g) \supset \{2kp, 3kp, \ldots, \lambda kp\}$. The existence of this horseshoe is due to the (geometric) fact that there are no vertices of S between consecutive points of P. For a general tree map it is not true that g^{kp} has a horseshoe, and thus $\mathrm{Per}(g)$ does not necessarily contain $kp\mathbb{N}$.

Also the following natural question arises: do there exist monotone models whose set of periods contains exactly the periods of Theorem A and no other? Before answering this question, we must give the range of possible values of the constants p, q, n and k in Theorem A. We have that $p \geq 1$,

$q \geq 0$ and $n \in \{0,1,2\}$. Set $r = |P|/k$. In Corollary 1.8.2 we show that the values of p and q are bounded in terms of r. In particular, when $n = 0$ we have that

$$\begin{gathered} p \leq r-1, \\ q+4 \leq r \text{ when } p=1 \text{ and } \\ 2p+q+1 \leq r \text{ when } q>0. \end{gathered} \tag{1.1}$$

The answer to the above question is given by the following converse of Theorem A:

Theorem B. *Let $K \subset \mathbb{N}$ be a set of the form $\{1, k_1, k_2, \ldots, k_m\}$ such that $k_1 > 1$ and k_i strictly divides k_{i+1} for $1 \leq i < m$. Set $k = k_m$. Then:*

(a) There exists a periodic monotone model (R, B, h) with $|B| = k$ and $\operatorname{Per}(h) = K$.

(b) Given any $r > 1$, $p \geq 1$ and $q \geq 0$ verifying (1.1), there exists a periodic monotone model (S, P, g) and a complete reduction $\{(\overline{S}, \overline{P}, \overline{g}), K\}$ of (S, P, g) such that $|\overline{P}^{\circ}| = r$, $p(\overline{S}, \overline{P}, \overline{g}) = p$, $q(\overline{S}, \overline{P}, \overline{g}) = q$, $n(\overline{S}, \overline{P}, \overline{g}) = 0$ and $\operatorname{Per}(g) = K \cup \mathcal{C}$, where $\mathcal{C}$ is a set such that

$$\mathcal{S}^*_{kp}(|P| + lkp) \setminus \{2kp, 3kp, \ldots, \lambda kp\} \subset \mathcal{C} \subset \mathcal{S}^*_{kp}(|P|)$$

with $lp = p + q - (q \bmod p)$ and λp being the largest multiple of p smaller than $r + p + q + 1$.

In order to simplify the proof of Theorem B, we have considered only models for which $n = 0$. In fact, according to Theorem A, if one looks for a characterization of $\operatorname{Per}(g)$ up to a finite set then the values of q and n are irrelevant.

This chapter is organized as follows. In Section 1.3 we introduce the usual f-covering tools which relate the periodic orbits of a map and the loops of its associated Markov graph. In Section 1.4 we define a particular class of monotone models, which we call *y-expansive*, and we compute periodic orbits associated to the loops of the Markov graph of y-expansive models. In Section 1.5 we use the notion of a *canonical model* introduced in [3]. From each monotone model (S, P, g) we construct a canonical model (S', P', g') and find a relation between $\operatorname{Per}(g)$ and $\operatorname{Per}(g')$. Moreover, we prove that every canonical model is, in particular, y-expansive. This allows us to use the results of Section 1.4 for canonical models. In Section 1.7 we prove Theorem A for a monotone model (S, P, g). The complexity of the arguments of the proof depends strongly on the combinatorics of the g-invariant set $P \cup V(S)$ around a fixed point y of g. This combinatorics is studied in Section 1.6, where we define the notion of a *twist model around a fixed point* and we remark that if (S, P, g) is not a twist model around y then the theorems of

Section 1.4 can be directly used. The sets of periods of the twist models are studied in Section 1.6. In Section 1.8 we prove the inequalities (1.1). Finally Section 1.9 is devoted to prove Theorem B.

1.3 Markov Graphs and Periodic Orbits

Let T be a tree and let $Q \subset T$ be a finite set containing $V(T)$. An interval of T will be called *Q-basic* if it is the closure of a connected component of $T \setminus Q$. Given $f : T \longrightarrow T$ and $K, L \subset T$, we will say that K *f-covers* L if $f(K) \supset L$. We will use the notation $K \to L$ (or $K \xrightarrow{f} L$ if we want to specify the map) to denote that K f-covers L. In this setting, it makes sense to consider the *(Markov) f-graph of* Q, whose vertices are Q-basic intervals and, if I, J are Q-basic intervals, there is an arrow $I \to J$ if and only if I f-covers J.

A monotone model (T, Q, f) will be called a *Markov* model if $V(T) \subset Q$.

The results of this section are well known for interval and star maps and extend straightforwardly to the case of tree maps. However, we include some proofs for completeness.

Lemma 1.3.1. *Let (T, Q, f) be a Markov model. Let $K \subset T$ be a connected union of Q-basic intervals. Then for each Q-basic interval $J \subset f(K)$ there exists a Q-basic interval $I \subset K$ such that I f-covers J.*

Proof. Note that $\text{Int}(J) \cap V(T) = \emptyset$ because J is a Q-basic interval and $V(T) \subset Q$. Since f is continuous and T is a tree, it follows that there exists an interval $I' \subset K$ such that $f(I') = J$. Furthermore, since f is Q-monotone we can assume $\text{Int}(I') \cap V(T) = \emptyset$. Thus the lemma follows by taking a Q-basic interval I such that $I' \subset I \subset K$. ■

Let (T, Q, f) be a Markov model. There is a certain correspondence between periodic points of f and *loops* in the f-graph of Q. We will use the usual notions (see Chapter 1 of [8] or [21]): the *concatenation* of two loops α and β will be denoted by $\alpha\beta$, and $\alpha^n = \alpha\alpha\ldots\alpha$ (n times) will be called an *n-repetition of* α. A loop will be called *elementary* if it cannot be formed by concatenating two loops. A loop α is *simple* if it is not an n-repetition of any other loop with $n \geq 2$. The length of a loop α will be denoted by $|\alpha|$. If $J_0 \to J_1 \to \ldots \to J_{n-1} \to J_0$ is a loop α in the f-graph of Q and $x \in \text{Fix}(f^n)$ we say that x and α are *associated* if $f^i(x) \in J_i$ for $0 \leq i < n$. In this case we also will say that $\text{Orb}(x)$ and α are *associated*. We note that when x and α are associated the period of x can be a strict divisor of $|\alpha|$. As usual, to every arrow $I \to J$ in the f-graph of Q we associate a *sign* which is $+1$ if $f|_I$

is non-decreasing and -1 if it is non-increasing. Then we say that the loop $J_0 \to J_1 \to \ldots \to J_{n-1} \to J_0$ is *positive* if the product of the signs of the arrows $J_0 \to J_1, J_1 \to J_2, \ldots, J_{n-1} \to J_0$ is +1 and *negative* if it is -1.

Lemma 1.3.2. *Let (T, Q, f) be a Markov model. If P is a periodic orbit of f such that $P \cap Q = \emptyset$, then there exists a unique loop α of length $|P|$ in the f-graph of Q such that P and α are associated.*

Proof. Let $x \in P$. For each $0 \leq i < |P|$, there exists a unique Q-basic interval J_i such that $f^i(x) \in \operatorname{Int}(J_i)$. Since f is Q-monotone and $V(T) \subset Q$, it follows that J_i f-covers J_{i+1} for $0 \leq i < |P| - 1$ and $J_{|P|-1}$ f-covers J_0. ■

The next result follows easily from the ideas of Lemma 1.4 of [21]. See also Lemma 4.2.1.

Lemma 1.3.3. *Let (T, Q, f) be a Markov model. Let α be a loop $J_0 \to J_1 \to \ldots \to J_{n-1} \to J_0$ in the f-graph of Q. Then there exist closed intervals $K_i \subset J_i$ for $0 \leq i < n$ such that $f(K_i) = K_{i+1}$ for $0 \leq i < n - 1$ and $f(K_{n-1}) = J_0$. Moreover, there exists $x \in \operatorname{Fix}(f^n)$ such that $f^i(x) \in K_i$ for $0 \leq i < n$. In particular, x and α are associated.*

Remark 1.3.4. With the notation of Lemma 1.3.3, it is not difficult to see that f^n is monotone on K_0, and the loop is positive (respectively negative) if and only if $f^n|_{K_0}$ is non-decreasing (respectively non-increasing). □

Under the hypotheses of Lemma 1.3.3, there exists a periodic point x associated to α. Therefore, the loops of the f-graph of Q are useful to obtain periodic orbits of the map f. When doing this, the basic problem is to determine the exact period of the periodic point that one gets. The following result imposes some conditions on α in order to assure that the period of x coincides with the length of α.

Lemma 1.3.5. *Let (T, Q, f) be a Markov model. Let α be a simple loop $[a, b] \to J_1 \to J_2 \to \ldots \to J_{n-1} \to [a, b]$ in the f-graph of Q. Let x be the periodic point given by Lemma 1.3.3. If $x \in (a, b)$ then the period of x is n. This happens, in particular, when any of the following statements holds:*

(a) a and b are not fixed points of f^n.

(b) α is negative.

Proof. We use the notation of Lemma 1.3.3. A standard argument (see for instance the first part of the proof of Lemma 1.2.11 of [8]) assures that, since $x \in \operatorname{Int}(J_0)$ and α is simple, the period of x coincides with $|\alpha| = n$. Now we will see that $x \in \operatorname{Int}(J_0)$ when either (a) or (b) is satisfied.

When (a) holds, obviously $x \notin \{a, b\}$ since x is a fixed point of f^n.

Assume that (b) holds. Set $K_0 = [y, z] \subset J_0$. Then $x \in [y, z]$ and $f^n([y, z]) = J_0$. Since α is negative, by Remark 1.3.4 f^n is monotone and non-increasing on $[y, z]$. Since $x \in \mathrm{Fix}(f^n)$, it follows that $x \neq y$ and $x \neq z$. Thus $x \in (y, z) \subset \mathrm{Int}(J_0)$. ∎

In view of Lemma 1.3.5, simple loops are specially useful to calculate periodic orbits. The following lemma gives a tool to obtain a simple loop from a given one.

Lemma 1.3.6. *For each loop γ which is not a repetition of an elementary loop there exists a simple loop which can be obtained by permuting the elements of γ.*

Proof. Clearly γ can be written as $\alpha^l \beta$ for some $l \in \mathbb{N}$, a non-empty elementary loop α and $|\beta| > 0$. If $\beta = \beta' \alpha \beta''$ (where either β' or β'' can be empty) then the loop $\gamma' = \alpha^{l+1} \beta' \beta''$ is obtained by permuting the elements of γ. By iterating this procedure, if necessary, we obtain a loop $\tilde{\gamma} = \alpha^r \tilde{\beta}$ which is a permutation of the elements of γ such that $r \in \mathbb{N}$, $|\tilde{\beta}| > 0$ and $\tilde{\beta}$ does not contain α. Clearly $\tilde{\gamma}$ is simple. ∎

1.4 Periodic Orbits in y-expansive Monotone Models

In this section we introduce a particular class of monotone models, which will be called *y-expansive*. This kind of models satisfy certain properties of expansivity around a fixed point y. We study the Markov graph of y-expansive models and derive the structure of the set of periods.

Let T be a tree. Given a point $y \in T$, a (partial) ordering among the points of T may be defined: for $z, z' \in T$, we write $z \prec_y z'$ if and only if $z \in [y, z')$. We remark that if $(z, z') \cap (V(T) \cup \{y\}) = \emptyset$ then either $z \prec_y z'$ or $z' \prec_y z$. The notations ${}_y\!\succ$, $\preceq_y$ and ${}_y\!\succeq$ will be understood in the natural way, and for simplicity we will omit the subindex y when no confusion seems possible. If I, J are subsets of T, we will write $I \prec_y J$ if $z \prec_y z'$ for each $z \in \mathrm{Int}(I)$ and $z' \in \mathrm{Int}(J)$.

Given a finite set $Q \subset T$ and a point $y \in T$, we shall denote by $Z^\star(Q)$ the connected component of $(T \setminus Q) \cup \{y\}$ which contains y. Let n be the number of connected components of $T \setminus Z^\star(Q)$. These connected components will be denoted by $Z(Q)_i$ for $1 \leq i \leq n$ and we will call them *y-branches*. The set $\mathrm{Cl}(Z^\star(Q)) \cap Z(Q)_i$ consists of a single point which belongs to Q. This point will be denoted by $x(Q)_i$. We remark that, for each $z \in T$, $z \in Z(Q)_i$ if and only if $x(Q)_i \preceq_y z$. Finally we set $X(Q) = \{x(Q)_i\}_{i=1}^n$.

Let $f : T \longrightarrow T$ be a tree map, $y \in \text{Fix}(f)$ and let $Q \subset T$ be a finite f-invariant set. We will say that Q is *y-typifiable* if $f(X(Q)) \cap Z^\star(Q) = \emptyset$.

Remark 1.4.1. If $y \notin Q$ then Q is y-typifiable. If $y \in Q$ then Q is y-typifiable if and only if $f(x(Q)_i) \neq y$ for $1 \leq i \leq |X(Q)|$. Moreover, it follows that Q is y-typifiable if and only if $Q \cup \{y\}$ is y-typifiable. □

If Q is y-typifiable then we consider the map $\Phi_Q : X(Q) \longrightarrow X(Q)$ defined by $\Phi_Q(x(Q)_i) = x(Q)_j$ if and only if $f(x(Q)_i) \in Z(Q)_j$. Observe that Φ_Q is well defined and, since it acts on a finite set, it has periodic orbits. The period p of a periodic orbit of Φ_Q will be called a *type of Q* (note that the type of a y-typifiable set is not necessarily unique).

Given a type p of Q, in what follows we will assume that the y-branches are indexed in such a way that $f(x(Q)_i) \in Z(Q)_{i+1 \bmod p}$ for $1 \leq i \leq p$.

Observe that all the definitions introduced up to now in this section depend on the chosen point y. For simplicity, this dependence is not made explicit in the notation.

Lemma 1.4.2. *Let $f : T \longrightarrow T$ be a tree map. Let $y \in \text{Fix}(f)$ and let $Q \subset T$ be a y-typifiable set. If p is a type of Q then $p \in \text{Per}(f)$.*

Proof. Let $r : T \longrightarrow \text{Cl}(Z^\star)$ be the natural retraction. Then $r(f(x(Q)_i)) = x(Q)_{i+1 \bmod p}$ for $i = 1, 2, \ldots, p$. Then $x(Q)_1$ is a p-periodic point of $r \circ f$ and thus $p \in \text{Per}(r \circ f)$. The lemma follows because $\text{Per}(r \circ f) \subset \text{Per}(f)$ (see, for instance, Corollary 4.2 of [16]). ■

Let (T, A, f) be a monotone model. It is not difficult to prove that if $B \subset T$ is finite and f-invariant then f is $(A \cup B)$-monotone. Thus, since $A \cup V(T)$ is an f-invariant set by Remark 1.2.1, f is also $(A \cup V(T))$-monotone. We will say that a monotone model (T, A, f) is *y-expansive* for $y \in \text{Fix}(f) \setminus A$ if $\text{Orb}_f(v)$ is not contained in $Z^\star(A)$ for every $v \in V(T) \setminus \{y\}$. This sort of models will play an important role in this chapter. Lemma 1.4.3 states that on y-expansive models it is possible to define the type of some "natural" invariant sets.

Lemma 1.4.3. *Let (T, A, f) be a y-expansive model. Let $P \subset T$ be a finite (or empty) f-invariant set such that $y \notin P$. Then the sets A, $A \cup \{y\}$, P, $P \cup \{y\}$, $A \cup V(T) \cup P$ and $A \cup V(T) \cup P \cup \{y\}$ are y-typifiable.*

Proof. Let $Q = A \cup V(T) \cup P$. By Remark 1.4.1, to prove the lemma it suffices to show that the sets A, P and Q are y-typifiable. Since $y \notin A$ and $y \notin P$, it

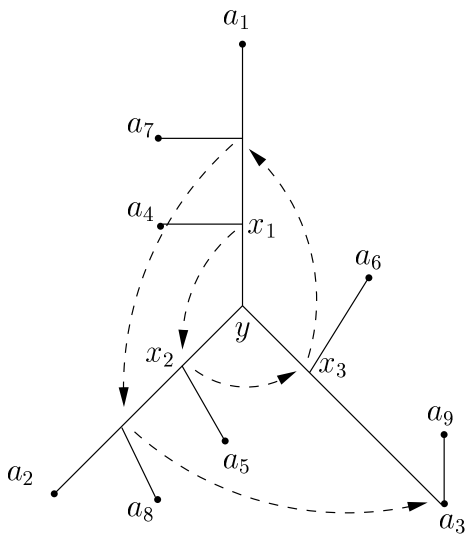

Figure 1.2: A y-expansive model (T, A, f) with $A = \{a_i\}_{i=1}^{9}$ and $f(a_i) = a_{i+1 \bmod 9}$. For this model, $\mathrm{Cl}(Z^\star) = \langle\{x_1, x_2, x_3\}\rangle$, A^y has a unique type $p = 3$ and $q_1 = 5$, $q_2 = 4$, $q_3 = 3$. Therefore, the rotation index associated to the type is 3.

follows that A and P are y-typifiable. If in addition $y \notin V(T)$ then $y \notin Q$ and we are done. Assume that $y \in V(T)$. Then $y \in Q$ and we must prove that if $i \in \{1, 2, \ldots, |X(Q)|\}$ then $f(x(Q)_i) \neq y$. This is obvious if $x(Q)_i \in A \cup P$. Assume that $x(Q)_i \in V(T) \setminus (A \cup P)$. We note that $(y, x(Q)_i) \cap Q = \emptyset$ and, in particular, $(y, x(Q)_i) \cap A = \emptyset$. Then, as an immediate consequence of the fact that $\mathrm{En}(T) \subset A$, we have that $x(Q)_i \prec x(A)_j$ for some $j \in \{1, 2, \ldots, |X(A)|\}$. This is equivalent to $x(Q)_i \in Z^\star(A)$, and then $f(x(Q)_i) \neq y$ since (T, A, f) is y-expansive. ■

Let (T, A, f) be a y-expansive model. The set $A \cup V(T) \cup \{y\}$ will be denoted by A^y, and $|X(A^y)|$ will be denoted by $n^\star$. Furthermore, from now on we will write $Z^\star$, Z_i and x_i instead of $Z^\star(A^y)$, $Z(A^y)_i$ and $x(A^y)_i$, for $1 \leq i \leq n^\star$.

By Lemma 1.4.3, A^y is y-typifiable. Let p be a type of A^y. For each $i \in \{1, 2, \ldots, p\}$ there exists a non-negative number, which we will denote by q_i, such that $[y, f^j(x_i)] \cap A = \emptyset$ for $0 \leq j < q_i$ and $[y, f^{q_i}(x_i)] \cap A \neq \emptyset$ (recall that $Z^\star = Z^\star(A^y) \subset Z^\star(A)$). Note that $x_i \in A$ if and only if $q_i = 0$. The non-negative integer $\min\{q_1, q_2, \ldots, q_p\}$ will be called a *rotation index of* (T, A, f) *associated to the type* p. Observe that the rotation index associated to a type p of A^y is not unique, since it depends on the chosen p-periodic orbit of Φ_{A^y}.

The following technical lemma concerns the dynamical behavior of a y-expansive model near the fixed point y. See Figure 1.2 for an example.

Lemma 1.4.4. *Let* (T, A, f) *be a* y*-expansive model and let* p *be a type of* A^y. *Let* $k \in \{1, 2, \ldots, p\}$ *be such that* $q_k > 0$. *Then* $x_{k+i \bmod p} \preceq_y f^i(x_k)$ *for* $1 \leq i \leq q_k$ *and* $f^{i-p}(x_k) \prec_y f^i(x_k)$ *for* $p < i \leq q_k$.

Proof. In the whole proof, the subindexes will be considered modulo p.

We will prove the first statement by induction on i. From the definition of type, it follows that $x_{k+1} \preceq_y f(x_k)$. Hence, the first statement holds for $i = 1$. Now take $1 < i \leq q_k$ and assume that $y \prec_y x_{k+i-1} \preceq_y f^{i-1}(x_k)$. Since $i-1 < q_k$, the definition of q_k implies that $[y, f^{i-1}(x_k)] \cap A = \emptyset$. Thus, from the A-monotonicity of f it follows that $f(x_{k+i-1}) \preceq_y f(f^{i-1}(x_k)) = f^i(x_k)$. Since from the definition of type $x_{k+i} \preceq_y f(x_{k+i-1})$, the first statement is proved.

Let us prove the second statement also by induction on i. Since $q_k > 0$, we have that $x_k \in V(T)$. We assume that $p < q_k$ since otherwise there is nothing to prove. For $i = p+1$ we must show that $f(x_k) \prec_y f^{p+1}(x_k)$. Since $p < q_k$, we know from the first statement that $y \prec_y x_k \preceq_y f^p(x_k)$. The fact that f is A-motonone implies, as above, that $f(x_k) \preceq_y f^{p+1}(x_k)$. If $f(x_k) = f^{p+1}(x_k)$, since $p < q_k$ it follows that $\mathrm{Orb}(x_k)$ is a finite f-invariant set contained in $(Z^\star(A) \cap V(T)) \setminus \{y\}$. This contradicts the fact that (T, A, f) is y-expansive and proves that $f(x_k) \prec_y f^{p+1}(x_k)$.

Now take $p+1 < i \leq q_k$ and assume that $f^{i-1-p}(x_k) \prec_y f^{i-1}(x_k)$. Then we obtain that $f^{i-p}(x_k) \prec_y f^i(x_k)$ in the same way as above. ■

Let (T, A, f) be a y-expansive model. Let p be a type of A^y. For $i \in \{1, 2, \ldots, p\}$, we write I_i for $[y, x_i]$. We note that these sets are A^y-basic intervals, and they are contained in $\mathrm{Cl}(Z^\star)$. Moreover, by the definition of p, the f-graph of A^y contains the loops $I_{i \bmod p} \to I_{i+1 \bmod p} \to \ldots \to I_{i+p \bmod p}$, which will be called *typical loops*. The intervals $I_1, I_2, \ldots, I_p$ will be called *typical intervals*.

Remark 1.4.5. Assume that a typical interval I_i f-covers an interval J which is not typical. Since $f(y) = y$ and $f|_{I_i}$ is monotone, it follows that $I_{i+1 \bmod p} \prec_y J$. □

The periods of f obtained in this section (see Lemma 1.4.6 and Theorem 1.4.7) will be computed by linking the typical loops with some special loops of the Markov f-graph of A^y. A loop in the f-graph of A^y will be called *external* if it starts and ends at a typical interval and it contains an element which is not a typical interval. We denote by $\mathcal{E}(T, A, f)$ the set of external loops in the f-graph of A^y. Observe that the notions of *typical interval* and *external loop* depend on the point y, the type p and the chosen p-periodic orbit of Φ_{A^y}. For simplicity, the notations do not take it into account.

Next we state and prove two results that allow us to obtain periodic orbits in the context of y-expansive models.

Lemma 1.4.6. *Let (T, A, f) be a y-expansive model and let p be a type of A^y. If $\beta \in \mathcal{E}(T, A, f)$ then $\{|\beta|i + pj : i, j \geq 1\} \subset \mathrm{Per}(f)$.*

Proof. Since β is external, β starts and ends at a typical interval I_t. Let α be the typical loop starting and ending at I_t. Set $k = |\beta|i + pj$ with $i, j \geq 1$. We consider the loop $\alpha^j\beta^i$, whose length is k. Since β is external, $\alpha^j\beta^i$ is not a repetition of α. So, Lemma 1.3.6 gives us a simple loop γ obtained by permuting the elements of $\alpha^j\beta^i$. By Lemma 1.3.3, there is a point $x \in T$ associated to γ such that $f^k(x) = x$. Since $j \geq 1$, we can assume that $x \in I_t$ and $f^n(x) \in I_{t+n \bmod p}$ for $1 \leq n \leq p$.

By Lemma 1.3.5, it is enough to prove that $x \in \mathrm{Int}(I_t)$. First we show that $x \neq y$. Since β is external, γ contains an arrow $I_r \to J$ for some $r \in \{1, 2, \ldots, p\}$ and some J which is not a typical interval. Then by Remark 1.4.5 $I_{r+1 \bmod p} \prec_y J$ and thus $y \notin J$. Since some iterate of x belongs to J, it follows that $x \neq y$. To end the proof of the claim we must show that $x \neq x_t$. Suppose that $x = x_t$. Then clearly $f^n(x) = x_{t+n \bmod p}$ for $1 \leq n \leq p$ and thus $f^p(x) = x$. Since f is monotone on each typical interval, it follows that for each $1 \leq n \leq p$, $I_{n+1 \bmod p}$ is the only A^y-basic interval f-covered by I_n. This contradicts the existence of the arrow $I_r \to J$. ■

Let (T, A, f) be a monotone model. We say that (T, A, f) is *orbital* if A contains a unique periodic orbit which is not a fixed point and there is at most one endpoint of T that does not belong to this periodic orbit. Observe that there exists $n \geq 0$ such that, for each $x \in A$, $f^n(x)$ belongs to the periodic orbit. Then we will also say that (T, A, f) is *n-orbital.* We note that an n-orbital model is also $(n + k)$-orbital for all $k \geq 0$. Obviously if A is a periodic orbit then (T, A, f) is 0-orbital.

Given a map f and an f-invariant set A containing a unique periodic orbit, we will denote this periodic orbit by A°.

Theorem 1.4.7. *Let (T, A, f) be a y-expansive n-orbital model. Let p be a type of A^y and let q be a rotation index associated to the type p. If $|A^\circ| \notin p\mathbb{N}$ then $\mathcal{E}(T, A, f) \neq \emptyset$ and $\mathrm{Per}(f) \supset \{(|A^\circ| + lp)i + pj : i, j \geq 1\}$ for some $0 \leq l \leq |A^\circ| + q + n - 1$. Furthermore, if $n = 0$ then $lp \leq p + q - (q \bmod p)$.*

Proof. In the whole proof, the subindexes will be considered modulo p. Let α be the typical loop starting at I_p. We can assume without loss of generality (by reindexing, if necessary) that $q = q_p$. Note that the assumption $|A^\circ| \notin p\mathbb{N}$ implies, in particular, that $p > 1$.

Since (T, A, f) is y-expansive and n-orbital, there exists $z \in f^r([y, x_p]) \cap A^\circ$ for some $r \leq q + n$. Furthermore, $|\operatorname{En}(T) \setminus A^\circ| \leq 1$ and thus each y-branch $Z_1, Z_2, \ldots, Z_p$ (except, at most, one of them) contains at least one endpoint of T which belongs to A°. Since $z \in A^\circ$ and A° is a periodic orbit, it follows easily that there exists $s \leq |A^\circ| - p$ such that $f^s(z) \succeq x_j$ for some $j \in \{1, 2, \ldots, p\}$.

Since $y \prec x_j \preceq f^s(z)$, we have that $I_j = [y, x_j] \subset [y, f^s(z)]$ and therefore $f([y, f^s(z)]) \supset f(I_j) \supset I_{j+1}$. In other words, $[y, f^s(z)]$ f-covers I_{j+1}. Furthermore, I_p f^r-covers $[y, z]$, $[y, z]$ f^s-covers $[y, f^s(z)]$ and $[y, f^s(z)]$ $f^{|A^\circ|}$-covers itself. Therefore we have the following sequence of coverings:

$$I_p \xrightarrow{f^r} [y, z] \xrightarrow{f^s} [y, f^s(z)] \xrightarrow{f^{|A^\circ|}} [y, f^s(z)] \to I_{j+1} \to I_{j+2} \to \ldots \to I_p.$$

Then, by using Lemma 1.3.1 by backwards induction, we obtain a loop γ in the f-graph of A^y such that $|\gamma| = r + s + |A^\circ| + p - j$. On the other hand, we can also consider the following sequence of coverings:

$$I_p \xrightarrow{f^r} [y, z] \xrightarrow{f^s} [y, f^s(z)] \to I_{j+1} \to I_{j+2} \to \ldots \to I_p.$$

Again by using Lemma 1.3.1 by backwards induction we obtain a loop σ in the f-graph of A^y such that $|\sigma| = r+s+p-j$. Let β be the loop $\gamma\sigma^{p-1}$, whose length is $|A^\circ|+lp$ with $l = r+s+p-j$. Note that $l \leq q+n+|A^\circ|-p+p-j \leq q + n + |A^\circ| - 1$. We claim that β is external. Indeed, if all the intervals of β were typical, by Remark 1.4.5, β would be a repetition of α and then $|\beta| \in p\mathbb{N}$, in contradiction with the fact that $|A^\circ| \notin p\mathbb{N}$. This proves the claim. By Lemma 1.4.6 we obtain that $\operatorname{Per}(f) \supset \{(|A^\circ|+lp)i+pj : i, j \geq 1\}$.

Finally note that, when $n = 0$, $A = A^\circ$. Moreover, it is not difficult to see that Lemma 1.4.4 gives $r = q$, $s = 0$, $z = x_r$ and $j = r \bmod p$ in the above construction of the loop β. Hence $|\beta| = |A^\circ|+lp = |A^\circ|+q+p-(q \bmod p)$. ■

1.5 Canonical and Monotone Models

In this section we use the notion of a *canonical model* introduced in [3]. From a monotone model (S, B, g), a canonical model (T, A, f) can be constructed, essentially, by collapsing the $V(S)$-basic intervals whose orbit does not intersect B. We prove that $\operatorname{Per}(f) \subset \operatorname{Per}(g)$ and that $\operatorname{Per}(g) \setminus \operatorname{Per}(f)$ is finite.

We start by recalling the definition of a canonical model. Let (S, B, g) be a monotone model. We will say that $v_1, v_2 \in V(S) \setminus B$ are *g-identifiable* if either:

(i) $[g^i(v_1), g^i(v_2)] \cap B = \emptyset$ for all $i \geq 0$, or
(ii) if $[g^n(v_1), g^n(v_2)] \cap B \neq \emptyset$ for some $n \geq 0$ then $g^n(v_1) = g^n(v_2)$.

Since g is B-monotone, it is easy to check that the g-identifiability is an equivalence relation. Moreover, since $V(S)$ is finite, there are finitely many equivalence classes.

Remark 1.5.1. From Remark 1.2.1 it follows that:
(i) If v_1, v_2 are g-identifiable then $g^i(v_1), g^i(v_2)$ are g-identifiable for each $i \geq 0$ such that $g^i(v_1), g^i(v_2) \in V(S) \setminus B$.
(ii) If v_1 and v_2 are g-identifiable and $v_3 \in [v_1, v_2] \cap V(S)$ then v_1, v_2, v_3 are pairwise g-identifiable.

□

A monotone model (T, A, f) such that every class of the f-identifiability relation contains exactly one point will be called a *canonical model.*

The following technical lemma is used in the proof of Theorem 1.5.3.

Lemma 1.5.2. *Let (S, B, g) be a monotone model and let $[v, v']$ be a $V(S)$-basic interval such that $v, v' \in V(S) \setminus B$ are g-identifiable. Let $x \in (v, v')$. Then either x is not periodic or there exist k, n, n' such that $g^k(v)$ is n-periodic, $g^k(v')$ is n'-periodic and $x \in \operatorname{Fix}(g^m)$, where m is the least common multiple of n and n'.*

Proof. Since v and v' are g-identifiable, $[v, v'] \cap B = \emptyset$. Furthermore, the B-monotonicity of g implies that g^i is monotone on $[v, v']$ for every $i \geq 0$ such that $[g^i(v), g^i(v')] \cap B = \emptyset$. In particular, $g^i([v, v']) = [g^i(v), g^i(v')]$ and thus $g^i(x) \in [g^i(v), g^i(v')]$. Hence, if there exists $n \geq 1$ such that $g^n(v) = g^n(v') \in B$ then $g^n([v, v'])$ reduces to a point of B. Therefore, there are no periodic points in (v, v') and we are done.

Assume now that $[g^i(v), g^i(v')] \cap B = \emptyset$ for all $i \geq 0$. Since $V(S)$ is finite, there exist $r, r' \geq 0$ such that $g^r(v)$ and $g^{r'}(v')$ are periodic points. Take $k = \max\{r, r'\}$. Then $g^k(v)$ and $g^k(v')$ are periodic points. Let n and n' be their respective periods, and let m be the least common multiple of n and n'. Then $g^k(v)$ and $g^k(v')$ are fixed points of g^m. Since g^m is monotone on $[g^k(v), g^k(v')]$, it follows that $\operatorname{Per}(g^m|_{[f^k(v), f^k(v')]}) = \{1\}$. Therefore, either $g^k(x)$ is not periodic or $g^k(x)$ is a fixed point of g^m. Observe that if x is periodic then the periods of x and $g^k(x)$ are the same. Thus either x is not periodic or it is a fixed point of g^m. This ends the proof. ■

Theorem 1.5.3. *Let (S, B, g) be a monotone model. There exists a canonical model (T, A, f) and a (possibly empty) finite set $\mathcal{V}$ such that*

$$\operatorname{Per}(g) = \operatorname{Per}(f) \cup \mathcal{V}$$

and each element of $\mathcal{V}$ divides the least common multiple of the periods of all periodic orbits of g contained in $V(S)$. Moreover, $|A| = |B|$ and if (S, B, g) is k-orbital then (T, A, f) is k-orbital.

Proof. Let K be the union of the convex hulls of all the classes of the g-identifiability relation. We remark that K has finitely many connected components, each of them contained in a connected component of $S \setminus B$. Let T be the tree obtained by contracting each connected component of K to a point and let $\phi : S \longrightarrow T$ be the standard projection. That is, ϕ is injective in a neighborhood of each point which does not belong to K, and the image of each point in a connected component C of K is the point to which C is contracted.

Define $f : T \longrightarrow T$ by $f(x) = \phi(g(x'))$ where $x' \in \phi^{-1}(x)$. By Remark 1.5.1, f is well defined. Set $A = \phi(B)$. Then $|A| = |B|$ and the fact that g is B-monotone implies that f is A-monotone. Furthermore, if $v, v' \in V(T) \setminus A$ are f-identifiable then $v = v'$. Hence (T, A, f) is a canonical model. Moreover, since $\mathrm{En}(T) = \phi(\mathrm{En}(S))$ and $f \circ \phi = \phi \circ g$, we easily get that if (S, B, g) is k-orbital then (T, A, f) is k-orbital.

To end the proof of the theorem, it remains to show that $\mathrm{Per}(g) = \mathrm{Per}(f) \cup \mathcal{V}$ for a finite set $\mathcal{V}$ verifying the prescribed properties. To do it, we claim that $B \cup K$ is g-invariant. Let us prove the claim. Since B is g-invariant, it is enough to show that the orbit of each point of K lies in $B \cup K$. Let $x \in K$. Assume first that $x \in V(S)$. Then $f^i(x) \in V(S) \cup B$ for all $i \geq 0$. Since each vertex of S belongs either to B or to its own g-identifiability class, we have that $V(S) \subset B \cup K$. Thus the claim follows in this case. Assume now that $x \notin V(S)$. Then there exist $v, v' \in V(S) \setminus B$ such that v and v' are f-identifiable and $x \in (v, v')$. By Remark 1.5.1, $g^i(v)$ and $g^i(v')$ are f-identifiable for each $i \geq 0$ such that $[g^i(v), g^i(v')] \cap B = \emptyset$. So, $[g^i(v), g^i(v')] \subset K$. Furthermore, since g is B-monotone, $g^i(x) \in [g^i(v), g^i(v')]$. Then it is clear that $g^i(x) \in K \cup B$ for all $i \geq 0$. Thus the claim is proved.

Since $B \cup K$ is g-invariant and $f \circ \phi = \phi \circ g$, $\phi(B \cup K)$ is f-invariant. Clearly, if $x \in S \setminus (K \cup B)$ is a periodic point of g then $\mathrm{Orb}_g(x) \subset S \setminus (B \cup K)$. Furthermore, $\phi(x)$ is a periodic point of f of the same period, and $\mathrm{Orb}_f(\phi(x)) \subset T \setminus \phi(K \cup B)$. Conversely, if $x \in T \setminus \phi(K \cup B)$ is a periodic point of f then $\mathrm{Orb}_f(x) \subset T \setminus \phi(K \cup B)$, $\phi^{-1}(x)$ is a periodic point of g of the same period and $\mathrm{Orb}_g(\phi^{-1}(x)) \subset S \setminus (K \cup B)$. Therefore, in order to complete the proof it is enough to show that

$$\mathrm{Per}(g|_{K \cup B}) = \mathrm{Per}(f|_{\phi(K \cup B)}) \cup \mathcal{V}$$

for some finite (or empty) set $\mathcal{V}$ satisfying the prescribed properties. From Lemma 1.5.2 and the fact that $V(S)$ is finite, we easily get that $\mathrm{Per}(g|_{K \cup B})$ is finite. Furthermore, for each n-periodic orbit of $g|_{K\cup B}$ there exist two periodic orbits of g contained in $V(S)$ in such a way that n divides the least common multiple of their periods. Thus it suffices to show that $\operatorname{Per}(f|_{\phi(K\cup B)}) \subset \operatorname{Per}(g|_{K\cup B})$.

Since $f \circ \phi|_B = g|_B$, it is enough to show that for each n-periodic point of f in $\phi(K)$ there exists an n-periodic point of g in K. Let $x \in \phi(K)$ be an n-periodic point of f. Let $K_i = \phi^{-1}(f^i(x))$ for $i = 1, 2, \dots, n$. By the definition of ϕ, each K_i is the convex hull of a class of g-identifiability and contains points of $V(S)\backslash B$. Furthermore, $K_i \neq K_j$ if $i \neq j$ since $f^i(x) \neq f^j(x)$. By the definition of f, for $i = 1, 2, \dots, n$ we have that $f^{i+1}(x) = f(f^i(x)) = \phi(g(x_i))$ for some $x_i \in \phi^{-1}(f^i(x)) = K_i$. We choose $x_i \in V(K_i)$ for each i. Take $i \in \{1, 2, \dots, n\}$. Then we have that $\phi(g(x_i)) = f(\phi(x_i)) = f(f^i(x)) = f^{i+1}(x)$. Therefore, $g(x_i) \in K_{i+1 \bmod n}$. Moreover, $g(K_i) \subset K_{i+1 \bmod n}$. Indeed, for each $z \in K_i$ there exists $v \in \operatorname{En}(K_i)$ such that $z \in [v, x_i]$ and g is monotone on $[v, x_i]$. Since $g(x_i) \in K_{i+1 \bmod n}$ and $g(v)$ and $g(x_i)$ are g-identifiable, $g(v) \in K_{i+1 \bmod n}$ and thus $g(z) \in K_{i+1 \bmod n}$.

From above we have $g^i(K_1) \subset K_{i+1 \bmod n}$ for $i \geq 0$, and hence $g^n(K_1) \subset K_1$. Then there exists a fixed point of g^n in K_1, which is obviously a point of period n of g. ∎

Let (S, P, g) be a monotone model and let (T, A, f) be the canonical model constructed from (S, P, g) as in the proof of Theorem 1.5.3. We will say that (S, P, g) and (T, A, f) are *associated* to each other. With this notion, Theorem 1.5.3 can be restated as follows: *each monotone model admits an associated canonical model.* This theorem allows us to restrict our attention to the study of the set of periods of canonical models rather than to generic monotone models.

The following proposition says that a canonical model is y-expansive, and therefore all the results of Section 1.4 can be applied to canonical models. This fact will be used in the rest of the chapter.

Proposition 1.5.4. *If (T, A, f) is an orbital canonical model, then there exists a fixed point y of f such that (T, A, f) is y-expansive.*

Proof. Since (T, A, f) is orbital, A does not contain fixed points and therefore $\operatorname{Fix}(f) \setminus A \neq \emptyset$. If $V(T) \cap \operatorname{Fix}(f) \neq \emptyset$, we take $y \in V(T) \cap \operatorname{Fix}(f)$. Otherwise we take any $y \in \operatorname{Fix}(f)$. Let $v \in (V(T) \cap Z^\star(A)) \setminus \{y\}$ (if v does not exist then (T, A, f) is obviously y-expansive). By Remark 1.2.1, $\operatorname{Orb}(v) \subset A \cup V(T)$. Assume that $\operatorname{Orb}(v) \subset Z^\star(A)$ (in particular, $\operatorname{Orb}(v) \cap A = \emptyset$ and hence $\operatorname{Orb}(v) \subset V(T)$) and we will arrive to a contradiction. If $v \in \operatorname{Fix}(f)$ then the choice of y implies that $y \in V(T)$. Since f is A-monotone and $[y, v] \cap A = \emptyset$,

$[y, v] = [f^i(y), f^i(v)]$ for each $i \geq 0$. Thus y and v are f-identifiable, a contradiction with the fact that (T, A, f) is a canonical model.

Assume now that there exist $z, z' \in \mathrm{Orb}(v) \subset V(T)$ such that $z \neq z'$. Then, as above, the A-monotonicity of f implies that $[f^i(z), f^i(z')] \cap A = \emptyset$ for each $i \geq 0$. So z and z' are f-identifiable, a contradiction with the fact that (T, A, f) is a canonical model. ■

1.6 Reduction of Monotone Models

When the Markov graph of a canonical model (T, A, f) contains external loops, we can calculate the set of periods of f by means of Lemma 1.4.6 and Theorem 1.4.7. If the Markov graph of (T, A, f) has no external loops, we will perform the strategy described in Section 1.2. This is done in Theorem 1.6.7, where we construct a sequence of partial reductions associated to the model (T, A, f). The proof of this theorem depends strongly on the notion of *twist model* and makes use of Propositions 1.6.4 and 1.6.5.

Let (T, A, f) be a y-expansive model. We will say that (T, A, f) is *twist around* y if $f(Z_i) \cap Z^\star = \emptyset$ for $i \in \{1, 2, \ldots, n^\star\}$. Otherwise we will say that (T, A, f) is *non-twist around* y.

Note that if (T, A, f) is twist around y and p is a type of A^y then, from the definition of a type and the A^y-monotonicity of f, it follows that $f(Z_i) \subset Z_{i+1 \bmod p}$ for each $1 \leq i \leq p$. Since A^y contains the set of vertices of T, $\mathrm{Cl}(Z^\star)$ is a star whose set of endpoints contains $\{x_1, x_2, \ldots, x_p\}$, and a unique y-branch hangs from each of these endpoints. This rotational behavior of f around the fixed point y justifies the terminology of a *twist model around* y.

Remark 1.6.1. When (T, A, f) is twist around a fixed point y, each A^y-basic interval contained in a y-branch does not f-cover any typical interval. Consequently, there cannot exist external loops in the Markov f-graph of A^y. That is, $\mathcal{E}(T, A, f) = \emptyset$. □

Given an orbital y-expansive model (T, A, f), by definition, there is at most one y-branch containing no points of A°. Such a y-branch (if it exists) contains exactly one endpoint of T and so it is an interval. We will call it the *residual branch.* From now on, the number of y-branches containing points of A° will be denoted by n°.

Remark 1.6.2. Let (T, A, f) be an orbital y-expansive model. By definition, A does not contain fixed points and thus $y \notin A$. Furthermore, since (T, A, f) is a monotone model, $\mathrm{En}(T) \subset A$. Therefore, $y \notin \mathrm{En}(T)$ and it follows that $n^\star \geq 2$. On the other hand, from the fact that (T, A, f) is orbital we have that $n^\star$ is either n° or $n^\circ + 1$, and $n^\star = n^\circ + 1$ if and only if there exists a residual branch. In summary, we have:

(i) $n^\star \geq 2$.

(ii) $n^\star \in \{n^\circ, n^\circ + 1\}$, and there exists a residual branch if and only if $n^\star = n^\circ + 1$.

□

The next lemma establishes some properties of the type of A^y when (T, A, f) is a twist model around y.

Lemma 1.6.3. *Let (T, A, f) be a y-expansive orbital model which is twist around y. Then A^y has a unique type and it coincides with n°.*

Proof. Assume that $X(A^y)$ contains two different periodic orbits of Φ_{A^y} of periods p and q. Then, since (T, A, f) is twist around y, there exist two subsets $\mathcal{Z} = \{Z_1, Z_2, \dots, Z_p\}$ and $\mathcal{W} = \{W_1, W_2, \dots, W_q\}$ of the set of y-branches such that $\mathcal{Z} \cap \mathcal{W} = \emptyset$, $f(Z_i) \subset Z_{i+1 \bmod p}$ for $i = 1, 2, \dots, p$ and $f(W_i) \subset W_{i+1 \bmod q}$ for $i = 1, 2, \dots, q$. Furthermore, by the definition of the y-branches we have that $Z_i \cap W_j = \emptyset$ for $1 \leq i \leq p$ and $1 \leq j \leq q$. Let $z \in A \cap Z_p$. Then $f^i(z) \in Z_{i \bmod p}$ for every $i \geq 0$. Since (T, A, f) is orbital, there is a $k \geq 0$ such that $f^k(z) \in A^\circ$. Consequently, $A^\circ \subset Z_1 \cup Z_2 \cup \ldots \cup Z_p$. But analogously, by taking some $w \in A \cap W_q$, we get that $A^\circ \subset W_1 \cup W_2 \cup \ldots \cup W_q$, a contradiction.

Let P be the the unique periodic orbit of Φ_{A^y} and let $p = |P|$. Then $p \leq n^\star$. By Remark 1.6.2, $n^\star \in \{n^\circ, n^\circ + 1\}$. Now we claim that $p \leq n^\circ$. Indeed, assume that $p = n^\star = n^\circ + 1$. Then there is one residual branch S and the unique point z of $X(A^y) \cap S$ belongs to P. By Remark 1.6.2, $p \geq 2$. Therefore, there exists another y-branch S' such that if z' is the only point of $X(A^y) \cap S'$ then $\Phi_{A^y}(z') = z$. In other words, $f(z') \in S$. Therefore, since (T, A, f) is twist around y, $f(S') \subset S$. In particular, $f(A^\circ \cap S') \subset S$, in contradiction with the fact that S is the residual branch. This proves the claim.

To prove $n^\circ = p$ we must see that $n^\circ \leq p$. It is enough to show that, given a y-branch S such that $S \cap A^\circ \neq \emptyset$, then $z \in P$ where z is the unique point of $X(A^y) \cap S$. On the contrary, since P is the unique periodic orbit of Φ_{A^y}, $\Phi^i_{A^y}(z) \neq z$ for all $i > 0$. Since (T, A, f) is twist around y, it follows that $f^i(S) \cap S = \emptyset$ for all $i > 0$. Take $z' \in A^\circ \cap S$. Then $f^{|A^\circ|}(z') = z' \in S$, a contradiction. ■

Proposition 1.6.4. *Let (T, A, f) be an n-orbital canonical model which is twist around a fixed point y and let p be the type of A^y. Then there exist a y-branch S and a finite set $B \subset S$ such that the following properties hold for $g = f^p|_S$:*

(a) (S, B, g) *is a canonical model.*

(b) B *contains a unique periodic orbit of* g*. Furthermore,* $B^\circ = A^\circ \cap S$ *and* $|B^\circ| = |A^\circ|/p$*.*

(c) If $|B^\circ| > 1$ *then* (S, B, g) *is* $(n+1)$*-orbital if* $n \in \{0, 1\}$ *and* n*-orbital if* $n \geq 2$*.*

(d) $\mathrm{Per}(f) \supset p \cdot \mathrm{Per}(g)$*.*

(e) If $p = 1$ *then* $A^\circ \subset B \subsetneq A$*.*

Proof. By Lemma 1.6.3 and Remark 1.6.2 we have $p = n^\circ$, $n^\star \geq 2$ and $n^\star \in \{n^\circ, n^\circ + 1\}$. Since $|\operatorname{En}(T) \setminus A^\circ| \leq 1$ and $n^\star \geq 2$, we can choose S to be a y-branch such that $\operatorname{En}(T) \cap S \subset A^\circ$. Without loss of generality, we can assume that $S = Z_1$.

In order to prove (d) it is enough to see that each k-periodic point of g is a kp-periodic point of f. This is a direct consequence of the fact that (T, A, f) is twist around y and the definitions of g and S.

Now we prove the other statements when $p = n^\circ = 1$. In this case, there are two y-branches: S and the residual one. Moreover, $A^\circ \subset S$. Since (T, A, f) is twist around y, $f(S) \subset S$. We take $B = A \cap S$. Thus A° is the only periodic orbit contained in B, and (b) and (e) hold. Since $\operatorname{En}(T) \cap S \subset A^\circ$, the only endpoint of S which possibly does not belong to A° is x_1. Hence $|\operatorname{En}(S) \setminus A^\circ| \leq 1$. It is obvious that (S, B, g) is n-orbital and thus (c) is satisfied. Finally, it is not difficult to prove that (S, B, g) is a canonical model. Thus (a) holds and we are done in this case.

Now we consider the case $p = n^\circ \geq 2$. Observe that the set $f^{-p}(A) \cap S$ is not necessarily finite, but the A-monotonicity of f implies that it has finitely many connected components, each of them being either a point or a subtree on which f^p is constant. Note that $A \cap S \subset f^{-p}(A) \cap S$. Then we construct the set B by taking all the points of $A \cap S$ and all vertices $V(K)$ for each connected component K of $f^{-p}(A) \cap S$. Thus B is finite and $A \cap S \subset B$.

Since A° is a periodic orbit and (T, A, f) is twist around y, we get that $|A^\circ \cap Z_i| = |A^\circ|/p$ for $i \in \{1, 2, \ldots, p\}$. Moreover, $g(S) \subset S$, $A^\circ \cap S$ is a periodic orbit of g of period $|A^\circ|/p$ and $B \subset f^{-p}(A \cap S) \cap S = g^{-1}(A \cap S)$. Thus $g(B) \subset A \cap S \subset B$ and hence B is g-invariant. On the other hand, $A^\circ \cap S$ is the only periodic orbit of g contained in B. Therefore, $B^\circ = A^\circ \cap S$ and (b) holds.

Next we prove (c). Assume that $|B^\circ| > 1$. Since $\operatorname{En}(T) \cap S \subset A^\circ$, the only element of $\operatorname{En}(S)$ which possibly does not belong to B° is x_1, and so we have that $|\operatorname{En}(S) \setminus B^\circ| \leq 1$. To finish the proof of (c) we claim that for each $x \in B$, $g^{n+1}(x) \in B^\circ$ if $n \in \{0, 1\}$ and $g^n(x) \in B^\circ$ if $n \geq 2$. To prove the claim, set $n = pq + r$ with $q \geq 0$ and $0 \leq r < p$. Since $x \in B$, $f^p(x) \in A \cap S$. Therefore, since $S = Z_1$ and (T, A, f) is n-orbital and twist, we have that

$f^n(f^p(x)) = f^{n+p}(x) = f^{(q+1)p+r}(x) \in A^\circ \cap Z_{r+1}$. Hence, for $i \geq 0$ we have

$$f^{n+p+i}(x) \in A^\circ \cap Z_{r+1+i \bmod p} \ . \tag{1.2}$$

When $n = 0$ we have $r = 0$, and by taking $i = 0$ in (1.2) we get that $g(x) = f^p(x) \in A^\circ \cap Z_1 = B^\circ$. If $n = 1$, since $p > 1$ we have $q = 0$ and $r = 1$. Then, by taking $i = p-1$ in (1.2) we get that $g^2(x) = f^{2p}(x) \in A^\circ \cap Z_1 = B^\circ$. Finally, when $n \geq 2$ we take $i = pn - n - p$. Since $p \geq 2$ and $n \geq 2$, it follows that $i \geq 0$. Then from (1.2) we obtain that $g^n(x) = f^{pn}(x) \in A^\circ \cap Z_{r+1+(pn-pq-r-p) \bmod p} = A^\circ \cap Z_1 = B^\circ$. This ends the proof of the claim, and hence (c) follows.

Finally we must prove (a), i.e. that (S, B, g) is a canonical model. First we will show that g is B-monotone. Let $[x, z]$ be an interval such that $[x, z] \cap B = \{x, z\}$. Since $g = f^p$, we must see that $f^p([x, z]) = [f^p(x), f^p(z)]$ and $f^p|_{[x,z]}$ is monotone. From the definition of B, we have that either $[x, z]$ is contained in a connected component of $f^{-p}(A) \cap S$ and thus $f^p([x, z])$ reduces to a point of A, or $(x, z) \cap f^{-p}(A) = \emptyset$. In the first case it is obvious that $f^p([x, z]) = [f^p(x), f^p(z)]$ and $f^p|_{[x,z]}$ is monotone. Now assume that $(x, z) \cap f^{-p}(A) = \emptyset$. Since A is f-invariant, $(x, z) \cap f^{-i}(A) = \emptyset$ for $0 \leq i < p$. Since $(x, z) \cap A = \emptyset$ and $\mathrm{En}(T) \subset A$, there exists a minimal interval in T (with respect to the inclusion relation) containing $[x, z]$ whose endpoints belong to A. Since f is A-monotone we have that $f|_{[x,z]}$ is monotone. In particular, $f([x, z]) = [f(x), f(z)]$. Moreover, $(f(x), f(z)) \cap A = \emptyset$, since otherwise $(x, z) \cap f^{-1}(A) \neq \emptyset$, a contradiction. In the same way, it can be proved inductively that $f^i|_{[x,z]}$ is monotone for each $1 < i \leq p$. Therefore, g is B-monotone.

To complete the proof we must show that there are no g-identifiable vertices. On the contrary, assume that there exist $v_1, v_2 \in V(S) \setminus B$ that are g-identifiable. Since the only possible point of $V(S) \setminus V(T)$ is the unique point of $X(A^y) \cap S$, which belongs to B, we have that $v_1, v_2 \in V(T)$. We consider two cases.

In the first case we assume that $[g^i(v_1), g^i(v_2)] \cap B = \emptyset$ for $i \geq 0$. In other words, $[f^{ip}(v_1), f^{ip}(v_2)] \cap B = \emptyset$ for $i \geq 0$. Moreover, since g is B-monotone, we have $[g^i(v_1), g^i(v_2)] = g^i([v_1, v_2])$ for $i \geq 0$. Since (T, A, f) is a canonical model, v_1 and v_2 are not f-identifiable. Therefore, there exists $j \geq 1$ (which we take as small as possible) such that $[f^j(v_1), f^j(v_2)] \cap A \neq \emptyset$ and $f^j(v_1) \neq f^j(v_2)$. Since f is A-monotone, $[f^j(v_1), f^j(v_2)] = f^j([v_1, v_2])$. Take $k \in \mathbb{N}$ such that $kp > j$. Since A is f-invariant, $f^{kp}([v_1, v_2]) \cap A \neq \emptyset$. Then $\emptyset \neq A \cap g^k([v_1, v_2]) = A \cap [g^k(v_1), g^k(v_2)] \subset B \cap [g^k(v_1), g^k(v_2)]$, a contradiction. This ends the proof of the proposition in this case.

Secondly, assume that there is a $j \geq 1$ such that $[g^i(v_1), g^i(v_2)] \cap B = \emptyset$ for $0 \leq i < j$ and $g^j(v_1) = g^j(v_2) \in B$. In other words, $[f^{ip}(v_1), f^{ip}(v_2)] \cap B = \emptyset$

for $0 \le i < j$ and $f^{jp}(v_1) = f^{jp}(v_2) \in B$. Moreover, since g is B-monotone, we have that $[f^{ip}(v_1), f^{ip}(v_2)] = f^{ip}([v_1, v_2])$ for $0 \le i \le j$. In particular,

$$f^{jp}([v_1, v_2]) = \{f^{jp}(v_1)\} = \{f^{jp}(v_2)\}. \tag{1.3}$$

Since $[f^{jp-p}(v_1), f^{jp-p}(v_2)] \cap B = \emptyset$, from the definition of B it follows that $[f^{jp-p}(v_1), f^{jp-p}(v_2)]$ does not intersect any connected component of $f^{-p}(A) \cap S$. Thus

$$f^{jp}(v_1) = f^{jp}(v_2) \in B \setminus A. \tag{1.4}$$

Since (T, A, f) is a canonical model, v_1 and v_2 are not f-identifiable. Therefore, there exists some $k \ge 1$ (which we take as small as possible) such that $[f^k(v_1), f^k(v_2)] \cap A \ne \emptyset$ and $f^k(v_1) \ne f^k(v_2)$. From (1.4) it follows that $k < jp$. Since f is A-monotone, $[f^k(v_1), f^k(v_2)] = f^k([v_1, v_2])$. Then, since $[f^k(v_1), f^k(v_2)] \cap A \ne \emptyset$, we have that $\emptyset \ne f^{jp-k}([f^k(v_1), f^k(v_2)]) \cap A = f^{jp}([v_1, v_2]) \cap A$. Therefore, (1.4) and (1.3) are in contradiction to each other. ■

To compute the set of periods of a canonical model we will use a subclass of the external loops whose length satisfies certain properties. Now we establish a notation for this kind of loops. Let (T, A, f) be a a y-expansive n-orbital model. Let p be a type of A^y and let q be a rotation index associated to p. Then we define

$$\tilde{\mathcal{E}}(T, A, f) = \{\beta \in \mathcal{E}(T, A, f) : |\beta| \in p\mathbb{N}, |\beta| \le |A^\circ| + p + q + n + 1\}.$$

Proposition 1.6.5. *Let (T, A, f) be an n-orbital canonical model which is non-twist around a fixed point y. Let p be a type of A^y and let q be a rotation index of (T, A, f) associated to p. Then at least one of the following statements hold:*

(a) There exist a tree $S \subset T$ and a finite set $B \subset S$ such that $A^\circ \subset B \subsetneq A$ and $(S, B, f|_S)$ is an n-orbital canonical model;

(b) $\tilde{\mathcal{E}}(T, A, f) \ne \emptyset$.

In particular, (b) holds if $\operatorname{En}(T) \subset A^\circ$

Proof. Let W be the set of points $z \in A^y$ that satisfy the following two properties:

(i) There exists $N \in \{1, 2, \dots, p\}$ such that $z \in Z_N$ but $f(z) \notin Z_{N+1 \bmod p}$;

(ii) There exist $z' \in A^y \setminus \{z\}$ and $w \in A^\circ$ such that $z' \preceq w$ and $z \preceq f(z')$.

A sufficient condition for (ii) is the following property:

(ii') There exists $z'' \in A^\circ$ such that $z \preceq z''$.

To see it, take $w = z'$ as the unique point of $f^{-1}(z'') \cap A^\circ$ when $\{z\} \neq f^{-1}(z'') \cap A^\circ$, and take $w = z'$ as the unique point of $f^{-1}(z) \cap A^\circ$ otherwise.

We start by claiming that if $\operatorname{En}(T) \subset A^\circ$ then $W \neq \emptyset$. To prove the claim assume that $W = \emptyset$. Since in this case (ii') holds for every $z \in A^y$, we see that (i) does not hold for any $z \in Z(A^y)$. Thus $f(A^y \cap Z_i) \subset Z_{i+1 \bmod p}$ for $i = 1, 2, \ldots, p$. Then, by the A^y-monotonicity of f, we have $f(Z_i) \subset Z_{i+1 \bmod p}$ for $i = 1, 2, \ldots, p$. Since A° is a periodic orbit and $\operatorname{En}(T) \subset A^\circ$, we easily get that $n^\star = n^\circ = p$. Therefore $f(Z_i) \cap Z^\star = \emptyset$ for $i = 1, 2, \ldots, n^\star$ and so (T, A, f) is twist around y, in contradiction with the hypotheses. This proves the claim.

To prove the proposition we consider two cases. First we assume that $W \neq \emptyset$ and we prove that (b) holds. In the proof of this case, the subindexes will be considered modulo p. Let $k \in \{1, 2, \ldots, p\}$ be such that $q = q_k$. By the A-monotonicity of f and the definition of q, $f([y, f^{i-1}(x_k)]) = [y, f^i(x_k)]$ for $1 \leq i \leq q$ and $[y, f^q(x_k)] \cap A \neq \emptyset$. We have $f^q(x_k) \in Z_{k+q}$. This is obvious if $q = 0$ and it follows from Lemma 1.4.4 if $q > 0$.

Let $a \in [y, f^q(x_k)] \cap A$. Take $z \in W$ and let $N \in \{1, 2, \ldots, p\}$, $w \in A^\circ$ and $z' \in A^y \setminus \{z\}$ be such that $z \in Z_N$, $f(z) \notin Z_{N+1}$, $z' \preceq w$ and $z \preceq f(z')$. Since (T, A, f) is n-orbital and $w \in A^\circ$, there exists $s \leq n + |A^\circ| - 1$ such that $f^s(a) = w$. Thus $z' \preceq f^s(a)$. If

$$f^i([y, f^q(x_k)]) \subset [y, x_{k+q+i}] \cup Z_{k+q+i} \tag{1.5}$$

is satisfied for each $0 \leq i \leq s+1$, then we have that $z' \in f^s([y, f^q(x_k)])$, $z \in f^{s+1}([y, f^q(x_k)])$, $Z_N = Z_{k+q+s+1}$ and $f(z) \notin Z_{k+q+s+2}$. Summarizing, there exists a minimum non-negative integer $t \leq s+1 \leq n + |A^\circ|$ such that (1.5) holds for each $0 \leq i \leq t$ and $f^t([y, f^q(x_k)])$ contains a point u whose image does not belong to $Z_{k+q+t+1}$.

Since $f(x_{k+q+t}) \in Z_{k+q+t+1}$, there exists a A^y-basic interval $L = [b, c] \subset [x_{k+q+t}, u]$ such that $f(b) \in Z_{k+q+t+1}$ and $f(c) \notin Z_{k+q+t+1}$. Then L f-covers $[y, x_{k+q+t+1}] = I_{k+q+t+1}$. By using $q + t$ times Lemma 1.3.1 by backwards induction we obtain the following loop β in the f-graph of A^y:

$$I_k \to J^1 \to J^2 \to \ldots \to J^{q+t-1} \to L \to I_{k+q+t+1} \to I_{k+q+t+2} \to \ldots \to I_k$$

where J^i is a A^y-basic interval contained in $f^i(I_k)$ for each $1 \leq i \leq q+t-1$. Since L is not a typical interval, β is an external loop, and $|\beta| = q+t+1+p-(q+t+1 \bmod p) \in p\mathbb{N}$. Observe that $|\beta| \leq q+t+1+p \leq q+n+|A^\circ|+1+p$. Hence, (b) holds when $W \neq \emptyset$ and, in particular, when $\operatorname{En}(T) \subset A^\circ$.

From now on we assume that $W = \emptyset$. From the above claim, $\operatorname{En}(T) \not\subset A^\circ$ and thus $|\operatorname{En}(T) \setminus A^\circ| = 1$. Hence, there is a unique y-branch containing some endpoint which does not belong to A°. By Remark 1.6.2, $n^\star \geq 2$ and

$n^\star \in \{n^\circ, n^\circ{+}1\}$. We also recall that the y-branches are labeled in such a way that $f(x_i) \in Z_{i+1 \bmod p}$ for $i = 1, 2, \ldots, p$. We shall consider the following cases:

Case 1. $p = 1$.

Assume that $Z_1 \cap A^\circ \neq \emptyset$. Each $z \in Z_1 \cap A^\circ$ verifies (ii') and, since $W = \emptyset$, it does not verify (i). In consequence, $A^\circ \subset Z_1$ and $n^\circ = 1$. Since $n^\star \in \{n^\circ, n^\circ + 1\}$ and $n^\star \geq 2$, it follows that $n^\star = 2$. Therefore Z_2 is the residual branch and $\mathrm{En}(T) \cap Z_1 \subset A^\circ$. In particular, each point in Z_1 verifies (ii'). Since $W = \emptyset$, no point in $Z_1 \cap A^y$ verifies (i) and thus $f(Z_1 \cap A^y) \subset Z_1$. Since f is A^y-monotone, it follows that $f(Z_1) \subset Z_1$. We set $S = Z_1$ and $B = A \cap S$. Then $B \subsetneq A$. It is not difficult to prove that $(S, B, f|_S)$ is an n-orbital canonical model. Therefore (a) holds and we are done in this case.

Now suppose that $Z_1 \cap A^\circ = \emptyset$. Then, from the fact that (T, A, f) is n-orbital, it follows that $f^r(x_1) \notin Z_1$ for some $r \leq n$ (which we take as small as possible). By the definition of type, $f(x_1) \in Z_1$ and hence I_1 f-covers $[x_1, f(x_1)]$. Also $[f^{i-1}(x_1), f^i(x_1)]$ f-covers $[f^i(x_1), f^{i+1}(x_1)]$ for $1 \leq i \leq r-2$ and $[f^{r-2}(x_1), f^{r-1}(x_1)]$ f-covers $I_1 \subset [f^{r-1}(x_1), f^r(x_1)]$. By using Lemma 1.3.1 by backwards induction, as above we obtain a loop in the f-graph of A^y of length $r \leq n$. Since $[x_1, f(x_1)]$ does not contain typical intervals, this loop is external. Hence (b) holds in this case.

Case 2. $p > 1$ *and* $n^\star = n^\circ + 1$.

In this case there is a residual branch Z_i for some $i \in \{1, 2, \ldots, n^\star\}$. We claim that $i > p$. Indeed, if $i \leq p$ then, since $p > 1$, it follows that $i{-}1 \pmod{p} \neq i$. Hence, $Z_{i-1 \bmod p}$ is not residual. Since $|\,\mathrm{En}(T) \setminus A^\circ| = 1$, it follows that $Z_{i-1 \bmod p} \cap \mathrm{En}(T) \subset A^\circ$. Thus each point in $A^y \cap Z_{i-1 \bmod p}$ verifies (ii') from the definition of W. On the other hand, each point $z \in A^\circ \cap Z_{i-1 \bmod p} \subset A^y \cap Z_{i-1 \bmod p}$ verifies $f(z) \notin Z_{i \bmod p}$. That is, it verifies (i). This implies $W \neq \emptyset$, a contradiction. This proves the claim.

From above, it follows that $\mathrm{En}(T) \cap Z_i \subset A^\circ$ for $i = 1, 2, \ldots, p$. Therefore, each $z \in A^y \cap Z_i$ satisfies (ii') from the definition of W and, since $W = \emptyset$, these points do not satisfy (i). Consequently, $f(A^y \cap Z_i) \subset Z_{i+1 \bmod p}$, $n^\circ = p$ and $Z_{n^\star} = Z_{n^\circ+1}$ is the residual branch. By the A^y-monotonicity, $f(Z_i) \subset Z_{i+1 \bmod p}$ and $f([y, x_i]) \subset [y, x_{i+1 \bmod p}] \cup Z_{i+1 \bmod p}$ for $i = 1, 2, \ldots, p$. Thus, if we define $S = \langle A^\circ \rangle_T = T \setminus (Z_{n^\star} \cup (y, x_{n^\star}])$, then $f(S) \subset S$. Finally, if we define $B = A \cap S$ then $B \subsetneq A$ and it is not difficult to prove that $(S, B, f|_S)$ is an n-orbital canonical model. Hence (a) holds and we are done.

Case 3. $p > 1$ *and* $n^\star = n^\circ$.

In this case, $A^\circ \cap Z_i \neq \emptyset$ for each $1 \leq i \leq n^\star$. We claim that, for some $N \in \{1, 2, \ldots, p\}$, there exists $a \in A^y \cap Z_N$ such that $f(a) \notin Z_{N+1 \bmod p}$. Indeed, when $p = n^\star$ the claim follows since (T, A, f) is non-twist around y. To end the proof of the claim we assume that $p < n^\star$ and $f(A^y \cap Z_i) \subset Z_{i+1 \bmod p}$ for $i = 1, 2, \ldots, p$. Then $A^\circ \subset Z_1 \cup Z_2 \cup \ldots \cup Z_p$ and $Z_{n^\star} \cap A^\circ = \emptyset$, a contradiction. Thus the claim follows.

Since a satisfies (i) from the definition of W and $W = \emptyset$ we have:

$$\{x \in A^\circ : x \succeq a\} = \emptyset. \tag{1.6}$$

Therefore, since $|\operatorname{En}(T) \setminus A^\circ| = 1$, the unique point e in $\operatorname{En}(T) \setminus A^\circ$ must satisfy $a \preceq e$, $[a, e] \cap V(T) = \{e\}$ and $[a, e] \cap A^\circ = \emptyset$. Since $Z_N \cap A^\circ \neq \emptyset$, there exists $v \in (V(T) \cup A^\circ) \cap Z_N$ such that $v \prec a$ and

$$(v, e) \cap (V(T) \cup A^\circ) = \emptyset. \tag{1.7}$$

Since $y \notin Z_N$, we get that $x \in (v, e) \cap A^y$ implies $x \in A \setminus A^\circ$. Let z be the minimum (with respect to the $\prec$ ordering) of the points of $(v, a] \cap A^y$ such that $f(z) \notin Z_{N+1 \bmod p}$ (this point exists since $f(a) \notin Z_{N+1 \bmod p}$). We have $x_N \preceq v \prec z \preceq a \preceq e$ and

$$f\Big((v, z) \cap A^y\Big) \subset Z_{N+1 \bmod p}. \tag{1.8}$$

Set $R = T \setminus (v, e]$. Observe that $R = \langle A^\circ \rangle_T \supset Z_{N+1 \bmod p}$. Clearly, for each point $z' \in R$ there exists $w \in \operatorname{En}(T) \cap A^\circ$ such that $z' \preceq w$. Consequently, if there exists $z' \in R \cap A^y$ such that $f(z') \in [z, e]$ (that is, $z \preceq f(z')$), it follows that z verifies (i) and (ii) from the definition of W; a contradiction since $W = \emptyset$. Therefore,

$$f(R \cap A^y) \cap [z, e] = \emptyset. \tag{1.9}$$

Furthermore, if the image of some $x \in R \cap A^y$ belongs to (v, z), then by (1.8) we have that $f^2(x) \in R \cap A^y$ and hence $f^2(x) \notin [z, e]$. This fact, together with (1.9), gives us that $f^i(R \cap A^y) \cap [z, e] = \emptyset$ for all $i \geq 0$. Since f is A^y-monotone, it follows that

$$f^i(R) \cap [z, e] = \emptyset \text{ for } i \geq 0. \tag{1.10}$$

We define $S = \operatorname{Cl}(\bigcup_{i \geq 0} f^i(R))$. Since $f(R) \supset R$, S is connected. That is, S is a subtree of T. Clearly, $f(S) \subset S$ and $R \subset S$. Moreover, by (1.10), $S \subset T \setminus (z, e]$. Thus there exists $v' \in \operatorname{En}(S)$ such that $v \preceq v' \preceq z$ and $S = T \setminus (v', e]$. From the definition of S and from the A^y-monotonicity of f we deduce immediately that $v' \in A^y$ (in fact, $S = f^{|A^y|}(R)$). Hence, $v' \in A \setminus A^\circ$. We define $B = A \cap S$. Observe that $A^\circ \subset B \subsetneq A$, since at least e does not belong to B. Clearly, $f(B) \subset B$. Moreover, it is not difficult to show that $(S, B, f|_S)$ is a canonical model. This model is n-orbital, since A° is the unique periodic orbit contained in B, and $\mathrm{En}(S) \setminus \{v'\} \subset A^\circ$. ■

In the rest of this section, we use recursively the above theorem to study the set of periods of a canonical model. To do it, we introduce the following notions.

Let (T, A, f) be a canonical model and let $p \in \mathbb{N}$. We say that a canonical model (T', A', f') is a *partial p-reduction of* (T, A, f) if $T' \subset T$, $f' = f^p|_{T'}$ and $\mathrm{Per}(f) \supset p\,\mathrm{Per}(f')$. Let (T, A, f) be a 2-orbital canonical model. A sequence $\{(T_i, A_i, f_i), y_i, p_i\}_{i=1}^m$ will be called a *sequence of partial reductions* of (T, A, f) if and only if:

(i) $(T_1, A_1, f_1) = (T, A, f)$

(ii) (T_i, A_i, f_i) is a y_i-expansive 2-orbital canonical model for $1 \le i < m$.

(iii) $(T_{i+1}, A_{i+1}, f_{i+1})$ is a partial p_i-reduction of (T_i, A_i, f_i) for $1 \le i < m$.

(iv) $|A_i{}^\circ| = p_i |A_{i+1}{}^\circ|$ for $1 \le i < m$. Moreover, $A_i{}^\circ \subset A_{i+1} \subsetneq A_i$ when $p_i = 1$.

(v) $\tilde{\mathcal{E}}(T_i, A_i, f_i) = \emptyset$ for $1 \le i < m$.

(vi) (T_m, A_m, f_m) is a canonical model such that A_m contains a unique periodic orbit and either

(vi.1) $|A_m{}^\circ| = 1$ and thus (T_m, A_m, f_m) is a trivial model

or

(vi.2) (T_m, A_m, f_m) is a y_m-expansive 2-orbital canonical model, p_m is a type of $A_m^{y_m}$ and $\tilde{\mathcal{E}}(T_m, A_m, f_m) \neq \emptyset$.

Observe that if $m = 1$ then, by (i) and (vi.2), (T, A, f) is a y_1-expansive 2-orbital canonical model, p_1 is a type of A^{y_1} and $\tilde{\mathcal{E}}(T, A, f) \neq \emptyset$.

Remark 1.6.6. Given a sequence of partial reductions $\{(T_i, A_i, f_i), y_i, p_i\}_{i=1}^m$ of (T, A, f), from (iv) it follows that $|A| = p_1 p_2 \cdots p_{m-1} |A_m^\circ|$. Moreover, since $\mathrm{Per}(f_i) \supset p_i \,\mathrm{Per}(f_{i+1}) \cup \{1\}$ for $1 \le i < m$, it follows that $\mathrm{Per}(f) \supset \{1, p_1, p_1 p_2, \dots, p_1 p_2 \cdots p_{m-1}\} \cup p_1 p_2 \dots p_{m-1} \,\mathrm{Per}(f_m)$. □

The next theorem and corollary are the main results of this section.

Theorem 1.6.7. *Each 2-orbital canonical model admits a sequence of partial reductions.*

Proof. Let (T, A, f) be a 2-orbital canonical model. During this proof, we will use the notation from the definition of a sequence of partial reductions. In particular, the roman numerals (i–vi) refer to the properties of that definition. We formally denote $\{(T_i, A_i, f_i), y_i, p_i\}_{i=1}^k$ by $\mathcal{S}^k$ for any $k \geq 0$ (note that $\mathcal{S}^0 = \emptyset$).

We start by setting $(T_1, A_1, f_1) = (T, A, f)$. Therefore (T_1, A_1, f_1) is a 2-orbital canonical model. Moreover, (i–v) hold (with 1 instead of m). Now we proceed by induction on k.

Let $k \geq 1$ and assume that we have constructed a sequence $\mathcal{S}^{k-1}$ and a canonical model (T_k, A_k, f_k) such that:

(a) A_k contains a unique periodic orbit of f_k and (T_k, A_k, f_k) is 2-orbital if $|A_k{}^\circ| > 1$.
(b) (i–v) hold (with k instead of m).

Observe that if, in addition, there exist y_k and p_k such that (vi) holds (with k instead of m) then $\mathcal{S}^k$ is a sequence of partial reductions.

Now we must define y_k and p_k and then decide whether $\mathcal{S}^k$ is a sequence of partial reductions (in this case we stop by setting $m = k$) or we construct a canonical model $(T_{k+1}, A_{k+1}, f_{k+1})$ such that $\mathcal{S}^k$ and $(T_{k+1}, A_{k+1}, f_{k+1})$ verify (a) and (b) (with $k+1$ instead of k).

Assume that $|A_k{}^\circ| = 1$. We set $p_k = 1$ and define y_k to be the unique element of $A_k{}^\circ$. Then $\mathcal{S}^k$ verifies (vi.1) and thus $\mathcal{S}^k$ is a sequence of partial reductions. In this case we are done by setting $m = k$.

Assume that $|A_k{}^\circ| > 1$. Then (T_k, A_k, f_k) is 2-orbital since (a) holds. By Proposition 1.5.4, there exists $y_k \in \operatorname{Fix}(f_k)$ such that (T_k, A_k, f_k) is y_k-expansive.

Let p be a type of $A_k^{y_k}$. If $\tilde{\mathcal{E}}(T_k, A_k, f_k) \neq \emptyset$ then we define $p_k = p$ and (vi.2) holds (with k instead of m). Hence $\mathcal{S}^k$ is a sequence of partial reductions and we are done by setting $m = k$.

From now on we assume that $\tilde{\mathcal{E}}(T_k, A_k, f_k) = \emptyset$. Since $|A_k{}^\circ| > 1$, the model (T_k, A_k, f_k) does not verify neither (vi.1) nor (vi.2) and $\mathcal{S}^k$ is not a sequence of partial reductions. In order to iterate the argument we will define a model $(T_{k+1}, A_{k+1}, f_{k+1})$ such that $\mathcal{S}^k$ and $(T_{k+1}, A_{k+1}, f_{k+1})$ verify (a) and (b) with $k+1$ instead of k. We consider two cases.

Case 1. *(T_k, A_k, f_k) is twist around y_k.*

We define $p_k = p$. By Proposition 1.6.4, there exists a y_k-branch T_{k+1} and a finite set $A_{k+1} \subset T_{k+1}$ such that if we define $f_{k+1} = (f_k)^{p_k}|_{T_{k+1}}$ then $(T_{k+1}, A_{k+1}, f_{k+1})$ is a canonical model and $\operatorname{Per}(f_k) \supset p_k \operatorname{Per}(f_{k+1})$. Hence, $(T_{k+1}, A_{k+1}, f_{k+1})$ is a partial p_k-reduction of (T_k, A_k, f_k). Furthermore, A_{k+1} contains a unique periodic orbit and $(T_{k+1}, A_{k+1}, f_{k+1})$ is 2-orbital if $|A_{k+1}{}^{\circ}| > 1$. Finally,

$$\begin{aligned} |A_k{}^{\circ}| = p_k|A_{k+1}{}^{\circ}| \text{ and} \\ \text{if } p_k = 1 \text{ then } A_k{}^{\circ} \subset A_{k+1} \subsetneq A_k. \end{aligned} \tag{1.11}$$

Summarizing, we have constructed a canonical model $(T_{k+1}, A_{k+1}, f_{k+1})$ in such a way that $\mathcal{S}^k$ and $(T_{k+1}, A_{k+1}, f_{k+1})$ verify (a) and (b) (with $k+1$ instead of k).

Case 2. (T_k, A_k, f_k) *is non-twist around* y_k.

Since $\tilde{\mathcal{E}}(T_k, A_k, f_k) = \emptyset$, by Proposition 1.6.5 there exists a tree $T_{k+1} \subset T_k$ and a finite set $A_{k+1} \subset T_{k+1}$ such that $(T_{k+1}, A_{k+1}, f_k|_{T_{k+1}})$ is a 2-orbital canonical model and

$$A_k{}^{\circ} \subset A_{k+1} \subsetneq A_k. \tag{1.12}$$

Then we set $p_k = 1$ and $f_{k+1} = f_k|_{T_{k+1}}$. Therefore, $\operatorname{Per}(f_k) \supset p_k \operatorname{Per}(f_{k+1})$ and $(T_{k+1}, A_{k+1}, f_{k+1})$ is a partial p_k-reduction of (T_k, A_k, f_k). As above, we have constructed a canonical model $(T_{k+1}, A_{k+1}, f_{k+1})$ such that $\mathcal{S}^k$ and $(T_{k+1}, A_{k+1}, f_{k+1})$ verify (a) and (b) (with $k+1$ instead of k).

Finally we must prove that this iterative construction stops after a finite number of steps. This is a direct consequence of (1.11), (1.12) and the finiteness of A_1. ■

Next we will use the notion of a sequence of partial reductions to estimate the set of periods of a canonical model. A serious drawback of this notion is that it is only defined for canonical models, whereas we are interested in studying the set of periods of the more general monotone models. However, by means of Theorem 1.5.3, for each monotone model (S, P, g) we can construct a canonical model (T, A, f) associated to it (see page 29). Then we can use a sequence of partial reductions to get an estimation of $\operatorname{Per}(f)$, which differs from $\operatorname{Per}(g)$ only in finitely many periods. This motivates the following definition.

Let (S, P, g) be non-trivial periodic monotone model. A pair $\{\mathcal{R}, K\}$, where $\mathcal{R}$ is a canonical model and $K \subset \mathbb{N}$, is said to be a *complete reduction of* (S, P, g) if there exists a sequence of partial reductions $\{(T_i, A_i, f_i), y_i, p_i\}_{i=1}^m$ with $K = \{1, p_1, p_1p_2, \dots, p_1p_2\cdots p_{m-1}\}$, $\mathcal{R} = (T_m, A_m, f_m)$ and (T_1, A_1, f_1) is a canonical model associated to (S, P, g). When $\mathcal{R}$ is non-trivial, we define the three non-negative numbers which play the central role in the characterization of $\operatorname{Per}(g)$ given by Theorem A. In this case, $p(\mathcal{R})$ will denote the type p_m of $A_m^{y_m}$, $q(\mathcal{R})$ will denote a rotation index of $\mathcal{R}$ associated to the type p_m, and $n(\mathcal{R})$ will denote the least n such that $\mathcal{R}$ is n-orbital. Observe that $n(\mathcal{R}) \in \{0, 1, 2\}$ since $\mathcal{R}$ is 2-orbital.

From this definition and Theorem 1.6.7 we obtain the following corollary.

Corollary 1.6.8. *Assume that (S, P, g) is a non-trivial periodic monotone model. Then (S, P, g) admits a complete reduction. For each complete reduction $\{(\overline{S}, \overline{P}, \overline{g}), K\}$ of (S, P, g), there exists a (possibly empty) finite set $\mathcal{V}$ such that*

$$\operatorname{Per}(g) \supset \mathcal{V} \cup K \cup (\max K) \operatorname{Per}(\overline{g})$$

and each element of $\mathcal{V}$ divides the least common multiple of the periods of all periodic orbits of g contained in $V(S)$. Moreover, $|P| = (\max K)|\overline{P}^{\circ}|$.

Proof. Since (S, P, g) is 0-orbital, by Theorem 1.5.3 there is a 0-orbital (and thus 2-orbital) canonical model (T, A, f) associated to (S, P, g). By Theorem 1.6.7, (T, A, f) admits a sequence $\{ (T_i, A_i, f_i),\ y_i, p_i \}_{i=1}^{m}$ of partial reductions. So $\{ (T_m, A_m, f_m),\ \{1, p_1, p_1p_2, \dots, p_1p_2 \cdots p_{m-1}\} \} = \{ (\overline{S}, \overline{P}, \overline{g}), K \}$ is a complete reduction of (S, P, g).

Since (T, A, f) and (S, P, g) are associated, A is a periodic orbit of f, $|A| = |P|$ and there exists a (possibly empty) finite set $\mathcal{V}$ verifying the prescribed properties and such that $\operatorname{Per}(g) = \operatorname{Per}(f) \cup \mathcal{V}$. By Remark 1.6.6, $\operatorname{Per}(f) \supset K \cup (\max K) \operatorname{Per}(\overline{g})$ and $|A| = (\max K)|\overline{P}^{\circ}|$. ■

1.7 Proof of Theorem A

The main results used in the proof of Theorem A are: Corollary 1.6.8, which allows us to work with a complete reduction instead of the original model, and both Lemma 1.4.6 and Theorem 1.4.7 which are used to calculate the set of periods of the reduced model.

Proof of Theorem A. By the definition of a monotone model, $\operatorname{En}(S) \subset P$. Therefore, S reduces to a point when P consists of a fixed point, and in this case the theorem follows obviously.

Assume that $|P| > 1$. The fact that there exist complete reductions of (S, P, g) follows from Corollary 1.6.8. Moreover, given a complete reduction $\{\mathcal{R}, K\}$, we have $\operatorname{Per}(g) \supset K$. If $\mathcal{R}$ is trivial, we are done.

Assume that $\mathcal{R}$ is non-trivial and set $\mathcal{R} = (\overline{S}, \overline{P}, \overline{g})$. By the definition of a complete reduction we have $\tilde{\mathcal{E}}(\mathcal{R}) \neq \emptyset$. Thus there exists $\beta \in \mathcal{E}(\mathcal{R})$ such that $|\beta| \in p\mathbb{N}$ and $|\beta| \leq |\overline{P}^{\circ}| + p + q + n + 1$. We define $\lambda = |\beta|/p$. Since β is external, by Lemma 1.4.6 we get $\operatorname{Per}(\overline{g}) \supset \{\lambda p i + p j : i, j \geq 1\}$. Moreover, by Lemma 1.4.2, $p \in \operatorname{Per}(\overline{g})$. Consequently, from Corollary 1.6.8 we have

$$\begin{aligned} \operatorname{Per}(g) &\supset K \cup \{kp\} \cup \{\lambda kpi + kpj : i, j \geq 1\} \\ &= K \cup kp\mathbb{N} \setminus \{2kp, 3kp, \dots, \lambda kp\}, \end{aligned} \tag{1.13}$$

and $|P| = k|\overline{P}^{\circ}|$. Hence, when $|\overline{P}^{\circ}| \in p\mathbb{N}$ it follows that for each $l \geq 0$ (see Remark 1.2.3) we have $|P| + lkp = k(|\overline{P}^{\circ}| + lp) \in kp\mathbb{N}$ and therefore $\mathcal{S}^*_{kp}(|P| + lkp) = S_{kp}(3kp) = \{1\} \cup kp\mathbb{N}$. Hence, from (1.13) we have

$$\mathrm{Per}(g) \supset K \cup \mathcal{S}^*_{kp}(|P| + lkp) \setminus \{2kp, 3kp, \ldots, \lambda kp\}$$

and the theorem follows in this case.

Assume now that $|\overline{P}^{\circ}| \notin p\mathbb{N}$. By Theorem 1.4.7 we have $\mathrm{Per}(\overline{g}) \supset \{(|\overline{P}^{\circ}| + lp)i + pj : i, j \geq 1\}$ for some $0 \leq l \leq |\overline{P}^{\circ}| + q + n - 1$. Then, $l \leq |P|/k + q + 1$ because $n \in \{0, 1, 2\}$. Furthermore, if $n = 0$ then, again by Theorem 1.4.7, $lp \leq p + q - (q \bmod p)$. Thus, from Corollary 1.6.8 it follows that

$$\mathrm{Per}(g) \supset \{(|P| + lkp)i + kpj : i, j \geq 1\}. \tag{1.14}$$

Since $|P| + lkp = |\overline{P}^{\circ}|k + lkp \notin kp\mathbb{N} \setminus \{1\}$, we have

$$\begin{aligned} \mathcal{S}^*_{kp}(|P| + lkp) &= S_{kp}(|P| + lkp) \\ &= \{1, |P| + lkp\} \cup \{(|P| + lkp)i + kpj : i \geq 0, j \geq 1\} \\ &= \{1\} \cup kp\mathbb{N} \cup \{(|P| + lkp)i + kpj : i, j \geq 1\}. \end{aligned}$$

Hence, from (1.13) and (1.14) we have

$$\mathrm{Per}(g) \supset K \cup \mathcal{S}^*_{kp}(|P| + lkp) \setminus \{2kp, 3kp, \ldots, \lambda kp\}.$$

■

1.8 Upper bounds for the type and the rotation index

This section is devoted to prove the inequalities (1.1), from Section 1.2. In the proof of Proposition 1.8.1 we use Lemma 1.4.4.

Proposition 1.8.1. *Let (T, A, f) be a y-expansive n-orbital model such that $\mathcal{E}(T, A, f) \neq \emptyset$. Let p be a type of A^y and let q be a rotation index of (T, A, f) associated to p. Then:*
(a) $p \leq |A^{\circ}| + 1$.
(b) If $q > 0$ *then* $2p + q - 2 \leq |A^{\circ}|$.

If, in addition, $n = 0$ *then:*
(c) $p \leq |A| - 1$.
(d) If $p = 1$ *then* $q + 4 \leq |A|$.

(e) If $q > 0$ then $2p + q + 1 \leq |A|$.

Proof. Until the end of the proof, the subindexes will be considered modulo p. Since p is a type of A^y we have $p \leq |\operatorname{En}(T)|$ and, since (T, A, f) is orbital, there is at most 1 endpoint which does not belong to A°. Therefore,

$$|\operatorname{En}(T)| \leq |A^\circ| + 1, \tag{1.15}$$

which proves (a). If $n = 0$ then $A = A^\circ$. Moreover, if $\operatorname{En}(T) = A$ then from the fact that (T, A, f) is a canonical model and the uniqueness of canonical models (see Theorem B of [3]) we get that T is a $|A|$-star whose central point is y and $f([y, x]) = [y, f(x)]$ for each $x \in A$. Then $\mathcal{E}(T, A, f) = \emptyset$, a contradiction. Consequently, $\operatorname{En}(T) \subsetneq A$ and

$$|\operatorname{En}(T)| \leq |A| - 1 \text{ when } n = 0, \tag{1.16}$$

which proves (c).

Next we prove (b). By assumption we have $q > 0$. So, $x_i \in V(T)$ for $1 \leq i \leq p$ and, hence, for each $1 \leq i \leq p$ there are at least 2 points of $X(A)$ in Z_i (we recall that Z_i stands for $Z(A^y)_i$). Therefore,

$$\Big|X(A) \cap \bigcup_{i=1}^{p} Z_i\Big| \geq 2p. \tag{1.17}$$

Let $k \in \{1, 2, \dots, p\}$ be such that $q = q_k$ and set $Q = \{f^i(x_k)\}_{i=0}^{q-1}$. Clearly $Q \subset V(T) \setminus A$. By Lemma 1.4.4, $f^i(x_k) \in Z_{k+i}$ for $0 \leq i \leq q - 1$. Moreover, since (T, A, f) is y-expansive we have that Q does not contain periodic orbits and thus $|Q| = q$.

We claim that $Q \cap \{x_i\}_{i=1}^{p} = \{x_k\}$. Indeed, assume that $f^j(x_k) \in \{x_i\}_{i=1}^{p}$ for some $1 \leq j \leq q - 1$. Then $f^j(x_k) = x_{k+j}$. Since Q does not contain periodic orbits, $x_{k+j} \neq x_k$ and then we easily get that $q_{k+j} = q_k - j$, in contradiction with the fact that $q_k = q = \min\{q_1, q_2, \dots, q_p\}$. Thus the claim follows.

Given $v \in Q \setminus \{x_k\}$, we have that $v \in V(T) \cap Z_i$ for some $1 \leq i \leq p$, $v \neq x_i$ and $[x_i, v] \cap A = \emptyset$. Since $v \in V(T)$, there exists some point $w \in X(A) \cap Z_i$ with $v \prec w$ which has not been taken into account in (1.17). Therefore,

$$\Big|X(A) \cap \bigcup_{i=1}^{p} Z_i\Big| \geq 2p + q - 1 \text{ when } q > 0. \tag{1.18}$$

Since $|\operatorname{En}(T)| \geq |X(A)|$, (b) follows from (1.15) and (1.18).

To prove (d), assume that $p = 1$. Let $a \in X(A) \cap Z_1$. Then $f(a) \neq a$ since (T, A, f) is orbital. Since $x_1 \prec f(x_1)$ and x_1 belongs to the $(A \cup \{y\})$-basic interval $[y, a]$, the $(A \cup \{y\})$-monotonicity of f implies that $f(a) \in Z_1$. By Remark 1.6.2, there is at least one y-branch different from Z_1. Therefore, since $\mathrm{En}(T) \subset A$ there exists $b \in X(A) \setminus Z_1$.

Now we claim that $f(b) \notin Z_1$. Indeed, if $f(b) \in Z_1$ then the fact that $(a, b) \cap A = \emptyset$ and the A-monotonicity of f imply that $f([a, b]) \subset Z_1$, in contradiction with the fact that $y \in [a, b]$ and hence $y \in f([a, b])$. Thus the claim follows.

Observe that $f(b) \neq b$ since (T, A, f) is orbital. Also, by the previous claim, $f(b) \notin \{a, f(a)\}$, and thus a, b, $f(a)$ and $f(b)$ are 4 different points contained in A. In consequence, $|A| \geq 4$. Then, (d) holds when $q = 0$. When $q > 0$ we have $q = q_1$ since $p = 1$. So, by Lemma 1.4.4, $f^i(x_1) \in (V(T) \cap Z_1) \setminus A$ for $0 \leq i < q$. Let S be the closure of the connected component of $Z_1 \setminus X(A)$ which contains x_1. Then S is a tree whose endpoints are the elements of $X(A) \cap Z_1$. From the definition of q it follows that, for $0 < i < q$, $f^i(x_1)$ are vertices of S which are not endpoints of S. Since any tree with n vertices has at least $n + 2$ endpoints, we get $|\mathrm{En}(S)| \geq q + 1$. As we noticed above, $f(a) \in Z_1$ and $f(a) \neq a$ for any $a \in X(A) \cap Z_1$. Thus $|A \cap Z_1| > |X(A) \cap Z_1|$ and, hence, $|A \cap Z_1| \geq q + 2$. Therefore, taking into account b and $f(b)$, which are in A but not in Z_1, we have $|A| \geq q + 4$ and (d) holds.

To end the proof of the proposition we must show that (e) holds. So we assume that $n = 0$, that is, A is a periodic orbit. By (d), it is enough to consider the case $p > 1$. Since $|\mathrm{En}(T)| \geq |X(A)|$, from (1.16) and (1.18) it follows that $|A| \geq 2p + q$. So, we must show that $|A| \neq 2p + q$.

In the rest of the proof we assume that $|A| = 2p + q$ (and $q > 0$ and $p > 1$) and we will arrive to a contradiction. If $A = X(A)$, as in the proof of (c) we get that f is a rigid rotation of a $|A|$-star and since $\mathcal{E}(T, A, f) \neq \emptyset$ we get that $X(A) \subsetneq A$. From (1.18) we have

$$2p + q - 1 \leq \Big|X(A) \cap \bigcup_{i=1}^{p} Z_i\Big| \leq |X(A)| < |A| = 2p + q.$$

Hence, there is exactly one point w in $A \setminus X(A)$ and $|X(A) \cap \cup_{i=1}^{p} Z_i| = 2p + q - 1$.

Now we claim that $n^\star = p$. Indeed, assume that there exists some y-branch W different from $Z_1, Z_2, \ldots, Z_p$. Since $\mathrm{En}(T) \subset A$, we have $W \cap A \neq \emptyset$. Since $|X(A) \cap \cup_{i=1}^{p} Z_i| = 2p + q - 1$ and $|A| = 2p + q$, it follows that $W \cap A = \{w\}$. Thus $w \in X(A)$, a contradiction. So the claim follows.

Let $j \in \{1, 2, \ldots, p\}$ be such that $w \in Z_j$. We have that $Z_j \cap A$ is the disjoint union of $\{w\}$ and $Z_j \cap X(A)$, while $Z_i \cap A = Z_i \cap X(A)$ when $i \neq j$.

Since $q > 0$, $x_i \notin X(A)$ for $1 \le i \le p$. For each point $z \in X(A) \cap Z_i$, we have that $[y, z]$ is a $(A \cup \{y\})$-basic interval containing x_i. Therefore, since $f(x_i) \in Z_{i+1}$ and f is $(A \cup \{y\})$-monotone, we get $f(X(A) \cap Z_i) \subset Z_{i+1}$ for $1 \le i \le p$. Moreover, $f(X(A) \cap Z_i) = X(A) \cap Z_{i+1}$ when $i \not\equiv j$ and $i \not\equiv j + 1 \,(\operatorname{mod} p)$. We set $N_i = |X(A) \cap Z_i|$ for $1 \le i \le p$. Since $X(A) \subset A$ and A is a periodic orbit, it follows that $N_{i+1} = N_i$ when $i \not\equiv j$ and $i \not\equiv j + 1 \,(\operatorname{mod} p)$. Consequently, we have

$$N_{j-1} = N_{j-2} = \ldots = N_{j+2} = N_{j+1}. \tag{1.19}$$

If $f(w) \in Z_{j+1}$ then (T, A, f) is twist around y and, by Remark 1.6.1, $\mathcal{E}(T, A, f) = \emptyset$, a contradiction. Therefore, $f(w) \notin Z_{j+1}$. Then $X(A) \cap Z_{j+1} = f(X(A) \cap Z_j)$ and it follows that $N_{j+1} = N_j$. Thus from (1.19) we get that $N_{j-1} = N_j$. On the other hand, since A is a periodic orbit, there exists a unique point $w' \in A \cap Z_{j-1} = X(A) \cap Z_{j-1}$ such that $f(w') = w$. Since $w \in Z_j \setminus X(A)$, we get $N_j = N_{j-1} - 1$, a contradiction. ■

The following result states that the inequalities (1.1) hold for complete reductions of monotone models.

Corollary 1.8.2. *Let (S, P, g) be a non-trivial periodic monotone model. Let $\{\mathcal{R}, K\}$ be a complete reduction of (S, P, g) such that $\mathcal{R}$ is non-trivial and $n(\mathcal{R}) = 0$. Then*

$$\begin{gathered} p \le r - 1, \\ q + 4 \le r \ \text{ when } p = 1 \ \text{ and} \\ 2p + q + 1 \le r \ \text{ when } q > 0, \end{gathered}$$

where we denote $p(\mathcal{R})$, $q(\mathcal{R})$, $n(\mathcal{R})$ and $\frac{|P|}{\max K}$ by p, q, n and r respectively.

Proof. Set $(T, A, f) = \mathcal{R}$. Since $n = 0$, we have $A^\circ = A$. By Corollary 1.6.8, $|A| = r$. Since $\mathcal{R}$ is non-trivial, by the definition of a complete reduction we have that $\mathcal{R}$ is a y-expansive 0-orbital model for some $y \in \operatorname{Fix}(f)$, p is a type of A^y, q is a rotation index of $\mathcal{R}$ associated to p and $\tilde{\mathcal{E}}(\mathcal{R}) \neq \emptyset$. In particular, $\mathcal{E}(\mathcal{R}) \neq \emptyset$. Therefore $\mathcal{R}$ verifies the hypotheses of Proposition 1.8.1 and the corollary follows. ■

1.9 Some Examples. Proof of Theorem B

This section is devoted to prove Theorem B. In fact, we prove the following stronger result from which Theorem B can be obviously derived. Since any tree can be imbedded in $\mathbb{R}^2$, in what follows we will consider each tree endowed with the topology induced by the topology of $\mathbb{R}^2$.

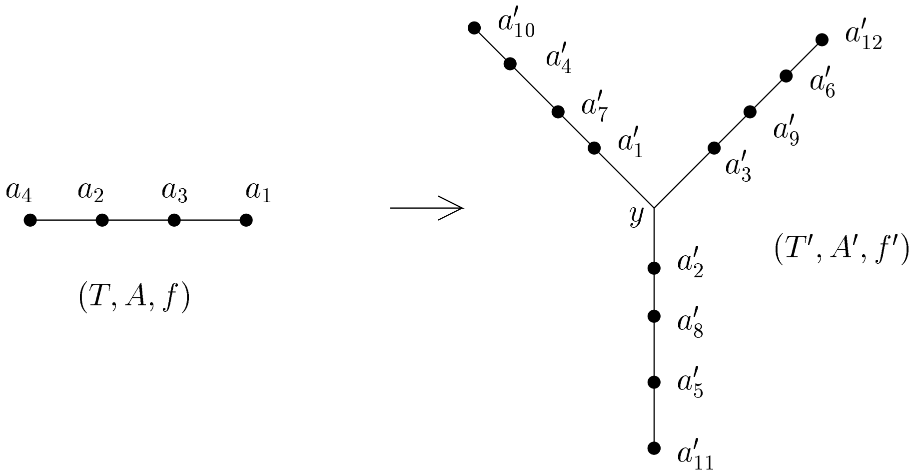

Figure 1.3: An example of the construction made in Proposition 1.9.2, with $|A| = 4$ and $s = 3$. We assume $f(a_i) = a_{i+1 \bmod 4}$ and $f'(a'_i) = a'_{i+1 \bmod 12}$.

Theorem 1.9.1. *Let $K \subset \mathbb{N}$ be a set of the form $\{1, k_1, k_2, \ldots, k_m\}$ such that $k_1 > 1$ and k_i strictly divides k_{i+1} for $1 \leq i < m$. Set $k = k_m$. Then:*

(a) There exists a canonical model (R, B, h) with $|B| = k$ and $\operatorname{Per}(h) = K$.

(b) Let $r > 1$, $p \geq 1$ and $q \geq 0$ verifying

$$\begin{gathered} p \leq r - 1, \\ q + 4 \leq r \ \text{ when } p = 1 \ \text{ and} \\ 2p + q + 1 \leq r \ \text{ when } q > 0. \end{gathered}$$

Then there exist a canonical model (S, P, g) and a complete reduction $\{(\overline{S}, \overline{P}, \overline{g}), K\}$ of (S, P, g) with $|\overline{P}^{\circ}| = r$, $p(\overline{S}, \overline{P}, \overline{g}) = p$, $q(\overline{S}, \overline{P}, \overline{g}) = q$, $n(\overline{S}, \overline{P}, \overline{g}) = 0$ and $\operatorname{Per}(g) = K \cup \mathcal{C}$, where $\mathcal{C}$ is a set such that

$$\mathcal{S}^*_{kp}(|P| + lkp) \setminus \{2kp, 3kp, \ldots, \lambda kp\} \subset \mathcal{C} \subset \mathcal{S}^*_{kp}(|P|)$$

with $lp = p + q - (q \bmod p)$ and λp being the largest multiple of p smaller than $r + p + q + 1$.

In order to prove Theorem 1.9.1 we will use the following two technical results. For Proposition 1.9.2 see Figure 1.3, which shows an example of the construction made in that proposition.

Proposition 1.9.2. *Let (T, A, f) be a periodic canonical model and let $s \geq 2$ be an integer. Then there exists a canonical model (T', A', f') such that:*

(a) There exists $y \in \operatorname{Fix}(f')$ such that s is a type of A'^y around y and (T', A', f') is y-expansive and twist around y.

(b) A' *is a periodic orbit with* $|A'| = s|A|$.

(c) (T, A, f) *is a partial* s*-reduction of* (T', A', f').

(d) $\operatorname{Per}(f') = \{1\} \cup s \cdot \operatorname{Per}(f)$.

Proof. Set $t = |A \cup V(T)|$ and $Q^1 = A \cup V(T) = \{v_1^1, v_2^1, \ldots, v_t^1\}$ in such a way that $v_1^1 \in \operatorname{En}(T)$ and $A = \{v_i^1\}_{i=1}^{|A|}$. Next we will construct T' by attaching one copy of T to each endpoint of an s-star.

For $2 \le i \le s$ we consider a tree T^i, a finite set $Q^i = \{v_1^i, v_2^i, \ldots, v_t^i\} \subset T^i$ and a homeomorphism $h_i : T^i \longrightarrow T$ such that $h_i(v_j^i) = v_j^1$ for each $1 \le j \le t$. We also set $T^1 = T$ and $h_1 = \operatorname{Id}|_{T^1}$. Now we define T' to be a tree which consists of the union of T^i for $1 \le i \le s$ and an s-star R such that $\operatorname{En}(R) = \{v_1^1, v_1^2, \ldots, v_1^s\}$. Thus T' consists of an s-star with one copy of T attached to each endpoint. Let y be the central point of R.

Now we are going to define the map f'. First we define it on each T^i. We set $f'|_{T^i} = h_{i+1}^{-1} \circ h_i$ for each $1 \le i < s$ and $f'|_{T^s} = f \circ h_s$. Note that f' is a homeomorphism between T^i and T^{i+1} and $f'(v_1^i) = v_1^{i+1}$ for each $1 \le i < s$. Moreover, $f'(v_1^s) = f(v_1^1) \in A \cap T^1$.

Now we define f' on $R \setminus \operatorname{En}(R)$. We set $f'(y) = y$ and take f' to be an affine homeomorphism between $[y, v_1^i]$ and $[y, f(v_1^i)]$ for each $1 \le i \le s$.

Finally set $A' = \cup_{i=1}^s \{v_1^i, v_2^i, \ldots, v_{|A|}^i\}$. Since $f'|_{T^i}$ is a homeomorphism for each $1 \le i < s$ and f is A-monotone, we easily get that (T', A', f') is a monotone model. Moreover, since (T, A, f) is canonical, there are no f-identifiable vertices in T. Then there are no f'-identifiable vertices in T' and (T', A', f') is a canonical model.

Now we prove (a). Set $Q = \{y\} \cup_{i=1}^s Q^i$. Clearly $A'^y = Q$ and for $1 \le i \le s$ we have $Z(Q)_i = T^i$ and $x(Q)_i = z_i$. Since $f'(x(Q)_i) = x(Q)_{i+1}$ for $1 \le i < s$ and $f'(x(Q)_s) \in Z(Q)_1$, it follows that s is a type of A'^y. Moreover, $f'(Z(Q)_i) \subset Z(Q)_{i+1 \bmod s}$ for $1 \le i \le s$ and so (S', P', g') is twist around y. Since there are no vertices of T' in $Z^\star(A') \setminus \{y\}$, obviously (T', A', f') is y-expansive and (a) holds.

To prove the rest of the statements first we consider the case $|A| = 1$. Then T reduces to the unique point of A, that is v_1^1. Moreover, T' coincides with R and $A' = \{v_1^1, v_1^2, \ldots, v_1^s\}$. Therefore f' is a rigid rotation of an s-star and (b), (c) and (d) follows obviously in this case.

Now we assume that $|A| > 1$. We claim that

$$f'^s(z) = f(z) \text{ for each } z \in T. \tag{1.20}$$

Indeed, take $z \in T$. We have that $f'^{s-1}(z) = h_s^{-1} \circ h_{s-1} \circ h_{s-1}^{-1} \circ h_{s-2} \circ \ldots \circ h_2 \circ h_2^{-1} \circ h_1(z)$. Since $h_1 = \operatorname{Id}$, we get $f'^{s-1}(z) = h_s^{-1}(z) \in T^s$. Therefore $f'^s(z) = f'(f'^{s-1}(z)) = f(h_s(h_s^{-1}(z))) = f(z)$ and the claim follows.

Now we prove (b) and (c). Let $x \in T$ be an n-periodic point of f. Since $f'^i(x) \in T^{i+1}$ for $1 \le i < s$, from (1.20) it follows that $\{f'^i(x)\}_{i=0}^{sn}$ is an sn-periodic orbit of f'. In particular, (b) holds. Thus we have $\{1\} \cup s\,\mathrm{Per}(f) \subset \mathrm{Per}(f')$. This inclusion, together with (1.20) and the fact that $T \subset T'$, proves that (T, A, f) is a partial s-reduction of (T', A', f') and (c) holds.

Finally we prove (d). It is enough to show that $\mathrm{Per}(f') \subset \{1\} \cup s\,\mathrm{Per}(f)$. Let P be an n-periodic orbit of f' with $n > 1$. The definition of f' on R implies that y is a repelling fixed point of f'^s on each edge of R. It follows that the unique periodic orbit of f' on R is $\{y\}$. Therefore, $P \subset T' \setminus R$. Moreover, since (T', A', f') is twist we have that $n = rs$ for some $r \ge 1$ and there exists $x \in P \cap T$ such that $f'^i(x) \in T^{i+1 \bmod s}$ for all $1 \le i \le rs$. From (1.20) we get that $\{f'^{is}(x)\}_{i=0}^{r-1}$ is an r-periodic orbit of f. Thus $n \in s\,\mathrm{Per}(f)$. ■

By convention, a tree T will be a 1-*star* if T reduces to a single point, which in addition will be called the *central point* of T.

In the next proposition we will construct canonical models which exhibit prescribed sets of periods. It is the main tool for the proof of Theorem 1.9.1.

Proposition 1.9.3. *Given integers $r > 1$, $p \ge 1$ and $q \ge 0$ verifying*

$$\begin{gathered} p \le r-1, \\ q+4 \le r \ \textit{ when } \ p = 1 \ \textit{ and} \\ 2p+q+1 \le r \ \textit{ when } \ q > 0, \end{gathered}$$

there exists a monotone model (T, A, f) satisfying:

(a) (T, A, f) is a canonical model and A is a periodic orbit with $|A| = r$.

(b) There exists $y \in \mathrm{Fix}(f)$ such that (T, A, f) is y-expansive and non-twist around y.

(c) p is a type of A^y around y and q is a rotation index of (T, A, f) associated to p.

(d) $\mathcal{S}_p^*(r+lp) \setminus \{2p, 3p, \dots, \lambda p\} \subset \mathrm{Per}(f) \subset \mathcal{S}_p^*(r)$ with $lp = p + q - (q \bmod p)$ and λp being the largest multiple of p smaller than $r + p + q + 1$.

A canonical model which satisfies the properties (a–d) stated in Proposition 1.9.3 will be called an (r, p, q)-*model*.

Proof of Proposition 1.9.3. We will make the construction of (T, A, f) according to four cases. In all cases, we set $A = \{a_1, a_2, \dots, a_r\}$ and $f(a_i) = a_{i+1 \bmod r}$ for $i = 1, 2, \dots, r$. Thus A is a periodic orbit and $|A| = r$. In each case the construction will consist of two steps. First we will choose T to be a tree whose endpoints are contained in A. We will describe T by enumerating all its vertices and edges. Secondly, we will choose a point $y \in T$ to be a fixed point of f and will define f on each vertex of T.

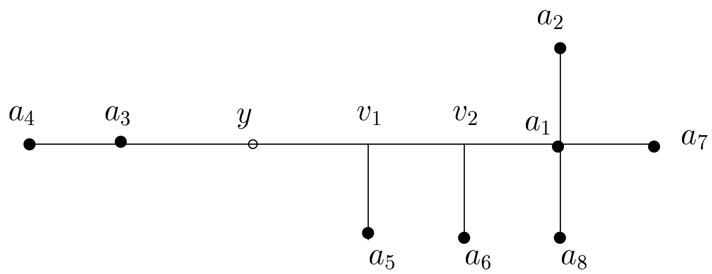

Figure 1.4: A (8,1,2)-model

In all cases, it is assumed that we define f on the whole tree by taking an A^y-extension (recall that $A^y = A \cup V(T) \cup \{y\}$). It is not difficult to check that the images of the vertices of T will be chosen in such a way that f is monotone on each A-basic interval. Therefore, by construction f is A-monotone and (T, A, f) is a monotone model.

Case 1. $p = 1$.

Definition of T. Assume first that $q = 0$. By assumption, $r \geq 4$. If $r = 4$ we take T as a closed interval $[a_4, a_2]$ with $a_4 < a_3 < a_1 < a_2$. If $r > 4$ then we choose T to be a closed interval $[a_4, a_r]$ with $a_4 < a_3 < a_1 < a_2 < a_5 < a_6 \ldots < a_r$. We also choose a point $y \in (a_3, a_1)$.

Now assume that $q > 0$. By assumption, $r \geq q + 4$. We choose T and a point $y \in T$ such that there are two y-branches and:

(i) $\mathrm{Cl}(Z^\star(A))$ has $q + 2$ endpoints and contains the edges: $[y, a_3]$, $[y, v_1]$, $[v_i, v_{i+1}]$ for $1 \leq i \leq q - 1$, $[v_q, a_1]$ and $[v_j, a_{4+j}]$ for $1 \leq j \leq q$.

(ii) $Z(A)_1$ is a star containing the edges $[a_1, a_2]$ and $[a_1, a_i]$ for $q+5 \leq i \leq r$.

(iii) $Z(A)_2 = [a_3, a_4]$.

See Figure 1.4 for an example with $r = 8$ and $q = 2$.

Images of the vertices. We define $f(y) = y$. Since $V(T) \setminus A = \emptyset$ when $q = 0$, we only have to consider the case $q > 0$. We define $f(v_i) = v_{i+1}$ for $1 \leq i < q$ and $f(v_q) = a_1$.

(a) In both cases, the orbit of each vertex of $T \setminus \{y\}$ intersects A. Therefore, there are no f-identifiable vertices in T and (T, A, f) is a canonical model.

(b) There exists $y \in \mathrm{Fix}(f) \cap (a_3, a_1)$ and a y-branch S such that $a_1, a_2 \in S$ and $a_3 \notin S$. Therefore $f(a_1) \in S$, $f(a_2) \notin S$ and (T, A, f) is non-twist around

y. Moreover, there are no periodic orbits of vertices contained in $Z^\star(A)$ and thus (T, A, f) is y-expansive.

(c) In the case $q = 0$, we have $x(A^y)_1 = a_1$. When $q > 0$, we have $x(A^y)_1 = v_1$. In both cases, $f(x(A^y)_1) \succ x(A^y)_1$ and thus 1 is a type of A^y. Moreover, since $f(v_i) = v_{i+1}$ for $1 \le i < q$ and $f(v_q) = a_1$, it follows that q is a rotation index of the type 1.

(d) Assume first that $q = 0$. In this case, the f-graph of A^y contains the loop $[y, a_1] \to [a_1, a_2] \to [y, a_1]$, which is external. Thus from Lemma 1.4.6 we get $\mathrm{Per}(f) \supset \mathbb{N} \setminus \{2\}$. Moreover, the loop $[a_1, a_2] \to [y, a_1] \to [a_1, a_2]$ is simple and a_1 and a_2 are not fixed points of f^2. Thus, by Lemma 1.3.5(a), f has a point of period 2. Therefore, $\mathrm{Per}(f) = \mathbb{N} = \mathcal{S}_p^*(r)$ and (d) holds.

Assume now that $q > 0$. The f-graph of A^y contains the loops $[y, v_1] \to [y, v_1]$, which we take as the typical loop α, and

$$[v_i, v_{i+1}] \to [v_{i+1}, v_{i+2}] \to \ldots \to [v_q, a_1] \to [a_1, a_2] \to [v_i, v_{i+1}],$$

which we call β_i for $0 \le i \le q$ (where v_0 stands for y). Note that β_i is negative and $|\beta_i| = q + 2 - i$ for each $0 \le i \le q$. Since $y \in \mathrm{Fix}(f)$ and each β_i satisfies (b) of Lemma 1.3.5, we get $\mathrm{Per}(f) \supset \{1, 2, 3, \ldots, q + 2\}$. Moreover, β_0 is an external loop of length $q + 2$ and thus, from Lemma 1.4.6, we get $\mathrm{Per}(f) \supset \{q + 3, q + 4, \ldots\}$. Therefore, $\mathrm{Per}(f) = \mathbb{N} = \mathcal{S}_p^*(r)$ and (d) holds.

Case 2. $p > 1$ *and* $q = 0$.

Definition of T. Since $p < r$, we can write $r = sp + k$ for some $s \ge 1$ and $0 < k \le p$. We choose T and a point $y \in T$ such that there are p y-branches and:

(i) $\mathrm{Cl}(Z^\star(A))$ is a $(p + k - 1)$-star whose endpoints are $a_1, a_2, \ldots, a_p$ and $a_{sp+2}, a_{sp+3}, \ldots, a_{sp+k}$ and whose central point is y.

(ii) $Z(A)_1$ is the union of $(a_1, a_{sp+1}]$ and an s-star whose endpoints are $a_1, a_{1+p}, \ldots, a_{1+(s-1)p}$.

(iii) For each $2 \le i \le s$, $Z(A)_i$ is an s-star whose endpoints are $a_i, a_{i+p}, \ldots,$ $a_{i+(s-1)p}$.

Observe that, when $s = 1$, $Z(A)_i$ reduces to the point a_i for each $2 \le i \le p$. When $s > 1$, we denote the central point of $Z(A)_i$ by y_i, and the central point of $Z(A)_1 \setminus (a_1, a_{sp+1}]$ by y_1. See Figure 1.5 for an example with $r = 14$ and $p = 4$.

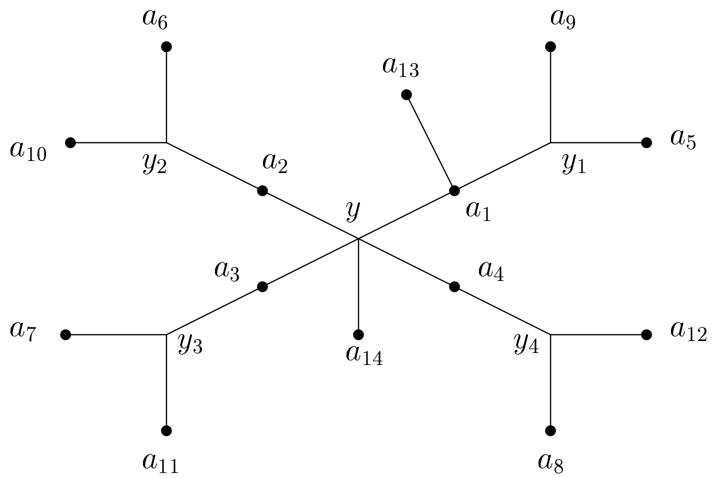

Figure 1.5: A (14,4,0)-model

Images of the vertices. We define $f(y) = y$ and $f(y_i) = y_{i+1 \bmod p}$ for each $1 \leq i \leq p$.

(a) Since there are no A^y-basic intervals with both endpoints contained in $V(T) \setminus A$, there are no f-identifiable vertices and thus (T, A, f) is a canonical model.

(b) Since a_1 and a_{sp+1} belong to $Z(A)_1$ and $Z^\star(A) \cap [f(a_1), f(a_{sp+1})] \neq \emptyset$, it follows that (T, A, f) is non-twist around y. Moreover, (T, A, f) is obviously y-expansive since there are no vertices of T in $Z^\star(A) \setminus \{y\}$.

(c) Observe that $(y, a_i) \cap A^y = \emptyset$ for $1 \leq i \leq p$. Therefore, $\{a_i\}_{i=1}^p = X(A^y)$. Since $f(a_p) = a_{p+1} \succ a_1$, it follows that p is a type of A^y and the rotation index of (T, A, f) associated to this type is 0.

(d) We set $I_i = [y, a_i]$ and $K_i = [a_i, y_i]$ for $1 \leq i \leq p$ (recall that K_i reduces to a single point when $s = 1$). We also set $I_i = [y_{i \bmod p}, a_i]$ for $p+1 \leq i \leq sp$, $I_j = [y, a_j]$ for $sp + 2 \leq j \leq r$ and $I_{sp+1} = [a_1, a_{sp+1}]$. All these intervals are A^y-basic intervals. Moreover, the f-graph of A^y contains exactly the following paths:

$$I_1 \to I_2 \to \ldots \to I_p \to I_1,$$
$$I_1 \to I_2 \to \ldots \to I_r \to I_1,$$
$$I_2 \to I_3 \to \ldots \to I_{ps+1} \to I_2,$$

which we call α, β and γ, respectively, and

$$K_1 \to K_2 \to \ldots \to K_p \to I_{p+1}, \; I_p \to K_1, \; I_{ps} \to K_1.$$

We take α as the typical loop. Hence, β and γ are external loops. Since $|\beta| = r$ and $|\gamma| = sp$, from Lemma 1.4.6 it follows that $\mathrm{Per}(f) \supset \{ir + jp : i, j \geq 1\} \cup \{isp + jp : i, j \geq 1\}$. On the other hand, γ is negative. Thus it verifies (b) of Lemma 1.3.5 and, consequently, $sp \in \mathrm{Per}(f)$. Moreover, $p \in \mathrm{Per}(f)$ by Lemma 1.4.2. Summarizing, we have

$$\mathrm{Per}(f) \supset \{ir + jp : i, j \geq 1\} \cup p\mathbb{N} \setminus \{2p, 3p, \ldots, (s-1)p\}.$$

It is not difficult to check that there are no loops of other lengths. Since, by Lemma 1.3.2, each periodic orbit which does not intersect A^y is associated to a loop in the f-graph of A^y, it follows that

$$\mathrm{Per}(f) = \{1, r\} \cup \{ir + jp : i, j \geq 1\} \cup p\mathbb{N} \setminus \{2p, 3p, \ldots, (s-1)p\}. \quad (1.21)$$

Since $q = 0$, by assumption $l = 0$ and λp is the largest multiple of p smaller than $r + p + 1$. Since $(s-1)p < r < r + p + 1$, it follows that $(s-1)p \leq \lambda p$. When $r \notin p\mathbb{N}$, from (1.21) we get

$$\mathrm{Per}(f) = \mathcal{S}_p^*(r) \setminus \{2p, 3p, \ldots, (s-1)p\} \quad (1.22)$$

and thus (d) holds. On the other hand, if $r \in p\mathbb{N}$ then $\{ir + jp : i, j \geq 1\} = \{m \in p\mathbb{N} : m > r\} \subset \{m \in p\mathbb{N} : m > (s-1)p\}$. Therefore (1.22) holds and (d) follows.

Case 3. $p > 1$, $q > 0$ *and* $r \notin p\mathbb{N}$.

Definition of T. By assumption, $r \geq 2p + q + 1$. Set $r = 2p + q + sp + k$ with $0 \leq k < p$. For simplicity, we assume $s > 0$. The same construction (even simpler) can be done in the case $s = 0$. The details are left to the reader (see Figure 1.6 for an example with $r = 11$, $p = 3$ and $q = 4$, for which $s = 0$). Under this assumption, we choose T and a point $y \in T$ such that there are p y-branches and:

(i) $\mathrm{Cl}(Z^\star(A))$ has $2p + q - 1$ endpoints.

(ii) For $1 \leq i \leq p$, $Z(A)_i$ contains an $(s+1)$-star whose endpoints are $a_i, a_{i+p}, \ldots, a_{i+sp}$. When $s > 1$, we denote the central point of this star by y_i.

(iii) $Z^\star(A) \setminus \{y\}$ contains $p+q-1$ vertices of T, that we call $v_1, v_2, \ldots, v_{p+q-1}$.

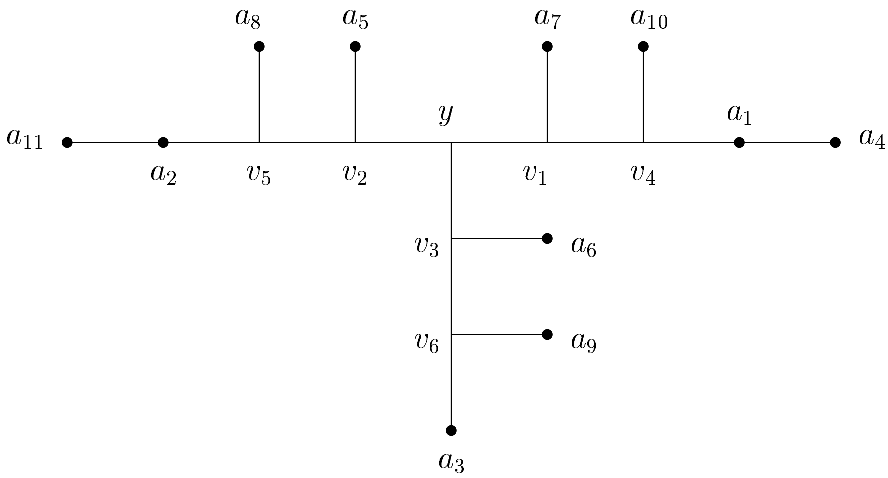

Figure 1.6: An (11,3,4)-model

(iv) Set $t = q \bmod p$. Now we define $r + p + q - t$ A^y-basic intervals, which are also edges of T:

$$I_i = [y, v_i] \text{ for } 1 \le i \le p,$$

$$I_{p+i} = [v_i, v_{p+i}] \text{ for } 1 \le i \le q - 1,$$

$$I_{p+q+i} = [v_{q+i}, a_{t+i}] \text{ for } 0 \le i \le p - t,$$

$$I_{2p+q-t+i} = [y_{i \bmod p}, a_{p+i}] \text{ for } 1 \le i \le sp,$$

$$I_{2p+q-t+sp+i} = [v_i, a_{p+sp+i}] \text{ for } 1 \le i \le p + q - 1 \text{ and}$$

$$I_{3p+2q-t+sp+i} = [a_{2p+sp+q+i}, a_{t+i}] \text{ for } 0 \le i \le k.$$

Finally, T contains also the following edges: $K_i = [a_i, y_i]$ for $1 \le i \le p$ (recall that K_i is not defined when $s < 2$) and $[v_i, a_{i \bmod p}]$ for $p + q - t + 1 \le i \le p + q - 1$. See Figure 1.7 for an example with $r = 19$, $p = 3$ and $q = 2$.

Images of the vertices. We define $f(y) = y$, $f(v_i) = v_{i+1}$ for each $1 \le i < p + q - 1$, $f(v_{p+q-1}) = a_t$ and $f(y_i) = y_{i+1 \bmod p}$ for each $1 \le i \le p$.

(a) There is a unique periodic orbit $P = \{y_i\}_{i=1}^{p}$ of vertices in $T \setminus \{y\}$. Moreover, for each A^y-basic interval $[v, v']$ such that $v \in P$ we have that $v' \in A$. Thus there are no f-identifiable vertices and (T, A, f) is a canonical model.

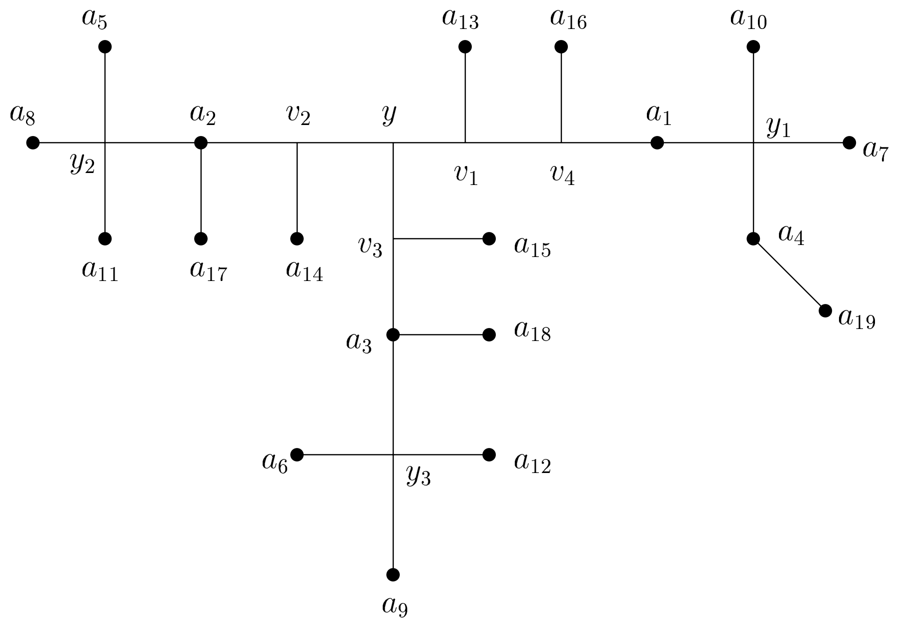

Figure 1.7: A (19,3,2)-model

(b) Observe that $a_i \in Z(A^y)_{i \bmod p}$ for every $1 \le i \le r$. In particular, the edge $I_{r+p+q-t} = [a_r, a_{t+k}]$ is contained in the y-branch $Z(A^y)_{r \bmod p}$, which is different from Z_p since, by assumption, $r \notin p\mathbb{N}$. Then $f(a_{t+k}) \in Z(A^y)_{r+1 \bmod p} \neq Z(A^y)_1$ and $f(a_r) = a_1 \in Z(A^y)_1$. Consequently, (T, A, f) is non-twist around y. Moreover, since the orbit of each vertex of $Z^\star(A) \setminus \{y\}$ contains $a_t \in A$, it follows that (T, A, f) is y-expansive.

(c) Since $x(A^y)_i = v_i$ and $f(v_i) = v_{i+1}$ for $1 \le i \le p$, we get that p is a type of A^y. Furthermore, $\min\{q_1, q_2, \ldots, q_p\} = q_p$. Since $[y, f^i(v_p)] \cap A = [y, v_{p+i}] \cap A = \emptyset$ for $1 \le i \le q-1$ and $[y, f(v_{p+q-1})] \cap A = \{a_t\}$, it follows that q is the rotation index of the type p.

(d) The f-graph of A^y contains exactly the following elementary loops:

(i) $I_1 \to I_2 \to \ldots \to I_p \to I_1$, which we denote by α and we take as the typical loop.

(ii) $I_i \to I_{i+1} \to \ldots \to I_{r+p+q-t} \to I_i$ for each $1 \le i \le 2p+q-t$ such that $i \equiv 1 \,(\mathrm{mod}\, p)$, which we denote by β_i. Each β_i is positive and $|\beta_i| = r+p+q-t-i+1 \in r + p\mathbb{N} \cup \{r\}$. Note that β_i is external only when $i = 1$.

(iii) $I_i \to I_{i+1} \to \ldots \to I_{r+p+q-t} \to I_i$ for each $1 \le i < 3p+q-t$ such that $i \equiv r+1 \,(\mathrm{mod}\, p)$, which we denote by γ_i. Each γ_i is negative and $|\gamma_i| = r + p + q - t - i + 1 \in p\mathbb{N}$. The maximum of the lengths of γ_i is $|\gamma_{r+1 \bmod p}| = r + p + q - t - (r \bmod p)$ and the minimum is $r - (r \bmod p) - p$. Note that γ_i is external only when $i = r + 1 \bmod p$.

(iv) $K_1 \to K_2 \to \ldots \to K_p \to I_{2p+q-t+1} \to \ldots \to I_{2p+q-t+sp} \to K_1$, which we denote by δ. Note that δ is negative and $|\delta| = (s+1)p$.

(v) $K_{r+1 \bmod p} \to K_{r+1 \bmod p+1} \to \ldots \to K_p \to I_{2p+q-t+1} \to \ldots \to I_{r+p+q-t} \to K_{r+1 \bmod p}$, which we denote by ϵ. Note that ϵ is positive and $|\epsilon| = r + 1 - (r + 1 \bmod p) \in p\mathbb{N}$.

(vi) $I_i \to I_{i+1} \to \ldots \to I_{2p+q-t+sp} \to I_i$ for each $p + 1 \leq i < 2p + q - t$ such that $i \equiv 1 \,(\mathrm{mod} p)$, which we denote by σ_i. Note that σ_i is negative and $|\sigma_i| = 2p + q - t + sp - i + 1 \in p\mathbb{N}$.

(vii) $I_{2p+q-t} \to K_1 \to K_2 \to \ldots \to K_p \to I_{2p+q-t+1} \to \ldots \to I_{r+p+q-t} \to I_1 \to \ldots \to I_{2p+q-t}$, which we denote by τ. Note that τ is negative and $|\tau| = r + 2p + q - t \in r + p\mathbb{N}$.

Since $p + q - t = lp$, we have $|\beta_1| = r + lp$. By Lemma 1.3.3, there is a fixed point $x \in I_1 = [y, v_1]$ of f^{r+lp} associated to β_1. Since v_1 is not periodic, $x \neq v_1$. Moreover, $x \neq y$ since $y \in \mathrm{Fix}(f)$ and some iterate of x belongs to I_{r+lp}, which does not contain y. Therefore, $x \in (y, v_1)$. Since β_1 is simple, from Lemma 1.3.5 it follows that the period of x is $r + lp$. On the other hand, $y \in \mathrm{Fix}(f)$ and $p \in \mathrm{Per}(f)$ by Lemma 1.4.2. Therefore,

$$\mathrm{Per}(f) \supset \{1, p, r + lp\}. \tag{1.23}$$

Since β_1 and $\gamma_{r+1 \bmod p}$ are external, by Lemma 1.4.6 we get $\mathrm{Per}(g) \supset \{(r + lp)i + pj : i, j \geq 1\} \cup \{(r - (r \bmod p) + lp)i + pj : i, j \geq 1\}$. Furthermore, since each γ_i is simple and negative it satisfies (b) of Lemma 1.3.5 and, hence, f has periodic orbits of periods $\{r - (r \bmod p) - p, r - (r \bmod p), \ldots, r - (r \bmod p) + lp\}$. So we have

$$\mathrm{Per}(f) \supset \{(r + lp)i + pj : i, j \geq 1\} \cup p\mathbb{N} \setminus \{2p, 3p, \ldots, r - (r \bmod p) - 2p\}.$$

From this and (1.23) it follows that

$$\mathrm{Per}(f) \supset \mathcal{S}_p^*(r + lp) \setminus \{2p, 3p, \ldots, r - (r \bmod p) - 2p\}. \tag{1.24}$$

Since λp is defined to be the larger multiple of p smaller than $r + p + q + 1$, we have $\lambda p > r - (r \bmod p) - 2p$ and from (1.24) we get

$$\mathrm{Per}(f) \supset \mathcal{S}_p^*(r + lp) \setminus \{2p, 3p, \ldots, \lambda p\}.$$

To end the proof we must show that $\mathrm{Per}(f) \subset \mathcal{S}_p^*(r)$. The only periodic orbits contained in A^y are $\{y\}$, A and $\{y_1, y_2, \ldots, y_p\}$. On the other hand, by

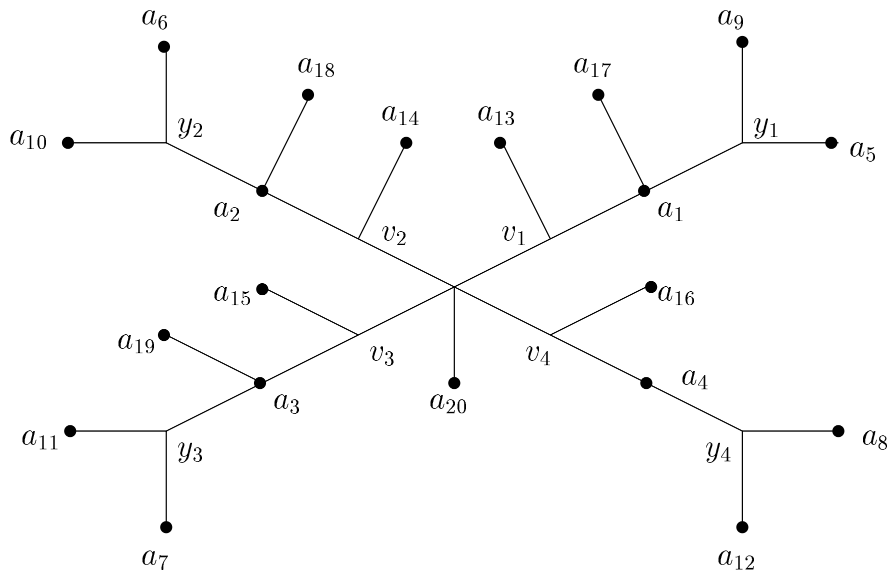

Figure 1.8: A (20,4,1)-model

Lemma 1.3.2 each periodic orbit which does not intersect A^y is associated to a loop. Each loop in the f-graph of A^y is either elementary or a concatenation of some elementary loops, which are α, β_i, γ_i, δ, ϵ, σ_i, τ and their respective shifts. The length of each of these elementary loops belongs to $p\mathbb{N} \cup (r + p\mathbb{N})$. Given $i, j \in \mathbb{N}$ and $m, n \in p\mathbb{N} \cup (r + p\mathbb{N})$, we have $im + jn \in \mathcal{S}_p^*(r)$. It follows that the length of each loop in the f-graph of A^y belongs to $\mathcal{S}_p^*(r)$ and thus (d) holds.

Case 4. *$p > 1$, $q > 0$ and $r \in p\mathbb{N}$.*

The construction is very similar to that of the previous case. We have $a_i \in Z(A^y)_{i \bmod p}$ for $1 \leq i < r$, but in this case we replace the edge $[a_r, a_{t+k}]$ by an edge $[y, a_r]$ contained in $\mathrm{Cl}(Z^*(A))$. The details are left to the reader. See Figure 1.8 for an example with $r = 20$, $p = 4$ and $q = 1$. ∎

Proof of Theorem 1.9.1. In order to prove (a) we use Proposition 1.9.2 iteratively. Set $k_i = p_1 p_2 \cdots p_i$ with $p_i > 1$ for $1 \leq i \leq m$. Let us consider a tree R_0 consisting of a single point x and the map $h_0(x) = x$. Then $(R_0, \{x\}, h_0)$ is a canonical model such that $\mathrm{Per}(h_0) = \{1\}$. We use Proposition 1.9.2 with $s = p_m$ and obtain a canonical model (R_1, B_1, h_1) such that $\mathrm{Per}(h_1) = \{1\} \cup p_m \,\mathrm{Per}(h_0) = \{1, p_m\}$ and $|B_1| = p_m$. We use again Proposition 1.9.2 with $s = p_{m-1}$ and obtain a canonical model (R_2, B_2, h_2) such that $\mathrm{Per}(h_2) = \{1\} \cup p_{m-1} \,\mathrm{Per}(h_1) = \{1, p_{m-1}, p_{m-1}p_m\}$ and $|B_2| = p_m p_{m-1}$. We can iterate this argument $m - 1$ times and finally obtain a canonical model

$(R_{m-1}, B_{m-1}, h_{m-1})$, which we denote as (R, B, h). Then $|B| = p_1 p_2 \cdots p_{m-1}$ and $\mathrm{Per}(h) = \{p_1, p_1p_2, \dots, p_1p_2\cdots p_m\} = K$. Thus (a) is proved.

Now we prove (b) analogously. We must construct a canonical model (S, P, g) with all the prescribed properties. By Proposition 1.9.3, we can consider an (r, p, q)-model, which we denote by (T_m, A_m, f_m). Since A_m is a periodic orbit, (T_m, A_m, f_m) is 0-orbital. As above, we use $m-1$ times Proposition 1.9.2 to obtain a sequence of canonical models (T_i, A_i, f_i) for $1 \le i < m$ such that

(i) For $1 \le i < m$ there exists $y_i \in \mathrm{Fix}(f_i)$ such that (T_i, A_i, f_i) is a y_i-expansive 0-orbital canonical model.

(ii) $(T_{i+1}, A_{i+1}, f_{i+1})$ is a partial p_i-reduction of (T_i, A_i, f_i) for each $1 \le i < m$. Furthermore, $\mathrm{Per}(f_i) = \{1\} \cup p_i \,\mathrm{Per}(f_{i+1})$.

(iii) $|A_i| = p_i |A_{i+1}|$ for $1 \le i < m$.

(iv) (T_i, A_i, f_i) is twist around y_i for $1 \le i < m$. Thus, by Remark 1.6.1, $\mathcal{E}(T_i, A_i, f_i) = \emptyset$. In particular, $\tilde{\mathcal{E}}(T_i, A_i, f_i) = \emptyset$.

By Proposition 1.9.3(a–c), $|A_m| = r$, there exists $y_m \in \mathrm{Fix}(f_m)$ such that (T_m, A_m, f_m) is non-twist around y_m and p is a type of $A_m^{y_m}$. Furthermore, since $\mathrm{En}(T_m) \subset A_m$, from Proposition 1.6.5 we get $\tilde{\mathcal{E}}(T_m, A_m, f_m) \neq \emptyset$.

So if we set $p_m = p$ then $\{(T_i, A_i, f_i), y_i, p_i\}_{i=1}^m$ is a sequence of partial reductions of (T_1, A_1, f_1). Therefore, if we define $(S, P, g) = (T_1, A_1, f_1)$ and $(\overline{S}, \overline{P}, \overline{g}) = (T_m, A_m, f_m)$ then $\{(\overline{S}, \overline{P}, \overline{g}), K\}$ is a complete reduction of (S, P, g).

From the properties of an (r, p, q)-model listed in Proposition 1.9.3 we get that $|\overline{P}| = |\overline{P}^{\circ}| = r$ and q is a rotation index associated to the type p. It follows that $p(\overline{S}, \overline{P}, \overline{g}) = p$ and $q(\overline{S}, \overline{P}, \overline{g}) = q$. Moreover, $n(\overline{S}, \overline{P}, \overline{g}) = 0$ since $(\overline{S}, \overline{P}, \overline{g})$ is 0-orbital. Finally,

$$\mathcal{S}_p^*(r + lp) \setminus \{2p, 3p, \dots, \lambda p\} \subset \mathrm{Per}(\overline{g}) \subset \mathcal{S}_p^*(r), \tag{1.25}$$

where $lp = p + q - (q \bmod p)$ and λp is the largest multiple of p smaller than $r + p + q + 1$. From (ii) it follows that

$$\mathrm{Per}(g) = K \cup k \,\mathrm{Per}(\overline{g}). \tag{1.26}$$

By Remark 1.2.2, the theorem follows from (1.25) and (1.26) by taking $\mathcal{C} = k \,\mathrm{Per}(\overline{g})$.

■

Chapter 2

The set of periods for tree maps

2.1 Introduction. How to compute the set of periods of a tree map

The aim of this chapter is to characterize the set of periods of a tree map $g : S \longrightarrow S$. In the Introduction to Chapter 1 we made a detailed discussion about which is the more natural approach to this problem, and we proposed a strategy consisting of three consecutive stages. Let us briefly recall that programme:

1. For each periodic orbit P of g, calculate the set Λ_P of periods of the corresponding P-monotone model $f_P : T_P \longrightarrow T_P$ or, if it is not possible, estimate the largest possible subset of Λ_P.
2. Prove that Λ_P is contained in the set of periods of each tree map exhibiting an orbit with the pattern of P. In particular, $\Lambda_P \subset \mathrm{Per}(g)$.
3. Consider each orbit P of g and its associated Λ_P, and then obtain the structure of the set of periods of g by describing the structure of the (perhaps uncountable) union of all sets Λ_P.

Chapter 1 has been devoted to perform Step 1 of the above programme in the case of piecewise monotone tree maps. That is, given a monotone model (T, P, f), we have calculated (as large as possible) subsets of $\mathrm{Per}(f)$. This estimation has been done by studying the loops of the Markov P-graph of f. This task is relatively simple (Lemma 1.4.6 and Theorem 1.4.7) when the Markov P-graph of f contains a special kind of loops, which we called *external* in Section 1.4. However, when the orbit P is *twist* (that is, when it presents certain rotational behavior around a fixed point according to the definition in page 30), the Markov P-graph of f lacks this kind of loops, and thus the results of Section 1.4 cannot be directly used. This gave rise to all the work of Section 1.6, where the sets of periods of twist models were studied. In particular, the notion of a *sequence of partial reductions* $\{(T_i, P_i, f_i), y_i, p_i\}_{i=1}^m$ was introduced in order to relate the set of periods of a monotone twist model (T_1, P_1, f_1) to the set of periods of its associated (non-twist) *reduced model* (T_m, P_m, f_m). The construction of a sequence of partial reductions was the most complicated part of Chapter 1, involving many results with lengthy technical proofs.

The goal of this chapter is to implement the above strategy in full. In particular we perform again Step 1 but, in doing it, we use a new simple and powerful argument which allows us to simplify the proofs at the same time that we improve the results we obtain. This will give rise to a revised version of the above strategy that will be implemented below. Next we briefly outline this slightly modified version of Step 1. Its detailed implementation is the matter of Section 2.3.

Let $g : S \longrightarrow S$ be a tree map and let P be a periodic orbit of g. We want to know which other orbits the map g will necessarily have, depending only on the combinatorics of $g|_P$. The set of periods of all these orbits will be called *the set of periods forced by* P, and will be denoted by Λ_P. To estimate it we proceed as follows. First we consider the model (S_1, P_1, g_1) with $S_1 = \langle P \rangle_S$, $P_1 = P$ and $g_1 = r \circ g$, where $r : S \longrightarrow \langle P \rangle_S$ is the natural retraction. It is easy to see that $\operatorname{Per}(g_1) \subset \operatorname{Per}(g)$. Next we carry out a reduction process to obtain what we have called a *sequence of block reductions* $\{(S_i, P_i, g_i), y_i, p_i\}_{i=1}^m$, such that:

(i) (S_i, P_i, g_i) is a periodic model with $|P_i| = p_i |P_{i+1}|$ for $1 \le i \le m$.
(ii) P_i is twist around y_i for $1 \le i < m$.
(iii) $\operatorname{Per}(g_i) \supset p_i \operatorname{Per}(g_{i+1})$ for $1 \le i < m$.
(iv) P_m is non-twist.

Observe that no additional assumptions of monotonicity or y-expansivity are made on each model (S_i, P_i, g_i). In particular, we are making use of a generalized notion of a *twist* orbit (see Section 2.3) which does not impose that the model is y-expansive as in Chapter 1.

From the definition of a sequence of block reductions it easily follows that $\operatorname{Per}(g) \supset \{1, p_1, p_1p_2, \ldots, p_1p_2 \cdots p_{m-1}\} \cup p_1p_2 \ldots p_{m-1} \operatorname{Per}(g_m)$. To complete Step 1 of our new strategy we consider a P_m-monotone model (T, P_m, f) associated to (having the same pattern of) (S_m, P_m, g_m) and we estimate the set of periods of f. We prove that, since (S_m, P_m, g_m) is non-twist, (T, P_m, f) is also non-twist. Therefore, it is unnecessary to construct a sequence of partial reductions of (T, P_m, f), and we can directly use Lemma 1.4.6 and Theorem 1.4.7 to calculate $\operatorname{Per}(f)$.

In this new approach, we can forget about all the complicated and technical work of Section 1.6, which was necessary in order to construct a sequence of partial reductions of a twist monotone model. The key point of this improvement is that a sequence of block reductions is much easier to construct. As an extra advantage, in this new approach it is easier to relate the set of periods of g to that of (S_m, P_m, g_m).

In view of what has been said above, we propose a revised strategy in order to estimate the set of periods of a tree map $g\colon S \longrightarrow S$, which can be arranged according to the following scheme:

Step 1. Given a model (S, P, g), construct a sequence of block reductions $\{(S_i, P_i, g_i), y_i, p_i\}_{i=1}^{m}$ of (S, P, g). Then,

$$\operatorname{Per}(g) \supset \{1, p_1, p_1p_2, \dots, p_1p_2\cdots p_{m-1}\} \cup p_1p_2\dots p_{m-1}\operatorname{Per}(g_m).$$

Step 2. Suppose that (S_m, P_m, g_m) is a non-twist model and let (T, P_m, f) be a monotone model with the same pattern of (S_m, P_m, g_m). We want to estimate a subset Γ_{P_m} of the set of periods of (T, P_m, f), which is as large as possible. Since (S_m, P_m, g_m) is non-twist, as we shall see, (T, P_m, f) is also non-twist. Therefore, the set Γ_{P_m} can be estimated directly by using Lemma 1.4.6 and Theorem 1.4.7.

Step 3. Prove that Γ_{P_m} is contained in the set of periods of each model having the pattern of (T, P_m, f). In particular, $\Gamma_{P_m} \subset \operatorname{Per}(g_m)$. Thus, from Step 1,

$$\operatorname{Per}(g) \supset \Lambda_P := \{1, p_1, p_1p_2, \dots, p_1p_2\cdots p_{m-1}\} \cup p_1p_2\dots p_{m-1}\Gamma_{P_m}$$

(observe that the set Λ_P depends on the sequence of block reductions $\{(S_i, P_i, g_i), y_i, p_i\}_{i=1}^{m}$ and, therefore, it is not uniquely defined).

Step 4. Consider each orbit P of g and its associated Λ_P. Then we obtain the structure of the set $\operatorname{Per}(g)$ by describing the structure of the (perhaps uncountable) union of all sets

$$\Lambda_P = \{1, p_1, p_1p_2, \dots, p_1p_2\cdots p_{m-1}\} \cup p_1p_2\dots p_{m-1}\Gamma_{P_m}.$$

To state the main result of this chapter, which gives a characterization of the sets of periods of tree maps, we need to introduce some notation. The basic tool in giving the structure of these sets of periods are the *Baldwin's orderings* ${}_p\geq$, which were defined in Chapter 1 (see page 8). We also recall that an *initial segment* of the ordering ${}_p\geq$ is defined to be any set $\mathcal{I}$ such

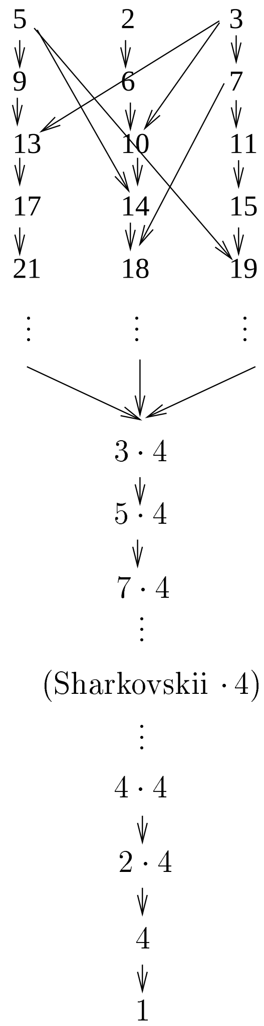

Figure 2.1: The ${}_4\!\geq$ ordering. The symbol $\to$ stands for ${}_4\!>$.

that if $m \in \mathcal{I}$ and $m \;{}_p\!> k$ then $k \in \mathcal{I}$. It is not difficult to see that $\mathcal{I}$ has at most $\max\{1, p-1\}$ ${}_p\!\geq$-maximal elements.

Let $p \in \mathbb{N}$. For each $1 \leq j < p$, the set $\{n \in \mathbb{N} \setminus \{1\} : n \equiv j \pmod{p}\}$ will be denoted by $\mathbb{N}_p^j$. Given an initial segment $\mathcal{I}$ of the ${}_p\!\geq$ ordering, the set of all $1 \leq j < p$ such that $\mathcal{I} \cap \mathbb{N}_p^j \neq \emptyset$ will be denoted by $B(\mathcal{I})$.

To understand how the set $B(\mathcal{I})$ relates to the structure of the ${}_p\!\geq$ ordering, it may be useful to visualize ${}_p\!\geq$ as a diagram where 1 is the least element, above it there are all the multiples of p arranged in the order induced by Sharkovskii ordering, and finally above there are all the nonmultiples of p arranged as $p-1$ *branches* according to their congruence class modulo p, in reverse ordering with respect to the usual ordering on the integers (see

Figure 2.1 for an example where $p = 4$).

Observe that, in the diagram of the ${}_p\geq$ ordering proposed above, $B(\mathcal{I})$ can be thought as the set of branches of nonmultiples of p which intersect $\mathcal{I}$.

Let S and $\widetilde{S}$ be trees and let $p \geq 2$ be an integer. We will write $S \sqsupset p\widetilde{S}$ to denote that S contains a subtree W with p endpoints such that $\widetilde{S}$ is homeomorphic to a connected component of $S \setminus \operatorname{Int}(W)$, and each connected component of $S \setminus \operatorname{Int}(W)$ has at least $|\operatorname{En}(\widetilde{S})|$ endpoints. Observe that, since there are p connected components of $S \setminus \operatorname{Int}(W)$ and each of them has at most one endpoint which is not an endpoint of S (the common point with W), it follows that $|\operatorname{En}(S)| \geq p(|\operatorname{En}(\widetilde{S})| - 1)$.

We will denote by Σ the set of all finite non-empty sequences of positive integers $(p_1, p_2, \ldots, p_m)$ such that $p_i \geq 2$ for $1 \leq i < m$. Given a tree S we will denote by Σ_S the set of all $(p_1, p_2, \ldots, p_m) \in \Sigma$ for which there exists a sequence of trees $(S_1, S_2, \ldots, S_m)$ verifying the following conditions:

(S 1) $S \supset S_1$, $S_i \sqsupset p_i S_{i+1}$ for $1 \leq i < m$ and $|\operatorname{En}(S_m)| \geq p_m$.

(S 2) S_i is not a star for $1 \leq i < m$.

In Lemma 2.6.1 we show that conditions (S 1–2) imply the following ones:

(S 3) $m \leq 1 + \log_2(|\operatorname{En}(S)| - 2)$ whenever $m \geq 3$.

(S 4) $p_1 \leq E_1 := |\operatorname{En}(S)|$ and $p_i \leq E_i := \frac{E_{i-1}}{p_{i-1}} + 1$ for $2 \leq i \leq m$.

Observe that, by (S 3–4), Σ_S is finite for each tree S.

For any finite non-empty sequence of positive integers $\underline{\mathsf{s}} = (p_1, p_2, \ldots, p_m)$ we set $\mathcal{K}_{\underline{\mathsf{s}}} = \{p_1, p_1p_2, \ldots, p_1p_2\cdots p_{m-1}\}$ (which is empty whenever $m = 1$) and $\lceil \underline{\mathsf{s}} \rceil = p_1p_2\cdots p_m \in \mathbb{N}$.

Theorem C (Characterization of the sets of periods of tree maps). *For each tree map $g : S \longrightarrow S$ there exists a finite set $\mathsf{S} \subset \Sigma_S$ such that*

$$\operatorname{Per}(g) = \bigcup_{\underline{\mathsf{s}} \in \mathsf{S}} \left(\mathcal{K}_{\underline{\mathsf{s}}} \cup \mathcal{F}_{\underline{\mathsf{s}}} \cup (\mathcal{I}_{\underline{\mathsf{s}}} \setminus \lceil \underline{\mathsf{s}} \rceil \{2, 3, \ldots, \lambda_{\underline{\mathsf{s}}}\})\right)$$

where, for each $\underline{\mathsf{s}} = (p_1, p_2, \ldots, p_m) \in \mathsf{S}$, $\lambda_{\underline{\mathsf{s}}}$ is a non-negative integer (when $\lambda_{\underline{\mathsf{s}}} < 2$ we understand that $\lceil \underline{\mathsf{s}} \rceil \{2, 3, \ldots, \lambda_{\underline{\mathsf{s}}}\} = \emptyset$) and

(a) $\mathcal{I}_{\underline{\mathsf{s}}}$ is an initial segment of the ${}_{\lceil \underline{\mathsf{s}} \rceil}\geq$ ordering such that each ${}_{\lceil \underline{\mathsf{s}} \rceil}\geq$-maximal element of $\mathcal{I}_{\underline{\mathsf{s}}}$ belongs to $\{1\} \cup p_1p_2\cdots p_{m-1}(\mathbb{N} \cup 2^\infty)$.

(b) If $\mathcal{I}_{\underline{\mathsf{s}}} \subsetneq \{1\} \cup \lceil \underline{\mathsf{s}} \rceil \mathbb{N}$ then $\lambda_{\underline{\mathsf{s}}} = 0$ and $\mathcal{F}_{\underline{\mathsf{s}}} = \emptyset$.

(c) $\mathcal{F}_{\underline{\mathsf{s}}}$ is disjoint from $\mathcal{K}_{\underline{\mathsf{s}}} \cup \mathcal{I}_{\underline{\mathsf{s}}} \setminus \lceil \underline{\mathsf{s}} \rceil \{2, 3, \ldots, \lambda_{\underline{\mathsf{s}}}\}$ and $\mathcal{F}_{\underline{\mathsf{s}}} \setminus \lceil \underline{\mathsf{s}} \rceil \mathbb{N} \subset \bigcup_{j \in B(\mathcal{I}_{\underline{\mathsf{s}}})} \mathbb{N}^j_{\lceil \underline{\mathsf{s}} \rceil}$.

(d) $\mathcal{F}_{\underline{\mathsf{s}}}$ is finite (or empty). When $\mathcal{F}_{\underline{\mathsf{s}}} \neq \emptyset$, we have $\min \mathcal{F}_{\underline{\mathsf{s}}} \geq \lambda_{\underline{\mathsf{s}}} \lceil \underline{\mathsf{s}} \rceil / 2$ and $|\mathcal{F}_{\underline{\mathsf{s}}}| < |B(\mathcal{I}_{\underline{\mathsf{s}}})| |\operatorname{En}(S)| + (\lambda_{\underline{\mathsf{s}}} + 2)/2$.

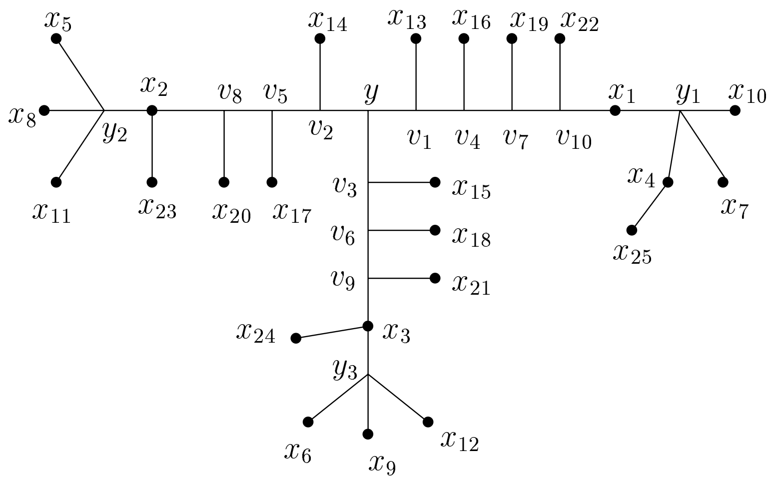

Figure 2.2: A $(25, 3, 8)$-model. We have $g(x_i) = x_{i+1 \bmod 25}$ for $1 \leq i \leq 25$, $g(y) = y$, $g(v_i) = v_{i+1}$ for $1 \leq i \leq 9$, $g(v_{10}) = x_2$ and $g(y_i) = y_{i+1 \bmod 3}$ for $1 \leq i \leq 3$.

Let us see some examples. Assume that $g\colon S \longrightarrow S$ is an (r, p, q)-model according to the definition given at page 48, and take $r = 14$, $p = 4$ and $q = 0$ (see Figure 1.5). By checking the loops of the Markov graph of g one easily gets (see also Case 2 of Proposition 1.9.3) that $\operatorname{Per}(g) = \mathcal{S}_4(14) \setminus \{8\}$. Therefore, Theorem C holds with $\underline{\mathsf{s}} = (4)$, $\mathsf{S} = \{\underline{\mathsf{s}}\}$, $\mathcal{F}_{\underline{\mathsf{s}}} = \emptyset$, $\mathcal{I}_{\underline{\mathsf{s}}} = \mathcal{S}_4(14)$ and $\lambda_{\underline{\mathsf{s}}} = 2$.

Assume now that $g\colon S \longrightarrow S$ is a $(25, 3, 8)$-model, which corresponds to Case 3 of Proposition 1.9.3 (see Figure 2.2). As above, it is not difficult to check that $\operatorname{Per}(g) = \{25, 28, 31, 62, 65, 68\} \cup \mathcal{S}_3(34) \setminus \{6, 9\}$. Hence, Theorem C holds by taking $\underline{\mathsf{s}} = (3)$, $\mathsf{S} = \{\underline{\mathsf{s}}\}$, $\mathcal{F}_{\underline{\mathsf{s}}} = \{25, 28, 31, 62, 65, 68\}$, $\mathcal{I}_{\underline{\mathsf{s}}} = \mathcal{S}_3(34)$ and $\lambda_{\underline{\mathsf{s}}} = 3$.

Theorem C tells us that (up to an explicitly bounded finite set) the set of periods of any tree map is a finite union of initial segments of Baldwin's orderings ${}_p\!\geq$ and the possible values of p are given in terms of the set of sequences Σ_S. This set, by definition, depends entirely on the combinatorial properties of the tree S. Let us see some examples (we use the above notations).

Assume that S is an n-star for some $n \geq 2$. Let us see that $\Sigma_S = \{(1), (2), \ldots, (n)\}$. Indeed, let $\underline{\mathsf{s}} = (p_1, p_2 \ldots, p_m) \in \Sigma_S$. By (S 1), S contains a tree S_1 which, by (S 2), is not a star when $m > 1$. Hence, $m = 1$ and $\mathcal{K}_{\underline{\mathsf{s}}} = \emptyset$ for each $\underline{\mathsf{s}} \in \Sigma_S$. On the other hand, $p_1 \leq n$ by (S 4). Therefore, by Theorem C, the set of periods of any continuous map from an n-star into itself is

$$\bigcup_{1 \le p \le n} (\mathcal{F}_p \cup (\mathcal{I}_p \setminus p\{2, 3, \ldots, \lambda_p\}))$$

where $\mathcal{I}_p$ is an initial segment of the ${}_p\!\geq$ ordering and $\mathcal{F}_p$ and λ_p satisfy (b–d). Although this statement differs (in a finite set) from that of Baldwin's theorem, in fact from the proof of Theorem C one gets that $\lambda_p = 0$ and $\mathcal{F}_p = \emptyset$ for all $1 \le p \le n$. This is due to the very special fact that if a star map g has a k-periodic orbit of type p with $k \notin p\mathbb{N}$, then g^p exhibits a *horseshoe* and thus $\mathrm{Per}(g) \supset p\mathbb{N}$. This is not true for a general tree map.

Assume now that S is the tree obtained as a union of a k-star T and k-many disjoint n-stars (with $k, n \ge 3$), each attached to T by identifying one of its endpoints with an endpoint of T. Observe that $|\operatorname{En}(S)| = k(n-1)$. In this case, by using (S 1–4), it is not difficult to check that

$$\Sigma_S = \{\, (p) : 1 \le p \le k(n-1)\} \cup \{\, (p_1, p_2) : 2 \le p_1 \le k \,,\, 1 \le p_2 \le n\}$$
$$\cup \{\, (p_1, p_2) : k < p_1 \le k(n-1) \,,\, 1 \le p_2 \le 2\}.$$

Thus, by Theorem C the set of periods of each tree map from S into itself is (up to a finite set) a union of initial segments of the orderings ${}_p\!\geq$, for $1 \le p \le 2k(n-1) = 2|\operatorname{En}(S)|$.

Although it is relatively simple (for a fixed tree S) to compute the set of sequences Σ_S in terms of the combinatorial properties of S, Theorem C does not tell us how to compute the subset S of Σ_S corresponding to a given tree map $g \colon S \longrightarrow S$. This set (as it can be easily inferred from the proof of the theorem) depends strongly on the map g. Therefore, Theorem C can be viewed as a "structure theorem", that is, a generic description of the set of periods of a tree map in terms of the combinatorics of the tree. Moving in this direction, the accuracy of that description arises as a natural question. In this sense, we have the following converse of Theorem C:

Theorem D. *Given a finite set* $\mathsf{S} \subset \Sigma$ *and a family* $\{\mathcal{F}_{\underline{\mathsf{s}}}, \mathcal{I}_{\underline{\mathsf{s}}}, \lambda_{\underline{\mathsf{s}}}\}_{\underline{\mathsf{s}} \in \mathsf{S}}$ *such that, for each* $\underline{\mathsf{s}} = (p_1, p_2, \ldots, p_m) \in \mathsf{S}$*,* $\mathcal{F}_{\underline{\mathsf{s}}}$ *is a finite subset of* $\mathbb{N}$*,* $\mathcal{I}_{\underline{\mathsf{s}}}$ *is an initial segment of the* ${}_{\lceil \underline{\mathsf{s}} \rceil}\!\geq$ *ordering whose* ${}_{\lceil \underline{\mathsf{s}} \rceil}\!\geq$*-maximal elements belong to* $\{1\} \cup p_1 p_2 \cdots p_{m-1}(\mathbb{N} \cup 2^\infty)$*, and* $\lambda_{\underline{\mathsf{s}}}$ *is a non-negative integer such that* $\lambda_{\underline{\mathsf{s}}} = 0$ *when* $\mathcal{I}_{\underline{\mathsf{s}}} \subsetneq \{1\} \cup \lceil \underline{\mathsf{s}} \rceil \mathbb{N}$*, there exist a tree* S *and a tree map* $g \colon S \longrightarrow S$ *such that* $\mathsf{S} \subset \Sigma_S$ *and*

$$Per(g) = \bigcup_{\underline{\mathsf{s}} \in \mathsf{S}} (\mathcal{K}_{\underline{\mathsf{s}}} \cup \mathcal{F}_{\underline{\mathsf{s}}} \cup (\mathcal{I}_{\underline{\mathsf{s}}} \setminus \lceil \underline{\mathsf{s}} \rceil \{2, 3, \ldots, \lambda_{\underline{\mathsf{s}}}\}))\,.$$

From the proof of Theorem D, one can easily compute an upper bound for the number of endpoints of the tree S in terms of the family $\{\mathcal{F}_{\underline{\mathsf{s}}}, \mathcal{I}_{\underline{\mathsf{s}}}, \lambda_{\underline{\mathsf{s}}}\}_{\underline{\mathsf{s}} \in \mathsf{S}}$.

However, in the spirit proposed above, this estimate for the size of S is unnecessary and has not been made explicit in the statement of the theorem. In fact, this is the reason why no additional assumptions on the finite sets $\mathcal{F}_{\underline{s}}$ are made. Indeed: given any tree map $f\colon T \longrightarrow T$ and any finite set $\mathcal{F} \subset \mathbb{N}$, it is not difficult to construct a tree $S \supset T$ and a tree map $g\colon S \longrightarrow S$ such that $\mathrm{Per}(g) = \mathrm{Per}(f) \cup \mathcal{F}$ (see Lemma 2.9.1).

Theorems C and D give a characterization of the *tree-realizable sets of periods*. That is, they describe which sort of subsets of $\mathbb{N}$ can be sets of periods of tree maps. This goes in the direction of Theorem 3.1 of [22], which states that a subset of $\mathbb{N}$ is a graph-realizable set of periods if and only if it coincides, up to a finite set, with a finite union of sets of the form $l\mathbb{N}$ and $\{2^n k : n \geq 0\}$. An analogous result for tree maps can be found in [14] (Theorem E). However, both characterizations depend on a finite set which is not described.

This chapter is organized as follows. In Section 2.2 we recall the notions and results from [3] concerning the minimality properties of monotone models, which will be strongly used in the rest of the chapter. In Sections 2.3, 2.4 and 2.5 we carry out respectively Steps 1, 2 and 3 of the strategy proposed above. In Section 2.6 we perform Step 4 and prove Theorem C. The full implementation of Step 3 depends on Theorem 2.5.1, a crucial result which we state (without proof) and use in Section 2.5. The proof of Theorem 2.5.1 has been separated into two cases. The first one is treated in Section 2.7. The proof of the second case, which we carry out in Section 2.8, is complicated and requires a variety of results and techniques borrowed from [2] (Nielsen fixed point classes, index theory, graph patterns). Due to the complexity of the proof, Section 2.8 has been divided into several subsections. Finally in Section 2.9 we prove Theorem D.

2.2 Minimality of the dynamics of monotone models. Preliminary results

The aim of this section is to recall the minimality properties of the dynamics of monotone models given in [3].

Let T be a tree and let $A \subset T$ be a finite subset of T. The pair (T, A) will be called a *pointed tree*. A nonempty set $Q \subset A$ is said to be a *discrete component* of (T, A) if either $|Q| > 1$ and there exists a connected component C of $T \setminus A$ such that $Q = \mathrm{Cl}(C) \cap A$, or $|Q| = 1$ and $Q = A$. We say that two pointed trees (T, A) and (T', A') are *equivalent* if there exists a bijection $\phi : A \longrightarrow A'$ which preserves discrete components. The equivalence class of a pointed tree (T, A) will be denoted by $[T, A]$.

Let (T, A) and (T', A') be equivalent pointed trees, and let $\theta : A \longrightarrow A$ and $\theta' : A' \longrightarrow A'$ be maps. We will say that θ and θ' are *equivalent* if $\theta' = \varphi \circ \theta \circ \varphi^{-1}$ for a bijection $\varphi : A \longrightarrow A'$ which preserves discrete components. The equivalence class of θ by this relation will be denoted by $[\theta]$. If $[T, A]$ is an equivalence class of pointed trees and $[\theta]$ is an equivalence class of maps then the pair $([T, A], [\theta])$ will be called a *pattern*. We say that a model (T, A, f) *exhibits* a pattern $(\mathcal{T}, \Theta)$ if $\mathcal{T} = [T, A]$ and $\Theta = [f|_A]$. This pattern will be denoted by $[T, A, f]$.

A well known quantitative measure of the dynamical complexity of a model which is widely used is the topological entropy (see [1]). It is an important topological invariant which is defined for continuous maps on compact metric spaces. The next theorem tells us that each pattern has a minimal model both from the point of view of its combinatorial simplicity (monotonicity) and its dynamical complexity (minimization of topological entropy).

Theorem 2.2.1 (Theorem A of [3]). *Let $(\mathcal{T}, \Theta)$ be a pattern. Then the following statements hold.*

(a) There exists a monotone model (T, A, f) exhibiting the pattern $(\mathcal{T}, \Theta)$.

(b) The topological entropy of f is the minimum within the class of models exhibiting the pattern $(\mathcal{T}, \Theta)$.

The dynamics of monotone models is also minimal in a sense different from the one given by Theorem 2.2.1(b). To precise this, we recall some more notions and results from [3].

Let $f : T \longrightarrow T$ be a tree map, and let $x, y \in T$ be fixed points of f^n for some $n \in \mathbb{N}$. We say that x and y are *f-monotone equivalent* if either $x = y$ or $f^n|_{\langle x,y \rangle}$ is monotone. Given a model (T, A, f), we say that a periodic point of f is *A-significant* if it is not f-monotone equivalent to any element of $A \cup V(T)$ and its period is minimal within its f-monotone equivalence class.

Remark 2.2.2. It is easy to see that if x and y are f-monotone equivalent then $f^i(x)$ and $f^i(y)$ are also f-monotone equivalent, for each $i \geq 0$. □

Let (T, A, f) be a model exhibiting a pattern $(\mathcal{T}, [\theta])$. Any (unordered) binary subset of a discrete component will be called a *basic path* of (T, A). The $(\mathcal{T}, [\theta])$*-path graph* is the oriented graph whose vertices are in one-to-one correspondence with the basic paths of (T, A) and there is an arrow from the vertex i to the vertex j if and only if the corresponding basic paths satisfy $\pi_j \subset \langle \theta(\pi_i) \rangle$. We will say that a loop $\pi_0 \to \pi_1 \to \ldots \to \pi_{n-1} \to \pi_0$ of the $(\mathcal{T}, [\theta])$-path graph and a point $x \in T$ are *associated* if $f^i(x) \in \langle \pi_{i \bmod n} \rangle$ for each $i \geq 0$. We recall that a loop is called *simple* if it is not an n-repetition of any other loop with $n \geq 2$ (see page 19).

Theorem 2.2.3 (Theorem C of [3]). *Let (T, A, f) be a monotone model exhibiting a pattern $(\mathcal{T}, \Theta)$. Then the following statements hold.*

(a) For each A-significant point x of f of period n there exists a unique simple loop β of length n of the $(\mathcal{T}, \Theta)$-path graph such that x and β are associated.

(b) Each simple loop β of length n of the $(\mathcal{T}, \Theta)$-path graph is associated either to an A-significant point of f of period n or to a periodic point which is f-monotone equivalent to a point of $A \cup V(T)$ and whose period is a divisor of n. In both cases, the point associated to β is unique up to f-monotone equivalence.

Theorem 2.2.4 (Theorem D of [3]). *Let (T, A, f) be a model exhibiting a pattern $(\mathcal{T}, \Theta)$. Let β be a simple loop of length n of the $(\mathcal{T}, \Theta)$-path graph. Then there exists a fixed point x of f^{2n} such that β and x are associated.*

Note that the $(\mathcal{T}, \Theta)$-path graph is a combinatorial object uniquely associated to the pattern $(\mathcal{T}, \Theta)$. That is, it can be constructed by using only combinatorial data of the pattern, independently from the particular choice of the representatives of $\mathcal{T}$ and Θ. This fact is crucial to understand why Theorems 2.2.3 and 2.2.4 imply that the dynamics of monotone models are minimal from the point of view of the set of periods. Indeed, by Theorem 2.2.3, essentially there is a one-to-one correspondence between the loops of the $(\mathcal{T}, \Theta)$-path graph and the significant periodic points of a monotone model exhibiting $(\mathcal{T}, \Theta)$. On the other hand, Theorem 2.2.4 tells us that each loop of length n in the $(\mathcal{T}, \Theta)$-path graph gives rise to a periodic point (whose period is a divisor of $2n$) on each model exhibiting the pattern $(\mathcal{T}, \Theta)$. Therefore, the set of periods of a monotone model is essentially contained (up to f-monotone equivalence and period-doubling) on the set of periods of each model of the same pattern.

2.3 Step 1. A reduction process

The aim of this section is to perform Step 1 of the strategy described in Section 2.1.

We start by introducing the notion of p-block reduction and studying its properties. This will be the key tool of the reduction process.

Let S be a tree and let $\widetilde{S}$ be a subtree of S (note that, for each connected component K of $S \setminus \widetilde{S}$, the set $\mathrm{Cl}(K) \cap \widetilde{S}$ consists of a single point). The *natural retraction* $r : S \longrightarrow \widetilde{S}$ is defined as follows. For each $x \in \widetilde{S}$ we set $r(x) = x$, and for each $x \in S \setminus \widetilde{S}$ we define $r(x)$ to be the unique point of

$\mathrm{Cl}(K) \cap \widetilde{S}$, where K is the connected component of $S \setminus \widetilde{S}$ containing x. It is not difficult to see that r is well defined and continuous.

Let (S, P, g) be a periodic model. We will say that (S, P, g) has a *p-block structure* (or simply a *block structure*) if there exists a partition $P = P_1 \cup P_2 \cup \ldots \cup P_p$ with $p \geq 2$ such that $\langle P_i \rangle_S \cap \langle P_j \rangle_S = \emptyset$ for $i \neq j$ and $f(P_i) = P_{i+1 \bmod p}$ for $1 \leq i \leq p$. Let $r_i : S \longrightarrow \langle P_i \rangle_S$ be the natural retraction from S onto $\langle P_i \rangle_S$ for each $1 \leq i \leq p$. A model $(\widetilde{S}, \widetilde{P}, \widetilde{g})$ is said to be a *p-block reduction* of (S, P, g) if and only if (S, P, g) admits a p-block structure and:

(i) There exists $1 \leq i \leq p$ such that $\widetilde{S} = \langle P_i \rangle_S$, $\widetilde{P} = P_i$ and $|\operatorname{En}(\widetilde{S})| = \min\{|\operatorname{En}(\langle P_j \rangle_S)|\}_{j=1}^{p}$.

(ii) $\widetilde{g} = r_1 \circ g|_{K_p} \circ r_p \circ g|_{K_{p-1}} \circ \ldots \circ r_3 \circ g|_{K_2} \circ r_2 \circ g|_{K_1}$, where K_j stands for $\langle P_{i-j+1 \bmod p} \rangle_S$ for each $1 \leq j \leq p$.

Observe that for a fixed partition $P = P_1 \cup P_2 \cup \ldots \cup P_p$ there are possibly several different p-block reductions of (S, P, g), since there can be several choices for $\widetilde{S}$.

The notion of p-block reduction is defined only for $p \geq 2$. In order to unify the notation and simplify the writing, given a monotone model (S, P, g) we will say that the model $(\langle P \rangle_S, P, r \circ g)$, where $r : S \longrightarrow \langle P \rangle_S$ is the natural retraction, is a *1-block reduction* of (S, P, g).

In the literature one can find several kinds of block structures and related notions for periodic orbits. In the interval case, Sharkovskii's *square root construction* (see [42] or [8]) is an earlier example of a block structure. Also the notion of *extension*, first appeared in [18], gives rise to some particular cases of block structures for interval periodic orbits. This notion was subsequently used by several authors to characterize the cyclic permutations exhibiting some kind of minimality property (the terms *simple*, *minimal* and *primary* are used depending on that property: see for instance [26], [30] or [10]). Finally, the notion of *division*, introduced in [35] for interval periodic orbits and generalized in [13], [12] and [14] to 3-star, n-star and tree maps respectively, has been used in a number of papers to study the set of periods and the topological entropy.

The next lemma studies some basic properties of a p-block reduction.

Lemma 2.3.1. *Let $(\widetilde{S}, \widetilde{P}, \widetilde{g})$ be a p-block reduction of a model (S, P, g), given by a partition $P = P_1 \cup P_2 \cup \ldots \cup P_p$. The following statements hold:*

(a) $\operatorname{En}(\widetilde{S}) \subset \widetilde{P}$.

(b) $(\widetilde{S}, \widetilde{P}, \widetilde{g})$ is a periodic model and $|\widetilde{P}| = |P|/p$.

(c) Let $\mathcal{Q}$ be the set of periodic points x of g such that $g^i(x) \in \langle P_{i+1 \bmod p} \rangle_S$ for $i \geq 0$. If $x \in \mathcal{Q}$ then x is an n-periodic point of $\widetilde{g}$ and an np-periodic

point of g, for some $n \in \mathbb{N}$. Therefore, $\operatorname{Per}(g) = p\operatorname{Per}(\widetilde{g}) \cup R$, *where R is the set of periods of all periodic points of g which do not belong to $\mathcal{Q}$.*

Proof. By assumption, $\langle P_i \rangle_S \cap \langle P_j \rangle_S = \emptyset$ for $i \neq j$ and $f(P_i) = P_{i+1 \bmod p}$ for $1 \leq i \leq p$. Set $K_i = \langle P_i \rangle_S$. Without loss of generality we can assume that $\widetilde{S} = K_1$ and $\widetilde{P} = P_1$. Statement (a) follows directly from the definition of $\widetilde{S}$ and $\widetilde{P}$. Since P is a periodic orbit and $f(P_i) = P_{i+1 \bmod p}$ for $1 \leq i \leq p$, statement (b) follows from the fact that the sets $\langle P_i \rangle_S$ are pairwise disjoint. Now we prove (c). Let $r_i : S \longrightarrow K_i$ be the natural retraction from S onto K_i for each $1 \leq i \leq p$ (we recall that $\widetilde{g} = r_1 \circ g|_{K_p} \circ r_p \circ g|_{K_{p-1}} \circ \ldots \circ r_3 \circ g|_{K_2} \circ r_2 \circ g|_{K_1}$). From the definition of a natural retraction and the fact that $\operatorname{En}(K_i) \subset P$ for $1 \leq i \leq p$, it follows that for each $z \in \widetilde{S}$ we have

$$\begin{cases} \operatorname{Orb}_{g^p}(z) = \operatorname{Orb}_{\widetilde{g}}(z) & \text{if } g^i(z) \in K_{i+1 \bmod p} \text{ for } i \geq 0, \text{ or} \\ \widetilde{g}^j(z) \in \widetilde{P} \text{ for some } j \geq 0 & \text{otherwise.} \end{cases} \tag{2.1}$$

Now let $x \in \mathcal{Q}$. In particular, x is a periodic point of g and $x \in K_1 = \widetilde{S}$. Since $g^i(x) \in K_{i+1 \bmod p}$ for $i \geq 0$ and $K_i \cap K_j = \emptyset$ for $i \neq j$, it follows that $|\operatorname{Orb}_g(x)| = np$ for some $n \in \mathbb{N}$. Moreover, from (2.1) we get that $\operatorname{Orb}_{g^p}(x) = \operatorname{Orb}_{\widetilde{g}}(x)$. Therefore, x is also a periodic point of $\widetilde{g}$, and $|\operatorname{Orb}_{\widetilde{g}}(x)| = n$. ■

Note that when $p = 1$, Lemma 2.3.1(c) states that $\operatorname{Per}(r \circ g) \subset \operatorname{Per}(g)$, where $r : S \longrightarrow \langle P \rangle_S$ is the natural retraction. This is well known even when one replaces $\langle P \rangle_S$ by any subtree of S (Corollary 4.2 of [16]).

When we are able to calculate the set of periods of a p-block reduction $\widetilde{g}$, Lemma 2.3.1(c) gives $p\operatorname{Per}(\widetilde{g})$ as an estimate of $\operatorname{Per}(g)$, the set of periods of the non-reduced model (S, P, g). However, this estimate can be very bad. In fact, as the following example shows, the set R of Lemma 2.3.1(c) can be infinite even when (S, P, g) is monotone. Let $S = [0, 1] \subset \mathbb{R}$ and set $P = \{x_i\}_{i=1}^6$ with $x_1 < x_4 < x_2 < x_5 < x_3 < x_6$. Let g be a P-monotone map verifying $g(x_i) = x_{i+1 \bmod 6}$ for $1 \leq i \leq 6$ (in the terminology of [8], the orbit P is a *2-extension* of the pattern $(1, 2, 3)$). It is not difficult to see that g has a 3-periodic orbit. Thus, from Sharkovskii's Theorem it follows that $\operatorname{Per}(g) = \mathbb{N}$. On the other hand, (S, P, g) has a 3-block structure defined by the partition $P = P_1 \cup P_2 \cup P_3$ where $P_1 = \{x_1, x_4\}$, $P_2 = \{x_2, x_5\}$ and $P_3 = \{x_3, x_6\}$. Thus (S, P, g) admits a 3-block reduction $(\widetilde{S}, \widetilde{P}, \widetilde{g})$ with $\widetilde{P} = P_1$ and $\widetilde{S} = [x_1, x_4]$. Moreover $\widetilde{P}$ is a 2-periodic orbit of $\widetilde{g}$ and $\widetilde{g}$ is $\widetilde{P}$-monotone. It follows that $\operatorname{Per}(\widetilde{g}) = \{1, 2\}$ and, hence, $p\operatorname{Per}(\widetilde{g}) = \{3, 6\}$.

To overcome the problem pointed out in the above example, we introduce the notion of a *twist orbit*. A twist orbit is a particular class of block structure for which it can be shown that, with the notation of Lemma 2.3.1, if (S, P, g)

is monotone and P is a twist orbit then R reduces to $\{1\}$ and thus $\operatorname{Per}(g) = p\operatorname{Per}(\widetilde{g}) \cup \{1\}$. The notion of a twist orbit is closely related to the idea of a division (see [14]). In fact, the name *twist orbit* was used in [14] to denote a notion that is very similar to the one we propose here.

In the rest of this chapter we will freely use the following notations and notions introduced in Chapter 1: $\mathcal{S}_p^*$ (page 14), *y-branches* (page 21), *type* and *y-typifiable set* (page 22), A^y, *y-expansive model* and *rotation index associated to a type* (page 23), *typical loop* and $\mathcal{E}(T, A, f)$ (page 24), A° and *n-orbital* (page 25) and, finally, $\widetilde{\mathcal{E}}(T, A, f)$ (page 34).

Let (S, P, g) be a periodic model, let $y \in \langle P \rangle \setminus P$ and let $Z_1, Z_2, \ldots, Z_p$ be the y-branches. We will say that (S, P, g) and P are *twist around* y (or simply *twist*) if either $|P| = 1$ or (S, P, g) has a p-block structure defined by the partition $P = \cup_{i=1}^p (P \cap Z_i)$. If $|P| > 1$ and there is no point y such that (S, P, g) is twist around y then (S, P, g) and P will be called *non-twist.*

We remark that if (S, P, g) is a twist model then P has a division (according to the definition given in [14]), but the converse is true only when S is an interval.

Remark 2.3.2. Let (S, P, g) be twist around y with $|P| > 1$ and let p be the number of y-branches. Since $y \notin P$, P is y-typifiable by Remark 1.4.1. Moreover, it is not difficult to see that the type of the set P is unique and it coincides with p. If in addition (S, P, g) is a y-expansive monotone model then it is twist around y according to the definition in page 30. Therefore, this new definition of a twist model is a generalization of the analogue concept introduced at the beginning of Section 1.6 for y-expansive models. $\square$

The next lemma shows that a twist model always admits a p-block reduction. Hence, in view of Lemma 2.3.1(c) the computation of its set of periods can be reduced to the computation of the (simpler) set of periods of the reduced model. This is what motivates the use of a reduction process of a model performed by a sequence of block reductions, as a first stage in the study of its set of periods. We recall that the notation $S \sqsupset p\widetilde{S}$ (where S and $\widetilde{S}$ are trees and $p \geq 2$ is an integer) has been introduced at page 62.

Lemma 2.3.3. *Let (S, P, g) be a twist model such that $|P| > 1$ and $\operatorname{En}(S) \subset P$. Then there exist an integer $p > 1$ and a p-block reduction $(\widetilde{S}, \widetilde{P}, \widetilde{g})$ of (S, P, g) such that $S \sqsupset p\widetilde{S}$ (hence, $|\operatorname{En}(\widetilde{S})| \leq |\operatorname{En}(S)|/p + 1$).*

Proof. Let $y \in \langle P \rangle \setminus P = S \setminus P$ be a point such that (S, P, g) is twist around y. Let $Z_1, Z_2, \ldots, Z_p$ be the y-branches and let W be the closure of the connected component of $S \setminus P$ containing y. Since $|P| > 1$ and $\operatorname{En}(S) \subset P$, it follows that $p > 1$ and $|\operatorname{En}(W)| = p$.

Since (S, P, g) is twist around y, the partition $\cup_{i=1}^{p}(P \cap Z_i)$ defines a block structure for (S, P, g). Thus there exists a p-block reduction $(\widetilde{S}, \widetilde{P}, \widetilde{g})$ of (S, P, g). In particular, $|\operatorname{En}(\widetilde{S})| = \min\{|\operatorname{En}(Z_i)|\}_{i=1}^{p}$. Hence, $S \sqsupset p\widetilde{S}$ holds since Z_i are the connected components of $S \setminus \operatorname{Int}(W)$, $|\operatorname{En}(W)| = p$ and Z_i has at least $|\operatorname{En}(\widetilde{S})|$ endpoints for $1 \leq i \leq p$. ■

Before defining the notion of a *sequence of block reductions*, we need to remark that until the end of this chapter we will deal frequently with monotone models (see for instance Step 2 of the strategy described in Section 2.1, where a monotone model of a given pattern has to be chosen). We stress the fact that, from now on, all the considered monotone models will be in addition *canonical models* (see page 26 for a definition). The notion of a canonical model was first introduced in [3] to be a special kind of "ultra-reduced" monotone model. In Theorem 1.5.3 we showed that, given a monotone model (S, B, g), a canonical model (T, A, f) can be constructed from (S, B, g) essentially by collapsing invariant forests which do not intersect B. It easily follows that the patterns $[T, A, f]$ and $[S, B, g]$ coincide. Thus from Theorem 2.2.1(a) it follows that *each pattern admits a canonical model* (compare with Theorem B of [3]).

Also we need to recall (see Proposition 1.5.4) that for each periodic (and thus 0-orbital) canonical model (T, A, f) there exists $y \in \operatorname{Fix}(f)$ such that (T, A, f) is y-expansive. This property will be frequently used for the rest of this chapter. We will simply say that (T, A, f) *is a y-expansive canonical model*, and this phrase will mean that y is a fixed point of f and that we have chosen it in such a way that (T, A, f) is y-expansive.

Now we are ready to define in detail the notion of a *sequence of block reductions*, that will allow us to perform the reduction of a twist model to a model whose set of periods can be computed in an easier way. This notion is similar in some sense to the notions *z-tower* and *snowflake* introduced by Blokh in [24].

Let (S, P, g) be a periodic model. A sequence $\{(S_i, P_i, g_i), y_i, p_i\}_{i=1}^{m}$ will be called a *sequence of block reductions* of (S, P, g) if and only if:

(SBR 1) (S_1, P_1, g_1) is a 1-block reduction of (S, P, g).

(SBR 2) $(S_{i+1}, P_{i+1}, g_{i+1})$ is a p_i-block reduction of (S_i, P_i, g_i) such that $S_i \sqsupset p_i S_{i+1}$ (thus $|\operatorname{En}(S_{i+1})| \leq |\operatorname{En}(S_i)|/p_i + 1$) and $p_i \geq 2$ for $1 \leq i < m$.

(SBR 3) P_i is twist around y_i and S_i is not a star (in particular, $3 < |\operatorname{En}(S_i)| \leq |P_i|$) for $1 \leq i < m$.

(SBR 4) Either P_m is non-twist or S_m is a star. In both cases, one of the following two statements holds:

(SBR 4.1) $|P_m| = 1$, $p_m = 1$ and y_m is the unique element of P_m

or

(SBR 4.2) (T, A, f) is y_m-expansive and p_m is a type of A^{y_m}, for some canonical model (T, A, f) of the pattern $[S_m, P_m, g_m]$.

Remark 2.3.4. Given a sequence of block reductions $\{(S_i, P_i, g_i), y_i, p_i\}_{i=1}^m$ of (S, P, g), from (SBR 1–2) and Lemma 2.3.1 it follows that $\operatorname{En}(S_i) \subset P_i$ and P_i is a periodic orbit of g_i for each $1 \leq i \leq m$. Moreover, $|P| = p_1 p_2 \cdots p_{m-1} |P_m|$, $\operatorname{Per}(g) \supset \operatorname{Per}(g_1)$ and $\operatorname{Per}(g_i) \supset p_i \operatorname{Per}(g_{i+1}) \cup \{1\}$ for $1 \leq i < m$. Thus $\operatorname{Per}(g) \supset \{1, p_1, p_1p_2, \ldots, p_1p_2 \cdots p_{m-1}\} \cup p_1p_2 \ldots p_{m-1} \operatorname{Per}(g_m)$. □

The next proposition tells us that the reduction process consisting of a sequence of block reductions is always possible, thus completing Step 1 of the programme described in Section 2.1.

Proposition 2.3.5. *Each periodic model (S, P, g) admits a sequence of block reductions.*

Proof. Let us formally denote $\{(S_i, P_i, g_i), y_i, p_i\}_{i=1}^k$ by $\mathcal{B}^k$ for any $k \geq 0$ (by convention we set $\mathcal{B}^0 = \emptyset$).

We start by defining (S_1, P_1, g_1) to be a 1-block reduction of (S, P, g). By Lemma 2.3.1(a), $\operatorname{En}(S_1) \subset P_1$. Moreover, (SBR 1–3) trivially hold with 1 instead of m. Now we proceed by induction on k.

Let $k \geq 1$ and assume that we have constructed a sequence $\mathcal{B}^{k-1}$ and a model (S_k, P_k, g_k) such that:

(a) P_k is a periodic orbit with $\operatorname{En}(S_k) \subset P_k$.

(b) (SBR 1–3) hold with k instead of m.

If, in addition, there exist y_k and p_k such that (SBR 4) holds with k instead of m then $\mathcal{B}^k$ is a sequence of block reductions and we are done. So, we must define y_k and p_k and then decide whether $\mathcal{B}^k$ is a sequence of block reductions (in this case we stop by setting $m = k$), or we construct a model $(S_{k+1}, P_{k+1}, g_{k+1})$ such that $\mathcal{B}^k$ and $(S_{k+1}, P_{k+1}, g_{k+1})$ verify (a) and (b) with $k+1$ instead of k and we continue the induction procedure.

Assume that $|P_k| = 1$. In this case we set $p_k = 1$ and define y_k to be the unique element of P_k. Then $\mathcal{B}^k$ verifies (SBR 4.1) and thus it is a sequence of block reductions.

When $|P_k| > 1$ we consider two cases.

Case 1. *Either P_k is non-twist or S_k is a star.*

Let (T, A, f) be a canonical model of $[S_k, P_k, g_k]$. Therefore, (T, A, f) is orbital, because $|A| = |P_k| > 1$ and A is a periodic orbit by (a). Then, by

Proposition 1.5.4, there exists $y_k \in \mathrm{Fix}(f)$ such that (T, A, f) is y_k-expansive. Let p_k be a type of A^{y_k}. Then $\mathcal{B}^k$ verifies (SBR 4.2) and thus it is a sequence of block reductions.

Case 2. *P_k is twist and S_k is not a star.*

Clearly, $|P_k| > 1$. We take y_k so that P_k is twist around y_k. Since (a) holds, by Lemma 2.3.3 there exist a model $(S_{k+1}, P_{k+1}, g_{k+1})$ and an integer $p_k \geq 2$ such that $(S_{k+1}, P_{k+1}, g_{k+1})$ is a p_k-block reduction of (S_k, P_k, g_k) and $S_k \sqsupset p_k S_{k+1}$. By Lemma 2.3.1, $\mathrm{En}(S_{k+1}) \subset P_{k+1}$ and P_{k+1} is a periodic orbit of g_{k+1}. Therefore, $\mathcal{B}^k$ and $(S_{k+1}, P_{k+1}, g_{k+1})$ verify (a) and (b) with $k+1$ instead of k. Moreover, from Lemma 2.3.1(b) we have that $|P_{k+1}| = |P_k|/p_k < |P_k|$. Therefore, by the finiteness of P, this iterative construction stops after finitely many steps. ∎

2.4 Step 2. Computing sets of periods of non-twist canonical models

The aim of this section is to perform Step 2 of the strategy described in Section 2.1 by estimating the set of periods of a canonical model exhibiting the pattern of a given non-twist periodic model. The following lemma ensures that such a canonical model is also non-twist.

Lemma 2.4.1. *Let (S, P, g) be a periodic model and let (T, A, f) be a canonical model of $[S, P, g]$. Then (T, A, f) is non-twist if and only if (S, P, g) is non-twist.*

Proof. Assume that (T, A, f) is twist around a point $y \in T \backslash A$. Since (T, A, f) is a canonical model, $\mathrm{En}(T) \subset A$. Set $Z = Z(A)^y$ and let $Z_1, Z_2, \ldots, Z_p$ be the y-branches. We have $f(A \cap Z_i) = A \cap Z_{i+1 \bmod p}$ for $1 \leq i \leq p$. Note that there exists a discrete component X of (T, A) such that $X = \mathrm{Cl}(Z) \setminus Z$. Moreover, $A \cap Z_i$ is a union of discrete components of (T, A) for $1 \leq i \leq p$. Let $\theta : A \longrightarrow P$ be a bijection which preserves discrete components and $g|_P = \theta \circ f \circ \theta^{-1}|_P$ (such a bijection exists because (T, A, f) exhibits the pattern $[S, P, g]$). Set $P_i = \theta(A \cap Z_i) \subset P$ and let W_i be the closure of the connected component of $S \setminus \langle \theta(X) \rangle_S$ which contains P_i for $1 \leq i \leq p$. Observe that $W_1, W_2, \ldots, W_p$ are the w-branches for each point $w \in \mathrm{Int}(\langle \theta(X) \rangle_S)$. We have $g(P \cap W_i) = \theta(f(\theta^{-1}(P \cap W_i))) = \theta(f(\theta^{-1}(P_i))) = \theta(f(A \cap Z_i)) \subset \theta(A \cap Z_{i+1 \bmod p}) = P_{i+1 \bmod p} = P \cap W_{i+1 \bmod p}$ for $1 \leq i \leq p$. Therefore, (S, P, g) is twist around any point from $\mathrm{Int}(\langle \theta(X) \rangle_S)$.

The proof of the "only if" part of the lemma is analogous. ∎

We recall that each periodic canonical model (T, A, f) is y-expansive for some fixed point y of f. If in addition (T, A, f) is non-twist, then the set of periods of f can be estimated by using the results of Chapter 1. This has already been done in the proof of Theorem A by using Lemma 1.4.6 and Theorem 1.4.7, which allowed us to compute periodic orbits of f by means of the Markov f-graph of A^y. It turns out that all orbits obtained in this manner satisfy the following property, which will be essential for the rest of this chapter.

($\star$)-property. Let (T, A, f) be a y-expansive canonical model and let x be a periodic point of f. We say that x *satisfies the* ($\star$)*-property* if $\operatorname{Orb}(x) \cap A^y = \emptyset$ and x is associated to a unique simple loop in the Markov f-graph of A^y which strictly contains a typical loop.

The next theorem, which is the main result of this section, is a particular instance of Theorem A for non-twist y-expansive canonical models. It completes Step 2.

Theorem 2.4.2. *Let (T, A, f) be a non-twist y-expansive canonical model and let p be a type of A^y. Then $Per(f) \supset \mathcal{S}_p^*(|A|+lp) \setminus \{p, 2p, 3p, \dots, \lambda p\}$ for some integers $\lambda, l \geq 0$ such that $lp \leq |\operatorname{En}(T)| - p$ and $\lambda p \leq 2|A|$. Moreover, each of these periods (except 1) corresponds to a periodic point which satisfies the ($\star$)-property.*

Proof. Since A is a periodic orbit, (T, A, f) is 0-orbital (and thus 2-orbital) and $A^\circ = A$. Hence, $\operatorname{En}(T) \subset A^\circ$. Then, by Proposition 1.6.5, $\widetilde{\mathcal{E}}(T, A, f) \neq \emptyset$. Since in addition (T, A, f) is y-expansive, in the terminology of Chapter 1 we have that $\{(T, A, f), y, p\}$ is a sequence of partial reductions of (T, A, f) (see page 38). Therefore, if we take a rotation index q associated to the type p, from the proof of Theorem A (see Section 1.7) by taking $K = \{1\}$ and $k = 1$, it follows that $Per(f) \supset \mathcal{S}_p^*(|A| + lp) \setminus \{p, 2p, 3p, \dots, \lambda p\}$ for some integers $\lambda, l \geq 0$ such that $lp \leq p + q - (q \bmod p)$ and $\lambda p \leq |A| + p + q + 1$. The fact that the periodic orbits which we obtain in this manner satisfy the ($\star$)-property follows from the proofs of Lemma 1.4.6 and Theorem 1.4.7.

Let us see that $lp \leq |\operatorname{En}(T)| - p$. Since $lp \leq p + q - (q \bmod p)$ and $p + q - (q \bmod p) = 0$ when $q = 0$ (recall that we are taking $\{1, 2, \dots, p\}$ as the representatives of the classes of $\mathbb{Z}/p\mathbb{Z}$), it follows that $lp = 0 \leq |\operatorname{En}(T)| - p$ when $q = 0$. Assume that $q > 0$ and observe that the hypotheses of Proposition 1.8.1 hold. From the proof of Proposition 1.8.1(b) (see, in particular, (1.18), and recall that $|X(A)| \leq |\operatorname{En}(T)|$) it follows that $2p + q - 1 \leq |\operatorname{En}(T)|$. Therefore, $lp \leq p + q - (q \bmod p) \leq p + q - 1 \leq |\operatorname{En}(T)| - p$.

Finally, $\lambda p \leq 2|A|$ follows from $\lambda p \leq |A| + p + q + 1$, Proposition 1.8.1(c) when $q = 0$ and Proposition 1.8.1(e) when $q > 0$. ■

2.5 Step 3 and inclusion of periods

We start this section by stating Theorem 2.5.1 that allows us to compare the periods of the models (S_m, P_m, g_m) and (T, P_m, f) (which exhibits the pattern $[S_m, P_m, g_m]$ by the assumptions of Step 2). This result, that will be proved later, tells us that when (T, P_m, f) is a non-twist y-expansive canonical model and it satisfies a piecewise linearity condition, then all the periods of f computed in Theorem 2.4.2 are also periods of g_m. In the second part of this section we will prove Theorem 2.5.2 which, given a sequence of block reductions, describes the set of periods forced by a periodic orbit of a tree map. Theorem 2.5.2, whose proof uses Theorems 2.4.2 and 2.5.1, is the final outcome of Steps 1, 2 and 3.

Next we introduce the piecewise linearity condition mentioned above. It is well known that a tree T admits a *taxicab metric* $d : T \longrightarrow T$ which, by definition, satisfies $d(x, y) = d(x, z) + d(z, y)$ for each $x, y \in T$ and $z \in [x, y]$. Let I be a closed interval and let $f : I \longrightarrow T$ be continuous. We say that f is *linear* if either $f(I)$ reduces to a point, or there exist isometries $\theta : [0, 1] \longrightarrow I$ and $\phi : f(I) \longrightarrow [0, 1]$ given by a taxicab metric on T and the Lebesgue measure on $[0, 1]$ such that $\phi \circ f \circ \theta$ is an affine map. A y-expansive canonical model (T, A, f) will be called *linear* if f is linear on each A^y-basic interval. Without loss of generality, every y-expansive canonical model can be assumed to be linear.

Theorem 2.5.1. *Let (S, P, g) be a non-twist model and let (T, A, f) be a y-expansive linear canonical model of $[S, P, g]$. For each $n \in \mathbb{N}$ such that f has a periodic point of period n satisfying the $(\star)$-property it follows that $n \in \operatorname{Per}(g)$.*

In Section 2.7 we prove Theorem 2.5.1 when n is larger than a constant $M(S)$ which only depends on the number of endpoints and the number of vertices of the tree S. The proof in the case $n \leq M(S)$ is much more involved and requires some strong machinery. It is performed in Section 2.8.

The next goal of this section is to summarize Steps 1, 2 and 3 by describing the set of periods forced by a periodic orbit of a tree map. To this end we need to introduce some more notation.

Let $\mathcal{B} = \{(S_i, P_i, g_i), y_i, p_i\}_{i=1}^{m}$ be a sequence of block reductions of a model (S, P, g), where $g : S \longrightarrow S$ is a tree map and P is a periodic orbit of g. To such a sequence of block reductions we will associate a 4-tuple

$(\underline{\mathsf{s}}, \tau, l, \lambda) = (\underline{\mathsf{s}}(\mathcal{B}), \tau(\mathcal{B}), l(\mathcal{B}), \lambda(\mathcal{B}))$, called a *characteristic of* $\mathcal{B}$, which is defined as follows:

- $\underline{\mathsf{s}} = \underline{\mathsf{s}}(\mathcal{B})$ denotes the sequence $(p_1, p_2, \dots, p_m)$;
- $\tau = \tau(\mathcal{B})$ denotes the formal symbol ϕ if (S_m, P_m, g_m) is twist around y_m, and the symbol $*$ otherwise;
- the non-negative integers $l = l(\mathcal{B})$ and $\lambda = \lambda(\mathcal{B})$ are set to 0 when $\tau = \phi$. In the case $\tau = *$, let (T, A, f) be the y_m-expansive canonical model of $[S_m, P_m, g_m]$ given by (SBR 4.2) (note that, since $\tau = *$, P_m is non-twist; in particular, $|P_m| > 1$ and thus (SBR 4.1) does not hold). Then, the numbers l and λ are the ones provided by Theorem 2.4.2 for the model (T, A, f). Observe that the assumptions of Theorem 2.4.2 hold (with $p = p_m$) because, by Lemma 2.4.1, we know that (T, A, f) is non-twist.

Note that, given a sequence of block reductions $\mathcal{B}$, the sequence $\underline{\mathsf{s}}(\mathcal{B})$ and the symbol $\tau(\mathcal{B})$ are uniquely determined, while the numbers $l(\mathcal{B})$ and $\lambda(\mathcal{B})$ depend on the election of a rotation index associated to the type p_m (see the proof of Theorem 2.4.2). Therefore, a characteristic of a sequence of block reductions is not uniquely determined.

In what follows, $\mathcal{S}_p^{\tau}(n)$ denotes $\mathcal{S}_p^{*}(n)$ when $\tau = *$ and $\mathcal{S}_p(n)$ when $\tau = \phi$. Then, given a characteristic $(\underline{\mathsf{s}}, \tau, l, \lambda)$ of a sequence of block reductions of a periodic model (S, P, g) we set:

$$\Lambda(\underline{\mathsf{s}}, \tau, l, \lambda) := \{|P|\} \cup \mathcal{K}_{\underline{\mathsf{s}}} \cup \left(\mathcal{S}^{\tau}_{\lceil \underline{\mathsf{s}} \rceil}(|P| + l\lceil \underline{\mathsf{s}} \rceil) \setminus \lceil \underline{\mathsf{s}} \rceil \{2, 3, \dots, \lambda\}\right).$$

Theorem 2.5.2 (The set of periods forced by a periodic orbit). *Let* $g : S \longrightarrow S$ *be a tree map, let* P *be a periodic orbit of* g *and let* $(\underline{\mathsf{s}}, \tau, l, \lambda)$ *be a characteristic of a sequence of block reductions of* (S, P, g)*. Then,* $\operatorname{Per}(g) \supset \Lambda(\underline{\mathsf{s}}, \tau, l, \lambda)$.

Proof. Let $\{(S_i, P_i, g_i), y_i, p_i\}_{i=1}^m$ be a sequence of block reductions of (S, P, g) which has $(\underline{\mathsf{s}}, \tau, l, \lambda)$ as characteristic and set $p = p_m$ and $k = \max \mathcal{K}_{\underline{\mathsf{s}}} = \lceil \underline{\mathsf{s}} \rceil / p$. By Remark 2.3.4,

$$|P| = k|P_m| \quad \text{and} \quad \operatorname{Per}(g) \supset \{|P|, 1\} \cup \mathcal{K}_{\underline{\mathsf{s}}} \cup k \operatorname{Per}(g_m). \tag{2.2}$$

Now we consider the case $\tau = \phi$ (that is, P_m is twist). Then, S_m is a star by (SBR 4) and, by Baldwin's theorem (Theorem 2.12 of [16]), $\operatorname{Per}(g_m) \supset \mathcal{S}_p(|P_m|)$. From the definitions it is not difficult to see that $\{1\} \cup k\mathcal{S}_p(|P_m|) = \{k\} \cup \mathcal{S}_{kp}(k|P_m|)$ (in fact this already follows from Remark 1.2.2 when $|P_m| \notin p\mathbb{N}$). Therefore, from (2.2) it follows that

$$\begin{aligned} \operatorname{Per}(g) &\supset \{|P|, 1\} \cup \mathcal{K}_{\underline{\mathsf{s}}} \cup k\mathcal{S}_p(|P_m|) = \{|P|\} \cup \mathcal{K}_{\underline{\mathsf{s}}} \cup \mathcal{S}_{kp}(k|P_m|) \\ &= \{|P|\} \cup \mathcal{K}_{\underline{\mathsf{s}}} \cup \left(\mathcal{S}^{\tau}_{\lceil \underline{\mathsf{s}} \rceil}(|P| + l\lceil \underline{\mathsf{s}} \rceil) \setminus \lceil \underline{\mathsf{s}} \rceil \{2, 3, \dots, \lambda\}\right). \end{aligned}$$

Now we consider the case $\tau = *$. Observe that, since P_m is non-twist, $|P_m| > 1$. Let (T, A, f), y_m and p be given by (SBR 4.2). Then, (T, A, f) is a y_m-expansive canonical model of $[S_m, P_m, g_m]$ and p is a type of A^{y_m}. Moreover, as it has been said before, the model (T, A, f) can be assumed to be linear without loss of generality, and it is non-twist by Lemma 2.4.1.

By Theorem 2.4.2, $\operatorname{Per}(f) \supset \mathcal{S}_p^*(|P_m| + lp) \setminus p\{1, 2, 3, \dots, \lambda\}$. Moreover, each of these periods corresponds to a periodic point which satisfies the $(\star)$-property. Thus, by Theorem 2.5.1, all these periods are also periods of g_m. In consequence, $\operatorname{Per}(g_m) \supset \mathcal{S}_p^*(|P_m|+lp) \setminus p\{1, 2, 3, \dots, \lambda\}$. Assume that we have proved also that $p \in \operatorname{Per}(g_m)$. Then, $\operatorname{Per}(g_m) \supset \mathcal{S}_p^*(|P_m| + lp) \setminus p\{2, 3, \dots, \lambda\}$ and the theorem follows as above by using (2.2) and Remark 1.2.2.

To end the proof of the theorem it is enough to show that $p \in \operatorname{Per}(g_m)$. To do it, we claim that A has type p. Let us prove the claim. To simplify the writing, until the end of this proof x_i will stand for $x(A^{y_m})_{i \bmod p}$ for each $i \geq 0$ (see page 22 for this notation and the ones used below; in particular, we recall that the points x_i are indexed according to the type p). Take $j \in \{1, 2, \dots, p\}$ and $z \in X(A)$ with $x_j \preceq z$ (recall that $X(A) \subset A \subset A^{y_m}$ and that, since (T, A, f) is a canonical model, $\operatorname{En}(T) \subset A$). Since f is $(A \cup \{y_m\})$-monotone and $(y_m, z) \cap A = \emptyset$, it follows that there exists $z' \in X(A)$ such that $y_m \prec x_{j+1} \preceq z' \preceq f(z)$. In particular we have $\Phi_A(z) = z'$. Since A is finite, by iterating this argument we obtain a periodic orbit $\{a^0, a^1, \dots, a^{pr-1}\} \subset X(A)$ of Φ_A with $r \geq 1$ such that $a^i \succeq x_{k+i}$ for some $k \in \{1, 2, \dots, p\}$ and $0 \leq i \leq pr - 1$. In particular, A has type pr.

Suppose that $a^i = x_{k+i}$ for some $0 \leq i \leq pr - 1$. Since p is a type of A^{y_m}, it follows that $a^{i+p-1 \bmod pr} \succeq x_{k+i+p-1} = x_{k+i-1}$. Therefore, since f is $(A \cup \{y_m\})$-monotone, it follows that $a^i = x_{k+i} \preceq f(x_{k+i-1}) \preceq f(a^{i+p-1 \bmod pr})$. That is, $a^{i+p \bmod pr} = \Phi_A(a^{i+p-1 \bmod pr}) = a^i$ and, hence, A has type p. This ends the proof of the claim in this case.

Now consider the case $x_{k+i} \prec a^i$ for all $0 \leq i \leq pr-1$ and suppose that $r > 1$. Then $x_k \prec a^0$, $x_k \prec a^p$ and $T^0 = \langle \{y_m, a^0, a^p\} \rangle$ is a 3-star whose central point is x_k. Since f is $(A \cup \{y_m\})$-monotone, $f(T^0) = \langle \{y_m, f(a^0), f(a^p)\} \rangle$. Furthermore, from Remark 1.2.1 it follows that $f(x_k) = x_{k+1}$. Since $a^1 \preceq f(a^0)$ and $a^{p+1} \preceq f(a^p)$, we have that $T^1 = \langle \{y_m, a^1, a^{p+1}\} \rangle$ is a 3-star contained in $f(T^0)$ whose central point is x_{k+1}. By iterating p times this argument we get that $\{x_1, x_2, \dots, x_p\}$ is a periodic orbit of f contained in $V(T) \cap Z^\star(A)$, in contradiction with the fact that (T, A, f) is y_m-expansive. This ends the proof of the claim.

Finally we prove that $p \in \operatorname{Per}(g_m)$. Observe that $X(A)$ is a discrete component of (T, A). Since $[T, A, f] = [S_m, P_m, g_m]$, there exists a bijection $\theta \colon A \longrightarrow P_m$ which sends discrete components of (T, A) to discrete components of (S_m, P_m). Let $Z = \operatorname{Int}(\langle \theta(X(A)) \rangle_{S_m})$ and let $w \in Z$. Since

$g_m \circ \theta|_A = \theta \circ f|_A$ and, by the claim above, A has type p, it easily follows that $g_m(\theta(a^i)) \succeq_w \theta(a^{i+1})$ for $0 \le i < p-1$ and $g_m(\theta(a^{p-1})) \succeq_w \theta(a^0)$.

Let $r : S_m \longrightarrow \mathrm{Cl}(Z)$ be the natural retraction. Then $r \circ g_m(\theta(a^i)) = \theta(a^{i+1})$ for $0 \le i < p-1$ and $r \circ g_m(\theta(a^{p-1})) = \theta(a^0)$. Hence $\theta(a^0)$ is a p-periodic point of $r \circ g_m$ and thus $p \in \mathrm{Per}(r \circ g_m)$. Then, the theorem follows because $\mathrm{Per}(r \circ g_m) \subset \mathrm{Per}(g_m)$ (see, for instance, Corollary 4.2 of [16]). ■

2.6 Step 4. Proof of Theorem C

The goal of this section is to perform Step 4 of the strategy described in Section 2.1, while proving Theorem C. In doing this we will use two technical lemmas. The first one shows that the first element of any characteristic of a model (S, P, g) belongs to Σ_S, and the second one studies the unions of the sets $\Lambda(\underline{s}, \tau, l, \lambda)$ corresponding to a fixed sequence $\underline{s}$.

Lemma 2.6.1. *Let $g : S \longrightarrow S$ be a tree map, let P be a periodic orbit of g and let $(\underline{s}, \tau, l, \lambda)$ be a characteristic of a sequence of block reductions of (S, P, g). Then,*

(a) $\underline{s} \in \Sigma_S$

(b) (S 3–4) hold

(c) $\lambda\lceil\underline{s}\rceil \le 2|P|$ *when* $\tau = *$

(d) $l < |\,\mathrm{En}(S)|$.

Proof. We will use the notation from properties (SBR 1–4) and from the definition of a characteristic. In particular, $\underline{s} = (p_1, p_2, \dots, p_m)$.

Let us prove (a). From (SBR 2) it follows that all numbers p_i are positive integers such that $p_i \ge 2$ for $1 \le i < m$. Moreover, p_m is also positive by (SBR 4). Therefore, $\underline{s} \in \Sigma$. On the other hand, (S 2) follows from (SBR 3), and by (SBR 2) we have $S \supset S_1$ and $S_i \sqsupset p_i S_{i+1}$ for $1 \le i < m$. Thus to prove (S 1) we only have to see that $p_m \le |\,\mathrm{En}(S_m)|$. This inequality trivially holds when $p_m = 1$. Hence, by (SBR 4), we may assume that p_m is a type of A^{y_m}, where (T, A, f) is a y_m-expansive canonical model of $[S_m, P_m, g_m]$. Therefore, $p_m \le |\,\mathrm{En}(T)|$. On the other hand, $\mathrm{En}(T) \subset A$ because (T, A, f) is a canonical model. Since $\mathrm{En}(S_m) \subset P_m$ by Remark 2.3.4 and (T, A, f) and (S_m, P_m, g_m) have the same pattern, it follows that $|\,\mathrm{En}(T)| = |\,\mathrm{En}(S_m)|$.

Let us prove (b). Set $e_i = |\,\mathrm{En}(S_i)|$ for $1 \le i \le m$. Since (S 1) holds, $e_{i+1} \le e_i/p_i + 1$ for $1 \le i \le m-1$. Consequently, when $m \ge 3$ we have:

$$
\begin{aligned}
e_{m-1} &\leq \frac{e_1}{p_1p_2\cdots p_{m-2}} + \frac{1}{p_2p_3\cdots p_{m-2}} + \cdots + \frac{1}{p_{m-2}} + 1 \\
&\leq \frac{e_1}{2^{m-2}} + \frac{1}{2^{m-3}} + \cdots + \frac{1}{2} + 1 \\
&= \frac{e_1}{2^{m-2}} + 2(1 - \frac{1}{2^{m-2}}) \leq \frac{1}{2^{m-2}}(|\operatorname{En}(S)| - 2) + 2.
\end{aligned}
$$

Thus, (S 3) follows from the fact that $e_{m-1} \geq 4$ by (S 2).

Now we prove (S 4). Since $e_1 \leq E_1$, it follows from the definition of E_i and a simple inductive argument that $e_i \leq E_i$ for $1 \leq i \leq m$. Consequently, we only have to show that $p_i \leq e_i$ for $1 \leq i \leq m$. For $1 \leq i \leq m-1$ this follows directly from the fact that $(S_{i+1}, P_{i+1}, g_{i+1})$ is a p_i-block reduction of (S_i, P_i, g_i) (see (SBR 2)). And $p_m \leq e_m$ holds by (S 1). This ends the proof of (S 4) and (b).

Now let us prove (c). When $\tau = *$, the model (S_m, P_m, g_m) is non-twist. Hence, by Lemma 2.4.1 (T, A, f) is also non-twist. From Theorem 2.4.2 we get that $\lambda p_m \leq 2|P_m|$. By multiplying this inequality on both sides by $p_1p_2\cdots p_{m-1}$, we have that $\lambda\lceil\underline{\mathsf{s}}\rceil \leq 2p_1p_2\cdots p_{m-1}|P_m|$. Then (c) holds since, by Remark 2.3.4, $|P| = p_1p_2\cdots p_{m-1}|P_m|$.

Finally we prove (d), which obviously holds when $l = 0$. Assume that $l > 0$ (in particular, $\tau = *$). As above, this implies that (T, A, f) is a non-twist model and from Theorem 2.4.2 we get that $lp_m \leq |\operatorname{En}(T)| - p_m = |\operatorname{En}(S_m)| - p_m \leq |\operatorname{En}(S)| - p_m$. By multiplying on both sides by $p_1p_2\cdots p_{m-1}$, we get $l\lceil\underline{\mathsf{s}}\rceil \leq |\operatorname{En}(S)|p_1p_2\cdots p_{m-1} - \lceil\underline{\mathsf{s}}\rceil$. Since $p_1p_2\cdots p_{m-1} \leq \lceil\underline{\mathsf{s}}\rceil$, we get $l\lceil\underline{\mathsf{s}}\rceil \leq |\operatorname{En}(S)|\lceil\underline{\mathsf{s}}\rceil - \lceil\underline{\mathsf{s}}\rceil$ and thus (d) follows. ∎

Lemma 2.6.2. *Let $g : S \longrightarrow S$ be a tree map and let $\{(\underline{\mathsf{s}}, \tau_i, l_i, \lambda_i)\}_{i\in R}$ with $\underline{\mathsf{s}} = (p_1, p_2, \ldots, p_m)$ be a family of characteristics of sequences of block reductions of periodic models (S, P_i, g). Then,*

$$\bigcup_{i\in R} \Lambda(\underline{\mathsf{s}}, \tau_i, l_i, \lambda_i) = \mathcal{K}_{\underline{\mathsf{s}}} \cup \mathcal{F}_{\underline{\mathsf{s}}} \cup (\mathcal{I}_{\underline{\mathsf{s}}} \setminus \lceil\underline{\mathsf{s}}\rceil\{2, 3, \ldots, \lambda_{\underline{\mathsf{s}}}\})$$

where $\lambda_{\underline{\mathsf{s}}}$ is a nonnegative integer and

(a) $\mathcal{I}_{\underline{\mathsf{s}}}$ is an initial segment of the ${}_{\lceil\underline{\mathsf{s}}\rceil}\!\geq$ ordering such that each ${}_{\lceil\underline{\mathsf{s}}\rceil}\!\geq$-maximal element of $\mathcal{I}_{\underline{\mathsf{s}}}$ belongs to $\{1\} \cup p_1p_2\cdots p_{m-1}(\mathbb{N} \cup 2^\infty)$.

(b) If $\mathcal{I}_{\underline{\mathsf{s}}} \subsetneq \{1\} \cup \lceil\underline{\mathsf{s}}\rceil\mathbb{N}$ then $\lambda_{\underline{\mathsf{s}}} = 0$ and $\mathcal{F}_{\underline{\mathsf{s}}} = \emptyset$.

(c) $\mathcal{F}_{\underline{\mathsf{s}}}$ is disjoint from $\mathcal{K}_{\underline{\mathsf{s}}} \cup \mathcal{I}_{\underline{\mathsf{s}}} \setminus \lceil\underline{\mathsf{s}}\rceil\{2, 3, \ldots, \lambda_{\underline{\mathsf{s}}}\}$ and $\mathcal{F}_{\underline{\mathsf{s}}} \setminus \lceil\underline{\mathsf{s}}\rceil\mathbb{N} \subset \bigcup_{j\in B(\mathcal{I}_{\underline{\mathsf{s}}})} \mathbb{N}^j_{\lceil\underline{\mathsf{s}}\rceil}$.

(d) $\mathcal{F}_{\underline{\mathsf{s}}}$ is finite (or empty). When $\mathcal{F}_{\underline{\mathsf{s}}} \neq \emptyset$, we have $\min \mathcal{F}_{\underline{\mathsf{s}}} \geq \lambda_{\underline{\mathsf{s}}}\lceil\underline{\mathsf{s}}\rceil/2$ and $|\mathcal{F}_{\underline{\mathsf{s}}}| < |B(\mathcal{I}_{\underline{\mathsf{s}}})||\operatorname{En}(S)| + (\lambda_{\underline{\mathsf{s}}} + 2)/2$.

Proof. Set

$$\mathcal{I}_{\underline{\mathsf{s}}} := \bigcup_{i\in R} \mathcal{S}^{\tau_i}_{\lceil\underline{\mathsf{s}}\rceil}(|P_i| + l_i\lceil\underline{\mathsf{s}}\rceil).$$

Observe that $\mathcal{S}^{\tau_i}_{\lceil \underline{s} \rceil}(|P_i| + l_i\lceil \underline{s} \rceil)$ is an initial segment of the ordering ${}_{\lceil \underline{s} \rceil}\geq$ for each $i \in R$, and recall that, by definition, the union of initial segments of a given ordering is an initial segment of the same ordering. By definition, $\lceil \underline{s} \rceil \in p_1 p_2 \cdots p_{m-1}\mathbb{N}$. Moreover, by Remark 2.3.4, $|P_i| \in p_1 p_2 \cdots p_{m-1}\mathbb{N}$ for each $i \in R$. Consequently, each ${}_{\lceil \underline{s} \rceil}\geq$-maximal element of $\mathcal{I}$ belongs to $\{1\} \cup p_1 p_2 \cdots p_{m-1}(\mathbb{N} \cup 2^\infty)$ and (a) holds.

Let us prove (b). Set

$$\lambda_{\underline{s}} := \min\{\lambda_i : i \in R\} \text{ and}$$

$$R^* := \{i \in R : \tau_i = *\}.$$

For each $i \in R \setminus R^*$, $l_i = \lambda_i = 0$ by definition. Moreover, since $\tau_i = \phi$, in this case we have

$$\mathcal{S}^{\tau_i}_{\lceil \underline{s} \rceil}(|P_i| + l_i\lceil \underline{s} \rceil) \setminus \lceil \underline{s} \rceil\{2, 3, \ldots, \lambda_i\} = \mathcal{S}_{\lceil \underline{s} \rceil}(|P_i|) \ni |P_i|.$$

Consequently,

$$\begin{aligned}
\bigcup_{i \in R} \Lambda(\underline{s}, \tau_i, l_i, \lambda_i) &= \bigcup_{i \in R} \left(\{|P_i|\} \cup \mathcal{K}_{\underline{s}} \cup \left(\mathcal{S}^{\tau_i}_{\lceil \underline{s} \rceil}(|P_i| + l_i\lceil \underline{s} \rceil) \setminus \lceil \underline{s} \rceil\{2, 3, \ldots, \lambda_i\} \right) \right) \\
&= \mathcal{K}_{\underline{s}} \cup \left(\bigcup_{i \in R^*} \{|P_i|\} \right) \cup (\mathcal{I}_{\underline{s}} \setminus \lceil \underline{s} \rceil\{2, 3, \ldots, \lambda_{\underline{s}}\}) \\
&= \mathcal{K}_{\underline{s}} \cup \mathcal{F}_{\underline{s}} \cup (\mathcal{I}_{\underline{s}} \setminus \lceil \underline{s} \rceil\{2, 3, \ldots, \lambda_{\underline{s}}\}),
\end{aligned}$$

where

$$\mathcal{F}_{\underline{s}} := \left(\bigcup_{i \in R^*} \{|P_i|\} \right) \setminus (\mathcal{K}_{\underline{s}} \cup (\mathcal{I}_{\underline{s}} \setminus \lceil \underline{s} \rceil\{2, 3, \ldots, \lambda_{\underline{s}}\})) \subset \mathbb{N}.$$

Now observe that, since $\{1\} \cup \lceil \underline{s} \rceil\mathbb{N} \subset \mathcal{S}^{*}_{\lceil \underline{s} \rceil}(|P_i| + l_i\lceil \underline{s} \rceil)$, when $i \in R^*$ it follows that $\{1\} \cup \lceil \underline{s} \rceil\mathbb{N} \subset \mathcal{S}^{\tau_i}_{\lceil \underline{s} \rceil}(|P_i| + l_i\lceil \underline{s} \rceil) \subset \mathcal{I}_{\underline{s}}$. Therefore, if $\mathcal{I}_{\underline{s}} \subsetneq \{1\} \cup \lceil \underline{s} \rceil\mathbb{N}$ then $R^* = \emptyset$. Hence, (b) follows from the definitions of $\lambda_{\underline{s}}$ and $\mathcal{F}_{\underline{s}}$.

Now we prove (c). The set $\mathcal{F}_{\underline{s}}$ is disjoint from $\mathcal{I}_{\underline{s}} \setminus \lceil \underline{s} \rceil\{2, 3, \ldots, \lambda_{\underline{s}}\}$ and $\mathcal{K}_{\underline{s}}$ by definition. Since $\mathcal{F}_{\underline{s}} \setminus \lceil \underline{s} \rceil\mathbb{N} \subset \bigcup_{j=1}^{\lceil \underline{s} \rceil - 1} \mathbb{N}^j_{\lceil \underline{s} \rceil}$, it is enough to show that if $(\mathcal{F}_{\underline{s}} \setminus \lceil \underline{s} \rceil\mathbb{N}) \cap \mathbb{N}^j_{\lceil \underline{s} \rceil} \neq \emptyset$ for some $1 \leq j < \lceil \underline{s} \rceil$, then $j \in B(\mathcal{I}_{\underline{s}})$. Assume that there is some $i \in R^*$ such that $|P_i| \in \mathbb{N}^j_{\lceil \underline{s} \rceil}$ for some $1 \leq j < \lceil \underline{s} \rceil$. From the definitions we obtain

$$\mathcal{I}_{\underline{s}} \supset \mathcal{S}^{\tau_i}_{\lceil \underline{s} \rceil}(|P_i| + l_i\lceil \underline{s} \rceil) = \mathcal{S}_{\lceil \underline{s} \rceil}(|P_i| + l_i\lceil \underline{s} \rceil) \supset \{|P_i| + t\lceil \underline{s} \rceil : t \geq l_i\}.$$

Consequently, $\mathcal{I}_{\underline{s}} \cap \mathbb{N}^j_{\lceil \underline{s} \rceil} \neq \emptyset$ because $(|P_i| + t\lceil \underline{s} \rceil) \equiv j \pmod{\lceil \underline{s} \rceil}$ for all t. Thus $j \in B(\mathcal{I}_{\underline{s}})$ and (c) follows.

Finally we prove (d). Assume that $\mathcal{F}_{\underline{s}} \neq \emptyset$. By definition, we have

$$\mathcal{F}_{\underline{s}} \subset \mathbb{N} \setminus (\mathcal{K}_{\underline{s}} \cup (\mathcal{I}_{\underline{s}} \setminus \lceil \underline{s} \rceil \{2, 3, \ldots, \lambda_{\underline{s}}\})) \subset (\mathbb{N} \setminus \mathcal{I}_{\underline{s}}) \cup \lceil \underline{s} \rceil \{2, 3, \ldots, \lambda_{\underline{s}}\}.$$

By Lemma 2.6.1(c), for each $i \in R^*$ we have $|P_i| \geq \lambda_i \lceil \underline{s} \rceil / 2 \geq \lambda_{\underline{s}} \lceil \underline{s} \rceil / 2$. Hence, $\min \mathcal{F}_{\underline{s}} \geq \lambda_{\underline{s}} \lceil \underline{s} \rceil / 2$ and

$$\mathcal{F}_{\underline{s}} \subset (\mathbb{N} \setminus \mathcal{I}_{\underline{s}}) \cup \{n \lceil \underline{s} \rceil : n \geq 2 \text{ and } \lambda_{\underline{s}}/2 \leq n \leq \lambda_{\underline{s}}\}.$$

Since $\mathcal{F}_{\underline{s}} \neq \emptyset$, from (b) it follows that $\mathcal{I}_{\underline{s}} \supset \{1\} \cup \lceil \underline{s} \rceil \mathbb{N}$. From this and (c) we get that

$$\mathcal{F}_{\underline{s}} \subset \bigcup_{j \in B(\mathcal{I}_{\underline{s}})} (\mathcal{F}_{\underline{s}} \cap \mathbb{N}^j_{\lceil \underline{s} \rceil}) \cup \{n \lceil \underline{s} \rceil : n \geq 2 \text{ and } \lambda_{\underline{s}}/2 \leq n \leq \lambda_{\underline{s}}\}.$$

Since $|\{n \lceil \underline{s} \rceil : n \geq 2 \text{ and } \lambda_{\underline{s}}/2 \leq n \leq \lambda_{\underline{s}}\}| \leq (\lambda_{\underline{s}} + 2)/2$, to end the proof of (d) it is enough to show that $|\mathcal{F}_{\underline{s}} \cap \mathbb{N}^j_{\lceil \underline{s} \rceil}| \leq |\operatorname{En}(S)|$ for each $j \in B(\mathcal{I}_{\underline{s}})$.

Fix $j \in B(\mathcal{I}_{\underline{s}})$. Observe that $\mathcal{F}_{\underline{s}} \cap \mathbb{N}^j_{\lceil \underline{s} \rceil} \subset \mathbb{N}^j_{\lceil \underline{s} \rceil} \setminus \mathcal{I}_{\underline{s}}$. Set

$$s_j = \max_{\lceil \underline{s} \rceil \geq} \left(\mathcal{I}_{\underline{s}} \cap \mathbb{N}^j_{\lceil \underline{s} \rceil} \right).$$

Since $\mathcal{I}_{\underline{s}}$ is an initial segment of the ordering ${}_{\lceil \underline{s} \rceil}\geq$, it follows that

$$\mathcal{I}_{\underline{s}} \cap \mathbb{N}^j_{\lceil \underline{s} \rceil} = \{s_j + t \lceil \underline{s} \rceil : t \geq 0\}.$$

Consequently, $\{j, j + \lceil \underline{s} \rceil, j + 2\lceil \underline{s} \rceil, \ldots, s_j - \lceil \underline{s} \rceil\} \supset \mathbb{N}^j_{\lceil \underline{s} \rceil} \setminus \mathcal{I}_{\underline{s}} \supset \mathcal{F}_{\underline{s}} \cap \mathbb{N}^j_{\lceil \underline{s} \rceil}$.

On the other hand, take $i \in R$ such that $|P_i| \in \mathcal{F}_{\underline{s}} \cap \mathbb{N}^j_{\lceil \underline{s} \rceil}$ and

$$|P_i| = \max_{\lceil \underline{s} \rceil \geq} \left(\mathcal{F}_{\underline{s}} \cap \mathbb{N}^j_{\lceil \underline{s} \rceil} \right).$$

Then $|P_i| \;{}_{\lceil \underline{s} \rceil}\geq s_j$ and $\mathcal{F}_{\underline{s}} \cap \mathbb{N}^j_{\lceil \underline{s} \rceil} \subset A_j := \{|P_i|, |P_i| + \lceil \underline{s} \rceil, |P_i| + 2\lceil \underline{s} \rceil, \ldots, s_j - \lceil \underline{s} \rceil\}$. Since $|P_i| + l_i \lceil \underline{s} \rceil \in \mathcal{I}_{\underline{s}}$, it follows that $s_j - \lceil \underline{s} \rceil \;{}_{\lceil \underline{s} \rceil}> |P_i| + l_i \lceil \underline{s} \rceil$. Hence,

$$A_j \subset \{|P_i|, |P_i| + \lceil \underline{s} \rceil, |P_i| + 2\lceil \underline{s} \rceil, \ldots, |P_i| + (l_i - 1)\lceil \underline{s} \rceil\}.$$

Consequently, $|\mathcal{F}_{\underline{s}} \cap \mathbb{N}^j_{\lceil \underline{s} \rceil}| \leq |A_j| \leq l_i$. The lemma follows since $l_i < |\operatorname{En}(S)|$ by Lemma 2.6.1(d). ∎

Proof of Theorem C. Let $\{(\underline{\mathsf{s}}_i, \tau_i, l_i, \lambda_i)\}_{i\in R}$ be the family of all characteristics of sequences of block reductions of periodic orbits of g. For each $i \in R$, denote by P_i the orbit which has generated the characteristic $(\underline{\mathsf{s}}_i, \tau_i, l_i, \lambda_i)$. By definition, $|P_i| \in \Lambda(\underline{\mathsf{s}}_i, \tau_i, l_i, \lambda_i)$ for each i. Therefore,

$$\operatorname{Per}(g) \subset \bigcup_{i\in R} \Lambda(\underline{\mathsf{s}}_i, \tau_i, l_i, \lambda_i) \subset \operatorname{Per}(g)$$

by Theorem 2.5.2.

Set $\mathsf{S} = \bigcup_{i\in R} \underline{\mathsf{s}}_i$. By Lemma 2.6.1 it follows that $\mathsf{S} \subset \Sigma_S$. Also, for each $\underline{\mathsf{s}} \in \mathsf{S}$, we set $R_{\underline{\mathsf{s}}} = \{i \in R : \underline{\mathsf{s}}_i = \underline{\mathsf{s}}\}$. Thus,

$$\operatorname{Per}(g) = \bigcup_{i\in R} \Lambda(\underline{\mathsf{s}}_i, \tau_i, l_i, \lambda_i) = \bigcup_{\underline{\mathsf{s}}\in\mathsf{S}} \left(\bigcup_{i\in R_{\underline{\mathsf{s}}}} \Lambda(\underline{\mathsf{s}}, \tau_i, l_i, \lambda_i) \right),$$

and the theorem holds by Lemma 2.6.2. ■

2.7 Proof of Theorem 2.5.1. Large periods

In this section we are going to prove Theorem 2.5.1 when n is larger than a constant $M(S)$ (defined in page 87) which depends only on the number of endpoints and vertices of the tree S.

We start by outlining the main ideas that we will use in the proof of this fact. Let (S, P, g), (T, A, f) and x be as in the statement of Theorem 2.5.1. If x is an A-significant periodic point of f then, by Theorem 2.2.3(a), there is a unique simple loop β in the $[S, P, g]$-path graph such that x and β are associated. By Theorem 2.2.4, there exists a fixed point z of g^{2n} such that z and β are associated. However, nothing is said about the period of z, which in general can be any divisor of $2n$. In Lemma 2.7.1 we prove that if z is a fixed point of g^n then its period is precisely n. We remark that this is true only when x is A-significant. Therefore, it will be important to check whether the periodic points of f which satisfy the $(\star)$-property correspond to A-significant points. This is done in Proposition 2.7.5, whose proof needs three previous technical lemmas. Proposition 2.7.5 will be also used in Section 2.8.

Lemma 2.7.1. *Let (S, P, g) be a periodic model, and let (T, A, f) be a canonical model of $[S, P, g]$. Let $x \in T$ be an A-significant n-periodic point of f and let β be a simple loop of length n in the $[S, P, g]$-path graph such that x and β are associated. If there exists a point $z \in S$ associated to β such that $g^n(z) = z$, then z is an n-periodic point of g.*

Proof. Unlike in the rest of the chapter, in this proof the subindexes will be considered modulo n and, given $k \in \mathbb{N}$, we take $\{0, 1, \ldots, k-1\}$ as the representatives of the classes of $\mathbb{Z}/k\mathbb{Z}$.

We start by proving that, given a loop $\pi_0 \to \pi_1 \to \ldots \to \pi_{n-1} \to \pi_0$ in the $[S, P, g]$-path graph such that $\pi_i = \pi_j$ for some $0 \leq i, j < n$, the following statement holds:

$$\begin{array}{l} \text{If } \operatorname{Int}(\langle \pi_{i+1} \rangle_T) \cap \operatorname{Int}(\langle \pi_{j+1} \rangle_T) \neq \emptyset \text{ or, equivalently,} \\ \operatorname{Int}(\langle \pi_{i+1} \rangle_S) \cap \operatorname{Int}(\langle \pi_{j+1} \rangle_S) \neq \emptyset \text{ , then } \pi_{i+1} = \pi_{j+1}. \end{array} \tag{2.3}$$

By the definition of the path graph, $\langle f(\pi_i) \rangle_T \supset \langle \pi_{i+1} \rangle_T$ and $\langle f(\pi_i) \rangle_T \supset \langle \pi_{j+1} \rangle_T$. Moreover, if $\operatorname{Int}(\langle \pi_{i+1} \rangle_T) \cap \operatorname{Int}(\langle \pi_{j+1} \rangle_T) \neq \emptyset$ then π_{i+1} and π_{j+1} belong to the same discrete component. Then (2.3) follows immediately from the fact that $\langle f(\pi_i) \rangle_T$ is an interval (since f is A-monotone).

Now let $\pi_0 \to \pi_1 \to \ldots \to \pi_{n-1} \to \pi_0$ be the loop β. Without loss of generality we may assume that $z \in \langle \pi_0 \rangle_S$. Let us show that $z \in \operatorname{Int}(\langle \pi_0 \rangle_S)$. Indeed: on the contrary, $\operatorname{Orb}(z) = P$ and, given a bijection $\theta : P \longrightarrow A$ which preserves discrete components, $\theta(z)$ is a point of A associated to β. Since β is simple and x is also associated to β, from Theorem 2.2.3(b) it follows that x and $\theta(z)$ are f-monotone equivalent, a contradiction with the fact that x is A-significant. In a similar way one can show that $\operatorname{Orb}_g(z) \subset \cup_{i=0}^{n-1} \operatorname{Int}(\langle \pi_i \rangle_S)$.

In order to prove the lemma we assume that $n = lk$ for some $l > 1$ and that $|\operatorname{Orb}_g(z)| = k$. This will lead us to a contradiction. We have $g^i(z) \in \operatorname{Int}(\langle \pi_i \rangle_S)$ for $i \geq 0$. Since $g^i(z) = g^{i+k}(z)$ for each $i \geq 0$, it follows that

$$\begin{array}{c} \operatorname{Int}(\langle \pi_i \rangle_S) \cap \operatorname{Int}(\langle \pi_{i+k} \rangle_S) \neq \emptyset \text{ for each } i \geq 0. \\ \text{Equivalently, } \operatorname{Int}(\langle \pi_i \rangle_T) \cap \operatorname{Int}(\langle \pi_{i+k} \rangle_T) \neq \emptyset \text{ for each } i \geq 0. \end{array} \tag{2.4}$$

For $s = 0, 1, \ldots, k-1$ set $\Pi_s = \{\pi_s, \pi_{s+k}, \ldots, \pi_{s+(l-1)k}\}$ (observe that, for a fixed s, all the basic paths of Π_s belong to the same discrete component).

Now we claim that $|\Pi_s| = l$ for each $0 \leq s < k$. Let us prove it. By considering (if necessary) a shift of β instead of β itself, we may assume that $s = 0$. Thus we must prove that if $i, j \in \{0, 1, \ldots, l-1\}$ and $i \neq j$ then $\pi_{ik} \neq \pi_{jk}$. By considering again a shift of β, we can assume that $i = 0$ and $1 \leq j < l$. We proceed by induction on j. For $j = 1$, we must see that $\pi_0 \neq \pi_k$. If $\pi_0 = \pi_k$ then from (2.3) and (2.4) it follows that $\pi_1 = \pi_{k+1}$, $\pi_2 = \pi_{k+2}, \ldots, \pi_{(l-1)k} = \pi_n$. Thus $\beta = \alpha^l$ with $\alpha = \pi_0 \to \pi_1 \to \ldots \to \pi_k$, contrary to the fact that β is simple. Now assume that we have proved that no basic path $\pi_k, \pi_{2k}, \ldots, \pi_{(j-1)k}$ is π_0 for some $j < l-1$. If $\pi_0 = \pi_{jk}$, then we write $l = qj + r$ for some $q \in \mathbb{N}$ and $0 \leq r < j$. As above, by using $j(q-1)k$ times (2.3) and (2.4) we get $\beta = \alpha^q \gamma$, where α is the loop $\pi_0 \to \pi_1 \to \ldots \to$

$\pi_{jk} = \pi_0$ and γ is $\pi_{qjk} \to \pi_{qjk+1} \to \pi_{qjk+2} \to \ldots \to \pi_{qjk+rk} = \pi_n = \pi_0$. If $r = 0$ then $\beta = \alpha^q$, a contradiction with the fact that β is simple. If $0 < r < j$, since $\pi_0 = \pi_{qjk}$ we can use again rk times (2.3) and (2.4) in order to obtain $\pi_1 = \pi_{qjk+1}, \pi_2 = \pi_{qjk+2}, \ldots, \pi_{rk} = \pi_{jqk+rk} = \pi_n = \pi_0$, contrary to the induction hypotheses. Therefore, $\pi_0 \neq \pi_{jk}$ and the claim is proved.

For $s \in \{0, 1, \ldots, k-1\}$, we define $C_s = \bigcap_{\pi \in \Pi_s} \langle \pi \rangle_T$. Since Π_s consists of l basic paths which belong to the same discrete component and $l \geq 2$, we have $(C_s \setminus A) \cap V(T) \neq \emptyset$ for $0 \leq s < k$. We claim that if $v \in (C_s \setminus A) \cap V(T)$ then $f(v) \in (C_{s+1 \bmod k-1} \setminus A) \cap V(T)$. To see it, choose three different points $a, b, c \in A \cap C_s$ such that $v \in \langle \{a, b, c\} \rangle_T$. Then $\langle \{a, b, c\} \rangle_T$ is a 3-star with central point v. Since the images of a, b, c belong to $C_{s+1 \bmod k-1}$ and A is a periodic orbit, we have that $f(a)$, $f(b)$ and $f(c)$ are three different points contained in a single discrete component and, hence, $\langle \{f(a), f(b), f(c)\} \rangle_T$ is a 3-star. By Remark 1.2.1, $f(v)$ is the central point of $\langle \{f(a), f(b), f(c)\} \rangle_T$ and therefore $f(v) \in V(T)$.

Since $V(T)$ is finite there is a periodic point w of f in $C_0 \cap V(T)$ such that $f^i(w) \in C_{i \bmod k-1} \cap V(T)$ for each $i \geq 0$. It is clear that w is associated to β and, by Theorem 2.2.3, x and w belong to the same f-monotone equivalence class. Since $w \in V(T)$, this is a contradiction with the fact that x is A-significant. ■

The following three technical lemmas are used in the proof of Proposition 2.7.5.

Lemma 2.7.2. *Let (T, A, f) be a model and let x be a periodic point of f whose f-monotone equivalence class contains no points of $A \cup V(T)$. Then, for each $i \geq 0$, the f-monotone equivalence class of $f^i(x)$ contains no points of $A \cup V(T)$. Moreover, if $z \in T$ is f-monotone equivalent to x then $[f^i(x), f^i(z)] \cap (A \cup V(T)) = \emptyset$ for each $i \geq 0$.*

Proof. Assume that there exist $y \in A \cup V(T)$ and $i \geq 0$ such that $f^i(x)$ and y are f-monotone equivalent. Let $n \in \mathbb{N}$ be such that $f^n(x) = x$. Then $x = f^n(x) = f^{n-(i \bmod n)}(f^{i \bmod n}(x)) = f^{n-(i \bmod n)}(f^i(x))$. Thus, by Remark 2.2.2, x is f-monotone equivalent to $f^{n-(i \bmod n)}(y)$ which, by Remark 1.2.1, belongs to $A \cup V(T)$. This contradiction proves the first assertion of the lemma. In particular, $f^i(x)$ does not belong to $A \cup V(T)$ for each $i \geq 0$.

Now we must prove that $[f^i(x), f^i(z)] \cap (A \cup V(T)) = \emptyset$ for each $i \geq 0$. This is obvious when $x = z$. So we assume that $x \neq z$. Let $i \geq 0$. By Remark 2.2.2, $f^i(x)$ and $f^i(z)$ are f-monotone equivalent. Hence, there exists $m \in \mathbb{N}$ such that $f^m(f^i(x)) = f^i(x)$, $f^m(f^i(z)) = f^i(z)$ and $f^m|_{[f^i(x), f^i(z)]}$ is monotone. Assume that there exists $w \in (f^i(x), f^i(z)) \cap (A \cup V(T))$.

Let $<$ be the orientation of $[f^i(x), f^i(z)]$ such that $f^i(x) < f^i(z)$. Since $f^m|_{[f^i(x),f^i(z)]}$ is monotone and $f^i(x)$ and $f^i(z)$ are fixed points of f^m, it follows that $f^m([f^i(x), f^i(z)]) = [f^i(x), f^i(z)]$ and f^m is increasing (with respect to the ordering chosen above). Therefore, $\operatorname{Orb}_{f^m}(w) \subset [f^i(x), f^i(z)]$. By Remark 1.2.1, $\operatorname{Orb}_{f^m}(w) \subset A \cup V(T)$. Then, from the finiteness of $A \cup V(T)$ and the monotonicity of f^m it follows that there is a fixed point $w' \in (A \cup V(T)) \cap (f^i(x), f^i(z))$ of f^m. Therefore, w' and $f^i(x)$ are f-monotone equivalent, in contradiction with the first assertion of the lemma. ∎

Lemma 2.7.3. *Let (T, Q, f) be a Markov model and let x be a periodic point associated to a simple loop β in the Markov f-graph of Q. If there are no points of Q in the f-monotone equivalence class of x then x is Q-significant.*

Proof. We only have to prove that the period of x is minimal within this f-monotone equivalence class. In fact, we will see that the period of any point in the class coincides with the period of x. Let $z \in T$ be f-monotone equivalent to x. By assumption, $x, z \notin Q$ and, by Lemma 2.7.2, for each $k \geq 0$ there exists a Q-basic interval J_k such that $f^k(x), f^k(z) \in \operatorname{Int}(J_k)$. Hence, x and z are associated to the same loop, β. By Lemma 1.3.5, $|\operatorname{Orb}(x)| = |\operatorname{Orb}(z)| = |\beta|$. ∎

Lemma 2.7.4. *Let (T, A, f) be a y-expansive model and let p be a type of A^y. If $1 \leq i \leq p$ then for each $n \in \mathbb{N}$ there exists $w \in (y, x_i]$ such that $f^n(w) = x_{n+i \bmod p}$ and $(y, w) \cap f^{-n}(A^y) = \emptyset$.*

Proof. We proceed by induction on n. Since $f(y) = y$, $f(x_i) \geq x_{i+1 \bmod p}$ and $f|_{[y,x_i]}$ is monotone, the lemma holds for $n = 1$.

Let $n > 1$ and assume that the lemma holds for $n - 1$. Let $w' \in (y, x_i]$ be such that $f^{n-1}(w') = x_{n-1+i \bmod p}$ and $(y, w') \cap f^{-(n-1)}(A^y) = \emptyset$. Since $f^n(y) = y$, $f^n(w') = f(f^{n-1}(w')) = f(x_{n-1+i \bmod p}) \geq x_{n+i \bmod p}$ and $f|_{[y,x_{n-1+i \bmod p}]}$ is monotone, it follows that there exists $w \in (y, w'] \subset (y, x_i]$ such that $f^n(w) = x_{n+i \bmod p}$ and $(y, w) \cap f^{-n}(A^y) = \emptyset$. This completes the induction. ∎

Proposition 2.7.5. *Let (T, A, f) be a non-twist y-expansive linear canonical model. If x is a periodic point of f satisfying the $(\star)$-property then x is A-significant.*

Proof. Let p by a type of A^y. Since A is a periodic orbit, (T, A, f) is 0-orbital, and, since (T, A, f) is a canonical model, $\operatorname{En}(T) \subset A = A^\circ$. Therefore, by Proposition 1.6.5, $\widetilde{\mathcal{E}}(T, A, f) \neq \emptyset$. In particular, $\mathcal{E}(T, A, f) \neq \emptyset$.

Since x satisfies the $(\star)$-property, there exists a simple loop β in the Markov f-graph of A^y which strictly contains a typical loop $I_1 \to I_2 \to$

$\ldots \to I_p \to I_1$. For each $k \in \mathbb{N}$ we define $C^{-k} = f^{-k}(A^y)$. Note that C^{-k} has finitely many connected components and $C^{-k} \supset A^y \supset V(T)$. Since $x \notin A^y$ and x is periodic, $x \notin C^{-k}$ for $k \geq 1$. Observe that if $(a,b) \cap C^{-k} = \emptyset$ then $f^k|_{[a,b]}$ is linear. By Remark 2.2.2, it is enough to prove that some point in the orbit of x is A-significant. Since $I_1 \in \beta$, we can choose a point of $\mathrm{Orb}_f(x)$ which belongs to (y, x_1). Moreover, we can take it to be the point of $\mathrm{Orb}_f(x) \cap (y, x_1)$ closest to y. Thus from now on we assume that $x \in (y, x_1)$ and $(y,x) \cap \mathrm{Orb}_f(x) = \emptyset$. Since β is simple, by Lemma 2.7.3 we only must prove that the f-monotone equivalence class of x does not contain points of A^y. Assume the contrary and let $z \in A^y$ be f-monotone equivalent to x. Thus there exists $n \in \mathbb{N}$ such that $f^n(x) = x$, $f^n(z) = z$ and $f^n|_{[x,z]}$ is monotone (in fact, piecewise linear). By Lemma 2.7.4, there is $w \in (y, x_1]$ such that $f^n(w) = x_{n+1 \bmod p}$ (so $w \in Q^{-n}$) and $Q^{-n} \cap (y,w) = \emptyset$. Since $V(T) \cap (y, x_1) = \emptyset$, it follows that $T \setminus (y, x_1)$ consists of two connected components and z belongs to one of them because $z \in A^y$ and $A^y \cap (y, x_1) = \emptyset$. We consider two cases.

Case $z \leq y < x < x_1$.

Since $f^n|_{[y,x]}$ is monotone and y and x are fixed points of f^n, we have $f^n([y,x]) = [y,x]$. Since $f^n(w) = x_{n+1 \bmod p} \notin [y,x]$, it follows that $w \notin [y,x]$. Thus we have $z \leq y < x < w \leq x_1$. Since $Q^{-n} \cap (y,w) = \emptyset$, f^n is linear on $[y,w]$. Moreover, since y and x are fixed points of f^n on $[y,w]$, f^n is the identity map on $[y,w]$. In particular, $f^n(w) = w$. This implies that $w = x_1$, and this is equivalent to the fact that $f^i(x_1) = x_{1+i \bmod p}$ for each $i \geq 0$. Therefore, the only interval f-covered by a typical interval I_i is $I_{i+1 \bmod p}$. Then $\mathcal{E}(T, A, f) = \emptyset$, a contradiction.

Case $y < x < x_1 \leq z$.

Since $f^n|_{[x,z]}$ is monotone and x and z are fixed points of f^n, we have $f^n([x,z]) = [x,z]$. We claim that $w \notin [x,z]$. Otherwise, $y < x < w$, $Q^{-n} \cap (y,w) = \emptyset$, f^n is linear on $[y,w]$ and y and w are fixed points of f^n. Hence, f^n is the identity map on $[y,w]$. In particular, $f^n(w) = w$. As above, this is equivalent to $\mathcal{E}(T, A, f) = \emptyset$, a contradiction. This proves the claim.

Since $x \notin Q^{-n}$, there exist $a, b \in Q^{-n}$ such that $(a,b) \cap Q^{-n} = \emptyset$ and $w \leq a < x < b$. Hence, f^n is linear on $[a,b]$. Since β is simple and strictly contains the typical loop, it contains a path of the form $I_1 \to I_2 \to \ldots \to I_p \to I_1 \to J$, where J is not typical. By Remark 1.4.5, $\mathrm{Int}(J) \cap I_2 = \emptyset$. Therefore, (y, x_1) contains at least two points of the orbit of x. Let x' be a point of $\mathrm{Orb}_f(x) \cap (y, x_1)$ different from x. Then $x < x' < x_1 \leq z$ by our choice of x. It is clear that x' is a fixed point of f^n. Since $f^n|_{[x,x']}$ is monotone and x and x' are fixed points of f^n, it follows that $f^n([x,x']) = [x,x']$. Thus

$x' < b$, because if $x < b < x'$ then $f^n(b) \in [x, x'] \cap A^y = \emptyset$, a contradiction. Hence, we have $a < x < x' < b \leq x_1 \leq z$. But f^n is linear on $[a, b]$, and x and x' are fixed points of f^n. It follows that f^n is the identity map on $[a, b]$. In particular, $f^n(a) = a$, a contradiction since $f^n(a) \in A^y$ and $A^y \cap (y, x_1) = \emptyset$. ∎

Lemma 2.7.6 will be used in the proof of Theorem 2.5.1 and also in the next section.

Lemma 2.7.6. *Let (S, P, g) be a non-twist model and let (T, A, f) be a y-expansive linear canonical model of $[S, P, g]$. Let x be a periodic point of f satisfying the $(\star)$-property, and let β be a simple loop in the Markov f-graph of A^y associated to x and containing strictly a typical loop. If β is positive then $n \in \mathrm{Per}(g)$.*

Proof. By Lemma 2.4.1, (T, A, f) is non-twist. Then by Proposition 2.7.5 x is A-significant. From Theorem 2.2.3 it follows that there is a unique simple loop $\beta = \pi_0 \to \pi_1 \to \ldots \to \pi_{n-1} \to \pi_n = \pi_0$ in the $[S, P, g]$-path graph, which we call β', such that x and β' are associated. It is not difficult to see that β and β' have the same sign. Thus β' is positive. By Lemma 7.4 of [3], there exists a finite union $J = \bigcup_{i=1}^{m} [a_i, b_i] \subset S$ of intervals with pairwise disjoint interiors such that, if $\langle \pi_0 \rangle_S = [a, b]$, then:

(i) $a \leq a_1 < b_1 \leq a_2 < b_2 \leq \ldots \leq a_m < b_m \leq b$ (where $<$ is an orientation on $[a, b]$).
(ii) $g^j(J) \subset \langle \pi_j \rangle$ for $j = 1, \ldots, n-1$ and $g^n(J) = [a, b]$.
(iii) $g^n|_{\{a_1, b_1, \ldots, a_m, b_m\}}$ is monotone.
(iv) $g^n(b_i) = g^n(a_{i+1})$ for $i = 1, 2, \ldots, m-1$.
(v) $g^n([a_i, b_i]) \subset [g^n(a_i), g^n(b_i)]$.

Since β' is positive, $g^n|_{\{a_1, b_1, \ldots, a_m, b_m\}}$ is increasing. An easy argument, analogous to the one used in Lemma 3 of [9], shows that there exists $z \in J$ such that $g^n(z) = z$, and z and β are associated. By Lemma 2.7.1 the period of z is n, and hence $n \in \mathrm{Per}(g)$. ∎

For any tree S, we define

$$M(S) := \frac{1}{2} |\operatorname{En}(S)| \cdot (|\operatorname{En}(S)| - 1) \cdot |V(S)|^2.$$

Theorem 2.7.7. *Theorem 2.5.1 holds whenever $n > M(S)$.*

Proof. Let K be the number of basic paths π of the pattern $[S, P, g]$ such that $\mathrm{Int}(\langle \pi \rangle) \cap V(S) \neq \emptyset$. Let L be the maximum number of vertices of S

contained in the interior of a basic path. Set $e = |\operatorname{En}(S)|$. Since a vertex of S belongs at most to $\binom{e}{2}$ basic paths, it follows that $K \leq |V(S)|\binom{e}{2}$. Obviously, $L \leq |V(S)|$. Therefore, $KL \leq M(S)$. By Lemma 2.4.1, (T, A, f) is non-twist. Thus from Proposition 2.7.5 it follows that x is A-significant. By Theorem 2.2.3 there is a simple loop $\pi_0 \rightarrow \pi_1 \rightarrow \ldots \rightarrow \pi_{n-1} \rightarrow \pi_0$ in the $[S, P, g]$-path graph, which we call β, such that x and β are associated. Since x satisfies the $(\star)$-property, there exists a simple loop β' in the Markov f-graph of A^y which strictly contains a typical loop and such that x and β' are associated. It is not difficult to see that β and β' have the same sign. If β' is positive, then $n \in \operatorname{Per}(g)$ follows from Lemma 2.7.6 and we are done.

Assume that β' (and thus β) is negative. If there is some $0 \leq i < n$ such that $\operatorname{Int}(\langle\pi_i\rangle) \cap V(S) = \emptyset$, then there exists $z \in \operatorname{Int}(\langle\pi_i\rangle)$ with $g^n(z) = z$. Then, by Lemma 2.7.1, $n \in \operatorname{Per}(g)$ and we are done.

From now on we assume that the interior of each basic path in the loop β contains at least one point of $V(S)$. From the definition of $M(S)$, K and L and the fact that $n > M(S) \geq KL$, it follows that there is a basic path π in the loop β satisfying the following property: if s is the number of occurrences of π in the loop β, and $r = |\operatorname{Int}(\langle\pi\rangle) \cap V(S)| \leq L$, then $s > r$. Assume without loss of generality that $\pi = \pi_0$. By considering the s shifts of β starting at π_0, we have s loops $\pi_0 \rightarrow \pi_1^j \rightarrow \ldots \rightarrow \pi_{n-1}^j \rightarrow \pi_0$ for $1 \leq j \leq s$. Since β is simple, it can be seen (see Lemma 3.3.2) that these loops are pairwise different. Then, by Lemma 3.2 of [3], there exist subsets $J^1, J^2, \ldots, J^s$ of $\langle\pi_0\rangle$ which consist of finite unions of closed intervals such that for each $j = 1, 2, \ldots, s$ we have $J^j \subset \langle\pi_0\rangle$, $g^i(J^j) \subset \langle\pi_i^j\rangle$ for $1 \leq i < n$, and $g^n(J^j) = \langle\pi_0\rangle$. Moreover, $\operatorname{Int}(\langle J^k\rangle) \cap \operatorname{Int}(\langle J^j\rangle) = \emptyset$ for $j, k \in \{1, 2, \ldots, s\}$ with $j \neq k$.

Choose an orientation for $\langle\pi_0\rangle$. Without loss of generality, assume that the sets J^j for $1 \leq j < s$ are labeled in such a way that $x \leq y$ for each pair of points x, y such that $x \in \langle J^j\rangle$ and $y \in \langle J^{j+1}\rangle$. Observe that, given $j \in \{1, 2, \ldots, s\}$, if there are no fixed points of g^n in J^j then there exist $w \in \langle J^j\rangle \cap V(S)$ and $a, b \in J^j$ such that $a < w < b$, $g^n(a) = g^n(b) = w$ and $g^n((a, b)) \cap \langle\pi_0\rangle = \emptyset$. In particular, $w \in \operatorname{Int}(\langle J^j\rangle)$. Since there are r vertices in $\operatorname{Int}(\langle\pi_0\rangle)$ and $s > r$, necessarily there exists $k \in \{1, 2, \ldots, s\}$ such that $\operatorname{Int}(\langle J^k\rangle) \cap V(S) = \emptyset$. Then there is a fixed point z of g^n in J^k. By Lemma 2.7.1, z is an n-periodic point of g and thus we are done. ∎

2.8 Proof of Theorem 2.5.1. Small periods

The aim of this section is to prove Theorem 2.5.1 when $n \leq M(S)$ (thus completing Step 3 of the strategy defined in Section 2.1). The method and the tools used in the proof of this case will be completely different from the ones used previously. Essentially they will come from the theory of patterns and minimal dynamics for graph maps. More exactly, we will use Theorem A of [2], which studies the persistence of patterns among all the graph maps being topological representatives of a fixed free group endomorphism.

2.8.1 General definitions and preliminary results

This subsection is devoted to recall some notions which are necessary in order to state and use Theorem A of [2]. It is organized into three sub-subsections.

Graphs, models and homotopies of pointed graphs

A *graph* is a connected Hausdorff space G which is the union of finitely many subspaces G_i, each being homeomorphic to a non-degenerate closed interval of the real line, and $G_i \cap G_j$ is finite for each $i \neq j$. It is not difficult to see that any tree is a graph. Next we extend some basic notions (*vertex*, *edge* and *endpoint*) from trees to graphs. The points of G which have no neighborhood homeomorphic to an open interval of the real line will be called *vertices*, and the set of vertices of G will be denoted by $V(G)$. This set is obviously finite (or empty). The closure of each connected component of $G \setminus V(G)$ is called an *edge* of G. A graph has finitely many edges, each of which is homeomorphic either to a closed interval or to the circle. If $v \in V(G)$ and there is exactly one edge of G containing v, then v will be called an *endpoint* of G. The set of endpoints of G will be denoted by $\mathrm{En}(G)$. Every subset of G homeomorphic to a non-empty closed (respectively open) interval of the real line will be called a *closed* (respectively *open*) *interval*. A continuous map from a graph into itself will be called a *graph map*.

Let G and G' be graphs and let $I \subset G$ be an open interval. Let $f : \mathrm{Cl}(I) \longrightarrow G'$ be continuous and let $x \in I$. We will say that f is *locally monotone at x* if there is a neighborhood U of the connected component of $f^{-1}(f(x))$ containing x such that $f(U)$ is homeomorphic to an interval of the real line and $f|_U$ is monotone as an interval map. We will say that f is *monotone* if it is locally monotone at each point of $\mathrm{Cl}(I)$. Observe that, even when the connected component of $f^{-1}(f(x))$ containing x reduces to $\{x\}$ for all $x \in I$, f is not necessarily injective on I, since the image of f can turn several times around a circle. If I and $f(\mathrm{Cl}(I))$ are oriented, then we will say that f is *increasing* (respectively *decreasing*) when f is orientation-preserving (respectively orientation-reversing).

A *pointed graph* is a couple (G, P) where G is a graph and P is a finite subset of G. A triplet (G, P, f) will be called a *graph model* if $f : G \longrightarrow G$ is a graph map and P is a finite f-invariant set. Note that if G is a tree then (G, P, f) is a *model* according to the definition given in Section 1.2. A graph model (G, P, f) will be called *Markov* if the orbit of each vertex of G is finite and f restricted to any connected component of $G \setminus (P \cup V(G))$ is monotone. We remark that if G is a tree then (G, P, f) is a Markov model according to the definition given in Section 1.3.

As usual, we will write $f \simeq g$ to indicate that two maps f and g are homotopic.

Given pointed graphs (G, P) and (G', Q), we say that $f\colon G \longrightarrow G'$ is a *pointed graph map*, written $f\colon (G, P) \longrightarrow (G', Q)$, if f is continuous and $f(P) \subset Q$. Let $f, g\colon (G, P) \longrightarrow (G', Q)$ be pointed graph maps. We recall that *f is homotopic to g relative to P*, written $f \simeq_P g$, if there exists a continuous family of pointed graph maps $h_t\colon (G, P) \longrightarrow (G', Q)$ with parameter t ranging over $[0, 1]$, and satisfying $h_0 = f$ and $h_1 = g$. In particular, $f|_P = h_t|_P = g|_P$ for all $t \in [0, 1]$.

The pointed graphs (G, P) and (G', Q) will be said to *have the same homotopy type* if there exist maps $r\colon (G, P) \longrightarrow (G', Q)$ and $s\colon (G', Q) \longrightarrow (G, P)$ such that $r \circ s \simeq_Q \mathrm{Id}_{G'}$ and $s \circ r \simeq_P \mathrm{Id}_G$. The maps r and s will be called *homotopy equivalences of pointed graphs.* Notice that if (G, P) and (G', Q) have the same homotopy type then $|P| = |Q|$ and the ranks of the fundamental groups of G and G' coincide. In fact it can be shown that the converse is also true.

Paths and groupoids

Given a graph G and $x, y \in G$, a *path from x to y* will be a continuous map $\sigma : [0, 1] \longrightarrow G$ such that $\sigma(0) = x$ and $\sigma(1) = y$. The points x and y will be called the *endpoints* of σ. The path $\sigma(1 - t)$ from y to x will be denoted by σ^{-1}. A path which begins and ends at the same point will be called a *loop*. Given two paths σ and τ such that $\sigma(1) = \tau(0)$, we denote their *concatenation* by $\sigma\tau$.

Let (G, P) be a pointed graph. Let σ and τ be two paths in G whose endpoints belong to P. We say that σ and τ are *equivalent* if σ is homotopic to τ relative to the endpoints. This defines an equivalence relation on the set of all paths in G whose endpoints belong to P. Let $\pi(G, P)$ denote the resulting quotient space, and let $[\sigma]$ denote the equivalence class of σ. The concatenation of paths induces a well-defined natural product on $\pi(G, P)$, defined by $[\sigma] \cdot [\tau] = [\sigma\tau]$. We also set $[\sigma]^{-1} = [\sigma^{-1}]$. This equips $\pi(G, P)$ with a groupoid structure (see subsection 2.2 of [2] for further details). We remark that $\pi(G, P)$ has $|P|$ trivial elements which are the classes of the trivial loops based at the points of P. Moreover, there exists a finite subset

$\mathcal{P}$ of $\pi(G,P)$ with the property that any element of $\pi(G,P)$ may be written in a unique way (without cancellation) as a product of elements of $\mathcal{P}$ (and their inverses). This set is called a *free system of generators of* $\pi(G,P)$. Any groupoid morphism is determined by its effect on a free system of generators. Moreover, a morphism $\phi\colon \pi(G,P) \longrightarrow \pi(G',P')$ is an isomorphism if and only if it induces a bijective map from a free system of generators of $\pi(G,P)$ to a free system of generators of $\pi(G',P')$.

Any pointed graph map $f\colon (G,P) \longrightarrow (G',Q)$ induces a groupoid morphism $f^*\colon \pi(G,P) \longrightarrow \pi(G',Q)$, defined by $f^*([\sigma]) = [f \circ \sigma]$ for all $[\sigma] \in \pi(G,P)$. It is not difficult to see that if there exist two homotopy equivalences $r\colon (G,P) \longrightarrow (G',Q)$ and $s\colon (G',Q) \longrightarrow (G,P)$ such that $r \circ s \simeq_Q \mathrm{Id}_{G'}$ and $s \circ r \simeq_P \mathrm{Id}_G$ (that is, (G,P) and (G',Q) have the same type of homotopy), then r^* and s^* are isomorphisms and $(r^*)^{-1} = s^*$.

Graph patterns, Nielsen fixed point classes and Theorem A of [2]

Two graph models (G,P,f) and (G',Q,g) will be said to *have the same graph pattern* if there exists a homotopy equivalence $r\colon G \longrightarrow G'$ such that r maps P bijectively onto Q and the following diagram:

$$\begin{array}{ccc} (G,P) & \xrightarrow{\ r\ } & (G',Q) \\ {\scriptstyle f}\big\downarrow & & \big\downarrow{\scriptstyle g} \\ (G,P) & \xrightarrow{\ r\ } & (G',Q) \end{array}$$

commutes up to homotopy relative to P. In other words, $g \circ r \simeq_P r \circ f$. This defines an equivalence relation on the set of graph models. The resulting equivalence class, or *graph pattern*, of (G,P,f) will be denoted by $[G,P,f]$. Observe that if $(G,P,f),(G',Q,g)$ have the same graph pattern then $|P| = |Q|$ and the ranks of the fundamental groups of G and G' coincide.

Remark 2.8.1. Let T,S be trees and let (T,P,f), (S,Q,g) be periodic models. From the fact that the fundamental group of any tree is trivial, it can be shown that (T,P,f) and (S,Q,g) have the same graph pattern if and only if $|P| = |Q|$. Therefore, the notion of *graph pattern* differs greatly from the notion of pattern which was defined at Section 2.2 for tree maps. In fact, in order to recover the different specific notions of pattern which can be found in the literature, it suffices to specify the hypotheses on the map r in such a way that the desired properties are preserved. In our framework, the homotopy type of the space will be preserved and the hypothesis is just that r be a homotopy equivalence. If one wants to preserve the space itself, r must be a homeomorphism, as in the case of the interval [38] or of fixed graphs [11]. We recall that for patterns of trees one wants to preserve the discrete components of the pointed tree, and this is the condition that must be satisfied by r in that setting. $\square$

Let (G, P, f) and (G', Q, g) be graph models such that P and Q are periodic orbits. We will say that $[G', Q, g]$ is a *reduction* of $[G, P, f]$ if $|P| > |Q|$ and there exists a homotopy equivalence $r\colon G \longrightarrow G'$ such that $r(P) = Q$ and the following diagram:

$$\begin{array}{ccc} (G,P) & \xrightarrow{\ r\ } & (G',Q) \\ {\scriptstyle f}\big\downarrow & & \big\downarrow{\scriptstyle g} \\ (G,P) & \xrightarrow{\ r\ } & (G',Q) \end{array}$$

commutes up to homotopy relative to P. This definition does not depend on the choice of representative of the patterns. Observe also that it differs from the definition of pattern since $r|_P$ need not be injective here. A pattern which admits a reduction will be called *reducible*, and *irreducible* otherwise. It can be shown (see Proposition 3.3 of [2]) that a pattern $[G, P, f]$ is reducible if and only if there exists $m < |P|$ with $|P| = qm$, for some $q > 1$, such that for any $x \in P$ there exists a path γ from x to $f^m(x)$ satisfying:

$$[\gamma(f^m \circ \gamma) \dots (f^{(q-1)m} \circ \gamma)] = [\sigma_x],$$

where σ_x denotes a trivial loop based at x.

Remark 2.8.2. Let T, S be trees and let (T, P, f), (S, Q, g) be periodic models. In view of Remark 2.8.1, one easily gets that $[S, Q, g]$ is a reduction of $[T, P, f]$ (as graph patterns) if and only if $|Q|$ strictly divides $|P|$. $\square$

Let G be a graph, and let x and y be fixed points of a graph map $f\colon G \longrightarrow G$. We say that x and y are *equivalent* if there exists a path γ from x to y such that $f \circ \gamma$ is equivalent to γ (recall that this means that $f \circ \gamma$ is homotopic to γ keeping endpoints fixed). This defines an equivalence relation on the set of fixed points of f, and the corresponding equivalence classes are called *(Nielsen) fixed point classes*. The fixed point class of x will be denoted by $[x, f]$, and $\operatorname{ind}[x; f]$ will denote the index of $[x, f]$ with respect to f. That is, $\operatorname{ind}[x; f] := \operatorname{ind}([x, f], f)$ (see [33] for further details). If this index is different from 0 then $[x, f]$ will be called an *essential* fixed point class. If P is a periodic orbit of f and n is a multiple of $|P|$, then we define the *index of* P with respect to f^n, denoted by $\operatorname{ind}[P; f^n]$, to be the integer $\operatorname{ind}[x; f^n]$ for each $x \in P$. It can be shown that this number does not depend on the chosen point $x \in P$ and thus the index of a periodic orbit is well defined. Finally P will be called an *essential* periodic orbit if $\operatorname{ind}[P; f^{|P|}] \neq 0$. Now we are ready to state Theorem A of [2].

Theorem 2.8.3 (Theorem A of [2]). *Let $f\colon G \longrightarrow G$ and $g\colon G' \longrightarrow G'$ be graph maps such that there exist homotopy equivalences $r\colon G \longrightarrow G'$ and $s\colon G' \longrightarrow G$ satisfying $r \circ s \simeq \mathrm{Id}_{G'}$, $s \circ r \simeq \mathrm{Id}_G$ and $f \simeq s \circ g \circ r$. Then:*

(a) there exists an index-preserving bijection κ that, for each $n \in \mathbb{N}$, sends essential fixed point classes of f^n to essential fixed point classes of g^n.

(b) let P be an essential periodic orbit of f, let C be the fixed point class for $f^{|P|}$ of a point of P, and let Q be the g-orbit of a point of $\kappa(C)$. Then either $[G', Q, g] = [G, P, f]$, or $[G', Q, g]$ is a reduction of $[G, P, f]$.

Remark 2.8.4 (The case of tree maps). Since any tree has a trivial homotopy, it easily follows that for each pair of tree maps $f\colon T \longrightarrow T$ and $g\colon S \longrightarrow S$ we have $f \simeq g$. Moreover, for any $n \in \mathbb{N}$, all fixed points of f^n are pairwise equivalent. Thus for any $n \in \mathbb{N}$ there is only one Nielsen class of fixed points of f^n. Recall also (Remark 2.8.1 and Remark 2.8.2) that if (T, P, f) and (S, Q, g) are periodic models then $[S, Q, g]$ is a reduction of $[T, P, f]$ if and only if $|Q|$ strictly divides $|P|$, and $[T, P, f] = [S, Q, g]$ if and only if $|P| = |Q|$. Taking it all into account, we easily get the following particular instance of Theorem 2.8.3 for tree maps: *Let $f\colon G \longrightarrow G$ and $g\colon G' \longrightarrow G'$ be tree maps. Then:*

(a) Let $n \in \mathbb{N}$, let C be the unique Nielsen class of fixed points of f^n and let C' be the unique Nielsen class of fixed points of g^n. Then $\mathrm{ind}(C) = \mathrm{ind}(C')$.

(b) Let P be a periodic orbit of f and let C be the unique Nielsen class of fixed points of $f^{|P|}$. If C is essential and Q is the g-orbit of a fixed point of $g^{|P|}$, then $|Q|$ divides $|P|$.

Observe that statement (b) in the case of trees does not provide any information. However, from statement (a) it follows that *for each tree map f and each $n \in \mathbb{N}$, the index of the unique Nielsen fixed point class of f^n is -1.* Indeed, consider a tree consisting of a single point, and let f be the identity map on it. Then, for each $n \in \mathbb{N}$ the index of the unique Nielsen class of fixed points of f^n is -1 (see Chapter 1 of [33] on how to compute the index of a fixed point; here we follow [2], where the considered index is minus that defined in [33]). □

2.8.2 Strategy of the proof of Theorem 2.5.1

In order to motivate the need of the introduction of further technical notions and partial results, next we are going to sketch the proof of Theorem 2.5.1. In sake of clarity, we first recall the hypotheses of the theorem: *Let (S, P, g) be a non-twist model and let (T, A, f) be a y-expansive linear canonical model of $[S, P, g]$. Let $n \in \mathbb{N}$ be such that f has an n-periodic point x satisfying* *the* ($\star$)*-property (that is,* $\mathrm{Orb}(x) \cap A^y = \emptyset$ *and* x *is associated to a unique simple loop in the Markov* f*-graph of* A^y *which strictly contains a typical loop).* We must prove that $n \in \mathrm{Per}(g)$. By Lemma 2.7.6 and Theorem 2.7.7, it is enough to prove this fact when the loop associated to x is negative and $n \leq M(S)$.

We will divide the proof into four stages. To carry out Stages 1, 2 and 3 we shall need some partial results, which will be stated and proved into three respective subsections.

Stage 1 (reduction to a Markov case). We will prove (Proposition 2.8.5) that for each $N \geq |P|$ there exists a Markov linear model $(S, \overline{P}, \overline{g})$ such that $P \cup V(S) \subset \overline{P}$, $\mathrm{Per}(\overline{g}) \cap \{1, 2, \dots, N\} = \mathrm{Per}(g) \cap \{1, 2, \dots, N\}$ and $\overline{g}|_P = g|_P$. Thus, the patterns $[S, P, g]$ and $[S, P, \overline{g}]$ coincide and (T, A, f) is also a canonical model of $[S, P, \overline{g}]$. By taking $N > M(S)$ we have that $n \leq N$ and, hence, it is enough to show that $n \in \mathrm{Per}(\overline{g})$. The advantage of working with $(S, P, \overline{g})$ instead of (S, P, g) is obvious, since we can use the loops of the Markov $\overline{g}$-graph of $\overline{P}$ to calculate periodic orbits of $\overline{g}$ and, in addition, $\overline{g}$ is piecewise linear.

Stage 2 (completion to graph models). As explained in Remark 2.8.4, the fact that the fundamental group of any tree is trivial implies that we cannot use directly Theorem 2.8.3 to obtain significant results of persistence of (graph) patterns (and thus of periods) for tree maps. To overcome this problem, we will proceed as follows. Consider pointed graphs (T^G, A) and (S^G, P) obtained by attaching $|A|$-many pairwise disjoint circles to T (respectively S), each circle being attached at a point of A (respectively P –see Figure 2.4 for an example). Then we will construct (Proposition 2.8.11) graph maps $\mathbf{f}: T^G \longrightarrow T^G$ and $\mathbf{g}: S^G \longrightarrow S^G$ and homotopy equivalences $r: T^G \longrightarrow S^G$ and $s: S^G \longrightarrow T^G$ such that:

(i) $r \circ s \simeq \mathrm{Id}_{S^G}$, $s \circ r \simeq \mathrm{Id}_{T^G}$ and $\mathbf{f} \simeq s \circ \mathbf{g} \circ r$. Thus $\mathbf{f}$ and $\mathbf{g}$ satisfy the hypotheses of Theorem 2.8.3.

(ii) $(T^G, A^y, \mathbf{f})$ is a Markov model with $\mathbf{f}|_{A^y} = f|_{A^y}$. Therefore, the Markov f-graph of A^y is a subgraph of the Markov $\mathbf{f}$-graph of A^y.

(iii) $(S^G, \overline{P}, \mathbf{g})$ is a Markov model with $\mathbf{g}|_{\overline{P}} = \overline{g}|_{\overline{P}}$. Therefore, the Markov $\overline{g}$-graph of $\overline{P}$ is a subgraph of the Markov $\mathbf{g}$-graph of $\overline{P}$.

Stage 3 (n is a period of the completion of $(S, \overline{P}, \overline{g})$). By hypotheses, there is a unique simple negative loop β in the Markov f-graph of A^y associated to $\mathrm{Orb}_f(x)$. By (ii), β is also a loop of the Markov $\mathbf{f}$-graph of A^y. In Proposition 2.8.15 we will show that this loop gives rise to an essential

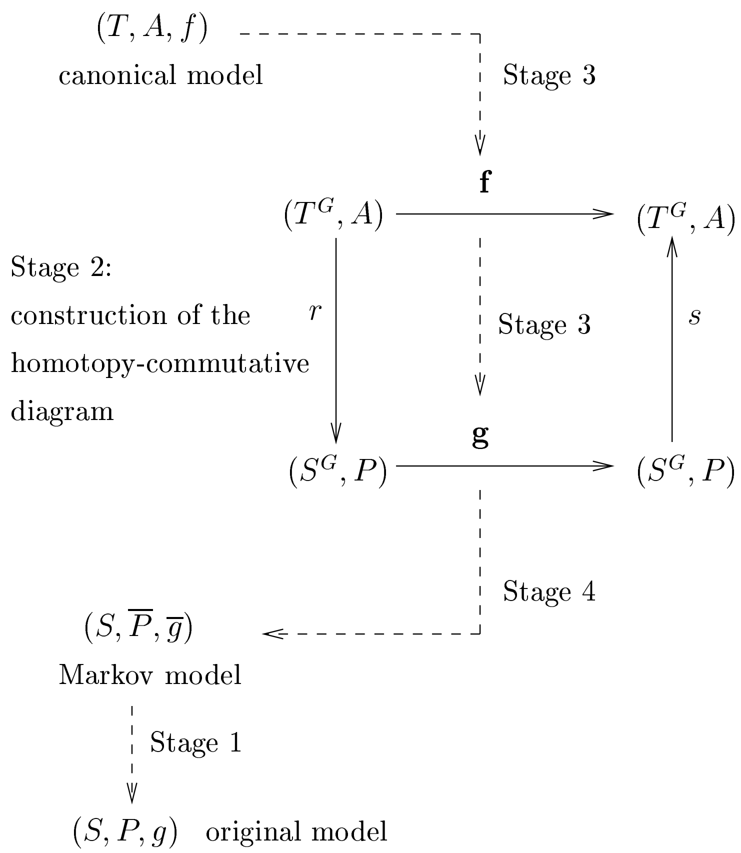

Figure 2.3: The strategy of the proof of Theorem 2.5.1.

periodic orbit Q of $\mathbf{f}$ such that $|Q| = |\operatorname{Orb}_f(x)| = n$ and the graph pattern $[T^G, Q, \mathbf{f}]$ is irreducible. Therefore, by (i) we can use Theorem 2.8.3(b) and we obtain a periodic orbit R of $\mathbf{g}$ such that $[S^G, R, \mathbf{g}] = [T^G, Q, \mathbf{f}]$. In particular, $|R| = |Q| = n$.

Stage 4 (comparing periodic orbits of $(S, \overline{P}, \overline{g})$ and its completion). We will show that R is associated to a simple negative loop of the Markov $\mathbf{g}$-graph of $\overline{P}$ which is also a loop of the Markov $\overline{g}$-graph of $\overline{P}$. Finally we will prove that this loop gives rise to an n-periodic orbit of $\overline{g}$.

In Figure 2.3 a scheme of the proof of Theorem 2.5.1 is shown. In this figure the symbol $f \dashrightarrow g$, where f and g are maps, stands for "we know that f has an n-periodic orbit and we prove that, in consequence, g has an n-periodic orbit".

2.8.3 Stage 1: reduction to a Markov case

The following proposition is the main tool used in Stage 1.

Proposition 2.8.5. *Let (T, D, f) be a model and let $N \geq |D|$, $N \in \mathbb{N}$. Then there exists a Markov linear model $(T, \overline{D}, \overline{f})$ with $D \subset \overline{D}$ such that* $\mathrm{Per}(\overline{f}) \cap \{1, 2, \dots, N\} = \mathrm{Per}(f) \cap \{1, 2, \dots, N\}$ *and* $\overline{f}|_{D \cup V(T)} = f|_{D \cup V(T)}$.

Proof. Let $m \geq N$ be an integer larger than

$$\max\{|\operatorname{Orb}_f(x)| : \operatorname{Orb}_f(x) \text{ is finite and } x \in D \cup V(T)\}.$$

For each $j \in \mathrm{Per}(f) \cap \{1, 2, \dots, m\}$ take a j-periodic point x_j and set

$$D' = D \cup \left(\cup_{j \in \mathrm{Per}(f) \cap \{1,2,\dots,m\}} \operatorname{Orb}_f(x_j)\right).$$

Assume that there exists a tree map $g : T \longrightarrow T$ verifying:

(g.1) Every element of $V(T)$ has a finite orbit,

(g.2) $\mathrm{Per}(g) \cap \{1, 2, \dots, m\} = \mathrm{Per}(f) \cap \{1, 2, \dots, m\}$ and

(g.3) $g|_{D' \cup V(T)} = f|_{D' \cup V(T)}$.

Then we take $\overline{D} = D' \cup (\cup_{v \in V(T)} \operatorname{Orb}_g(v))$ and define $\overline{f} : T \longrightarrow T$ to coincide with g on $\overline{D}$ and being linear on each $\overline{D}$-basic interval. It is not difficult to see that $\mathrm{Per}(\overline{f}) \subset \mathrm{Per}(g)$. Then, from (g.2), (g.3) and the way D' was defined, it easily follows that $\mathrm{Per}(\overline{f}) \cap \{1, 2, \dots, m\} = \mathrm{Per}(f) \cap \{1, 2, \dots, m\}$ and $\overline{f}|_{D' \cup V(T)} = f|_{D' \cup V(T)}$. Then the proposition follows since $m \geq N$. The rest of the proof is devoted to show that such a map g exists. To this end we introduce the following notation. Given a point $x \in T$ we shall denote by $K_f(x)$ the connected component of $f^{-1}(f(x))$ containing x. Then we set $C_f = \bigcup_{x \in D' \cup V(T)} K_f(x)$ and $C_f^n = \bigcup_{i=0}^n f^i(C_f)$ for each $n \geq 0$. Observe that C_f has finitely many connected components and the f-image of each of them is degenerate to a point. Hence, $C_f^n \setminus C_f$ is finite (thus C_f^n has also finitely many connected components) and C_f^n is closed.

We start by proving that if there exists some $v \in V(T)$ such that $\operatorname{Orb}_f(v)$ is not finite, then there exists some $w \in V(T)$ such that $\operatorname{Orb}_f(w)$ is not finite and $f^k(w) \notin C_f^m$ for all $k > m$. If $f^k(v) \notin C_f^m$ for all $k > m$ then we are done by taking $w = v$. Otherwise, there exists $j_0 > m$ with $f^{j_0}(v) \in C_f^m$ and $f^k(v) \notin C_f^m$ for all $k > j_0$. Since D' is a finite f-invariant set, $f^{j_0}(v) \in f^m(V(T))$ and thus, there exists $w \in V(T)$ with $f^{j_0}(v) = f^m(w)$. Hence $\operatorname{Orb}_f(f^m(w)) = \operatorname{Orb}_f(f^{j_0}(v))$ and $f^k(w) \notin C_f^m$ for all $k > m$.

Set $d_f = |\{v \in V(T) : \operatorname{Orb}_f(v) \text{ is infinite}\}|$. If $d_f = 0$ then we set $g = f$ and we are done. Assume that $d_f > 0$. Then, there exists $v \in V(T)$ such that $\operatorname{Orb}_f(v)$ is not finite and $f^k(v) \notin C_f^m$ for all $k > m$. We shall prove that there exists a map $f_1 : T \longrightarrow T$ which verifies:

(g.1') $\text{Orb}_{f_1}(v)$ is finite,
(g.2') $\text{Per}(f_1) \cap \{1, 2, \ldots, m\} \subset \text{Per}(f) \cap \{1, 2, \ldots, m\}$ and
(g.3') $f_1|_{C_f^m} = f|_{C_f^m}$.

Notice that $d_{f_1} < d_f$. Indeed, let $w \in V(T)$ have finite f-orbit. Since $|\text{Orb}_f(w)| < m$, $\text{Orb}_f(w) \subset C_f^m$ and hence $\text{Orb}_f(w) = \text{Orb}_{f_1}(w)$ by (g.3'). Thus $d_{f_1} \leq d_f$. Since $\text{Orb}_{f_1}(v)$ is finite while $\text{Orb}_f(v)$ is not, it follows that $d_{f_1} < d_f$. On the other hand, since $D' \subset C_f^m$, as above we easily get that $\text{Per}(f_1) \cap \{1, 2, \ldots, m\} = \text{Per}(f) \cap \{1, 2, \ldots, m\}$. Therefore, by using iteratively this argument at most d_f times, one can obtain the desired map g satisfying (g.1–3).

Let M be a connected component of $T \setminus C_f^m$ containing infinitely many points from $\text{Orb}_f(v)$. Since $V(T) \subset C_f^m$, M must be an interval. Let $c \in \text{Cl}(M)$ be an accumulation point of $\text{Orb}_f(v) \cap M$. In the rest of the proof we consider four cases. The simplest one is the case when c is not a periodic point of period smaller than or equal to m. The other three cases deal with this possibility.

Case 1. $c \notin \{f(c), f^2(c), \ldots, f^m(c)\}$.

There exists an open interval $I \subset M$ having c as endpoint, containing infinitely many elements of $\text{Orb}_f(v)$ accumulating at c and $f^i(I) \cap I = \emptyset$ for $1 \leq i \leq m$. We set

$$\begin{aligned} r &= \min\{i \in \mathbb{N} : f^i(v) \in I\}, \text{ and} \\ s &= \min\{i \in \mathbb{N} : i > r \text{ and } f^i(v) \in (c, f^r(v))\}, \end{aligned}$$

and we define the map $f_1 : T \to T$ by:

(1.i) $f_1|_{T \setminus (c, f^r(v))} = f|_{T \setminus (c, f^r(v))}$ (in particular it follows that $f_1^i(v) = f^i(v)$ for $i = 0, 1, \ldots, s$).
(1.ii) $f_1(x) = f_1(f^r(v))$ for all $x \in [f^s(v), f^r(v)]$ (in particular, $f_1^{s+1}(v) = f_1(f^s(v)) = f_1(f^r(v)) = f_1^{r+1}(v)$).
(1.iii) $f_1|_{[c, f^s(v)]}$ is linear.

Observe that f_1 is well defined and continuous. Moreover, (g.3') follows from (1.i) and (g.1') follows from (1.i) and (1.ii). Let Q be a periodic orbit of f_1 of period $j \notin \text{Per}(f)$. Since $f_1|_{T \setminus I} = f|_{T \setminus I}$ it follows that $Q \cap I \neq \emptyset$. But, since $f^i(I) \cap I = \emptyset$ for $1 \leq i \leq m$, it follows that $j > m$. This proves (g.2').

In the rest of the proof we assume that c is a periodic point of f of period $n \leq m$ (in particular, $c \in \{f(c), f^2(c), \ldots, f^m(c)\}$).

For each $i \in \{0, 1, \ldots, n\}$ we choose an open ball B^i of $f^i(c)$ (in T) such that

- $B^0 \subset B^n$,
- $\mathrm{Cl}(B^i) \cap (\mathrm{Orb}_f(c) \cup V(T)) = \{f^i(c)\}$ (in particular $\mathrm{Cl}(B^i)$ is a star with $f^i(c)$ as the central point),
- $f(B^i) \subset B^{i+1}$ whenever $i < n$,
- if B^i intersects C_f^m then it intersects a unique connected component of C_f^m and this component contains $f^i(c)$ (recall that C_f^m is closed and has finitely many connected components).

A connected component of $B^i \setminus \{f^i(c)\}$ will be called a *branch of* B^i. Any branch of B^i will be called *proper* if it contains infinitely many points from $\mathrm{Orb}_f(v)$ accumulating at $f^i(c)$. We can take the balls B^i small enough so that, in addition to the above properties, they satisfy

- each branch is either proper or does not intersect $\mathrm{Orb}_f(v)$.

We claim that

$$\text{any proper branch is disjoint from } C_f^m. \tag{2.5}$$

To see this observe that, by the definition of C_f^m and the choice of the balls B^i, if a branch I of B^i intersects C_f^m then it must intersect a component with non-empty interior which has the form $K_f(x) \ni f^i(c)$ with $x \in D' \cup V(T)$ (in particular, $I \cap K_f(x)$ is an interval having $f^i(c)$ as an endpoint). If I is proper, $I \cap K_f(x)$ contains an iterate of v, $f^l(v)$. Then, $f(f^l(v)) = f(f^i(c))$ and, hence, $f^{l+1}(v)$ is periodic (consequently, $\mathrm{Orb}_f(v)$ is finite); a contradiction. This proves the claim.

Case 2. *For some $0 \le i \le n$, there exist a proper branch B^i and $z \in B^i$ such that $f(z) = f^{i+1}(c)$.*

Let I denote the open interval having $f^i(c)$ and z as endpoints. Since z is contained in a proper branch there exist points of $\mathrm{Orb}_f(v)$ in I. Now we take the map $f_1 : T \to T$ such that $f_1|_{T\setminus I} = f|_{T\setminus I}$ and $f_1(x) = f^{i+1}(c)$ for all $x \in I$. As above f_1 is continuous and verifies (g.1') by construction. Also, (g.2') holds because $\mathrm{Per}(f_1) \subset \mathrm{Per}(f)$. On the other hand, I is disjoint from C_f^m by (2.5). Hence, (g.3') holds. This ends the construction of the map f_1 in this case.

From Case 2 we see that, in addition to the above hypotheses, in the rest of the proof we may assume that the image of any proper branch does not intersect $\mathrm{Orb}_f(c)$. In this situation, a branch intersecting the image of a proper branch is also proper and hence, each B^i has a proper branch because so does B^0. Moreover, since there are finitely many proper branches, there exist an open interval J with a point of $\mathrm{Orb}_f(c)$ as an endpoint and a positive integer k such that $f^i(J)$ is contained in a proper branch for $i = 0, 1, \ldots, kn$,

$f^i(J) \cap J = \emptyset$ for $i = 0, 1, \dots, kn-1$ and $f^{kn}(J) \cap J \neq \emptyset$. Relabeling, if necessary, we may assume that J is contained in B^0 (so, J has c as an endpoint). Recall that $f^{kn}(c) = c$ because c is a periodic point of f of period n.

Case 3. *There exists a point $b \in J$ such that $f^{kn}(b) \in (c, b]$.*

We set $W_i = (f^i(c), f^i(b)) \subset f^i(J)$ for $i = 0, 1, \dots, kn-1$ and $W = \cup_{i=0}^{kn-1} W_i$ (observe that in our situation $\mathrm{Orb}_f(c)$ is disjoint from W). Also, choose a point $z \in (c, b)$. Then, we define our map $f_1 \colon T \longrightarrow T$ by:

(3.i) $f_1|_{T\setminus W} = f|_{T\setminus W}$,

(3.ii) $f_1(x) = f(c)$ for all $x \in [c, z]$,

(3.iii) f_1 is linear on $[z, b]$ and $\mathrm{Cl}(W_i)$ for $i = 1, 2, \dots, kn-1$.

It is not difficult to see that f_1 is well defined and continuous, and satisfies (g.3') by (2.5). Moreover, $f_1|_{\mathrm{Orb}_f(c)} = f|_{\mathrm{Orb}_f(c)}$, $f_1(W_0) = \{f_1(c)\} \cup W_1$, $f_1(W_i) = W_{i+1}$ for $i = 1, 2, \dots, kn-2$ and $f_1(W_{kn-1}) \subset W_0$. Therefore, in view of (3.ii) and (3.iii), any point from W is eventually mapped by f_1 to $\mathrm{Orb}_{f_1}(c)$. Consequently, $\mathrm{Per}(f_1) \subset \mathrm{Per}(f)$ and $\mathrm{Orb}_{f_1}(v)$ is finite because W contains points from $\mathrm{Orb}_f(v)$. This proves (g.2') and (g.1').

Case 4. *$f^{kn}(x) \notin [c, x]$ for each $x \in J$.*

Let $l \in \mathbb{N}$ be such that $lkn > m$. Since $f^{kn}(c) = c$, by continuity we can choose an open interval K_0 having c as an endpoint and satisfying $f^{ikn}(K_0) \subsetneq f^{(i+1)kn}(K_0)$ for $0 \leq i < l$. Set $K_i = f^{ikn}(K_0)$ for $i = 1, \dots, l$. To define the map f_1 in this case we follow a procedure similar to the one used in Case 1. We set

$$r = \min\{i \in \mathbb{N} : f^i(v) \in K_0\}, \text{ and}$$
$$s = \min\{i \in \mathbb{N} : i > r \text{ and } f^i(v) \in (c, f^r(v))\}.$$

By construction we have $s > r + lkn > r + m$.

In the rest of the proof we will have to consider two more situations and define our map $f_1 : T \to T$ in a different way in each of these situations. However, in both cases f_1 will be continuous and well defined and, for a certain point $u \in (c, f^s(v)]$, it will satisfy $f_1|_{T\setminus(u, f^r(v))} = f|_{T\setminus(u, f^r(v))}$ and $f_1(x) = f(f^r(v))$ for all $x \in [f^s(v), f^r(v)]$ (in particular $f_1(f^s(v)) = f(f^r(v)) = f_1(f^r(v))$). Such a map f_1 verifies (g.3') by (2.5). Moreover, by definition, $f_1^i(v) = f^i(v)$ for $i = 0, 1, \dots, s$, and $Q = \{f^{r+1}(v), f^{r+2}(v), \dots, f^s(v)\}$ is a periodic orbit of f_1 of period larger than m. Then, $f_1^{r+1}(v) = f^{r+1}(v) \in Q$ and (g.1') holds. We have $f_1([f^s(v), f^r(v)]) = \{f^{r+1}(v)\} \subset Q$. Thus, any periodic point of f_1 of period smaller than or equal to m which is not a periodic point of f must be in $(u, f^s(v))$. So, to prove (g.2') we have to choose the point u and define f_1 on $(u, f^s(v))$ so that such a periodic point of f_1 does not exist.

First we assume that there exists $u \in (c, f^s(v)]$ such that $f(u) = f(f^r(v))$. We take a map f_1 that verifies the above conditions and $f_1(x) = f(f^r(v))$ for all $x \in (u, f^s(v))$. Since $f_1((u, f^s(v))) = \{f^{r+1}(v)\} \subset Q$ and $Q \cap (u, f^s(v)) = \emptyset$, there is no periodic point of f_1 in $[u, f^s(v)]$. This proves (g.2').

Finally we assume that there exists a point $b \in [f(c), f^{r+1}(v))$ such that $f([c, f^s(v)]) = [f(c), b]$. Let $u \in [c, f^s(v)]$ be an f-preimage of b, and notice that $f([f^s(v), f^r(v)]) \supset [b, f^{r+1}(v)]$. In this situation we take our map f_1 to verify the above assumptions and to be linear on $[u, f^s(v)]$. From the definiton of f_1 it follows that

$$f_1([u, f^s(v)]) = [f(u), f(f^r(v))] = [b, f^{r+1}(v)] \subset f([f^s(v), f^r(v)]).$$

Hence,

$$\begin{array}{rcl}
f_1^{kn}([u, f^s(v)]) & \subset & f_1^{kn-1}(f([f^s(v), f^r(v)])) = f^{kn}([f^s(v), f^r(v)]) \\
& \subset & K_1 \setminus [c, f^s(v)]; \\
f_1^{2kn}([u, f^s(v)]) & \subset & f_1^{kn}(K_1 \setminus [c, f^s(v)]) \\
& = & f_1^{kn-1}(f_1([f^s(v), f^r(v)) \cup K_1 \setminus [c, f^r(v)))) \\
& = & f_1^{kn-1}(f(K_1 \setminus [c, f^r(v)))) = f^{kn}(K_1 \setminus [c, f^r(v))) \\
& \subset & K_2 \setminus [c, f^r(v)]; \text{ and} \\
f_1^{ikn}([u, f^s(v)]) & \subset & f_1^{(i-2)kn}(K_2 \setminus [c, f^r(v)]) \\
& \subset & f_1^{(i-3)kn}(f(K_2 \setminus [c, f^r(v)])) \\
& \subset & f_1^{(i-3)kn}(K_3 \setminus [c, f^r(v)]) \subset \cdots \subset K_l \setminus [c, f^r(v)]
\end{array}$$

for $3 \le i \le l$. Therefore, $f_1^i(x) \notin [u, f^s(v)]$ for each $x \in [u, f^s(v)]$ and $i = 1, 2, \ldots, lkn$. This means that if there is a periodic point of f_1 in $[u, f^s(v)]$ it has period larger than $lkn > m$. This ends the proof of the proposition. ■

Remark 2.8.6. From the proof of Proposition 2.8.5 it follows that f and $\overline{f}$ not only coincide on $D \cup V(T)$ but also on $\bigcup_{i=0}^{N} f^i(D \cup V(T))$. □

2.8.4 Stage 2: completion to graph models

We start by introducing the notion of a *completion*. Given a pointed tree (T, A) we will denote by (T^G, A) a pointed graph obtained by attaching an oriented edge B_x for each $x \in A$ such that B_x is homeomorphic to a circle and x is the unique vertex of T^G which belongs to B_x. Observe that $T \subset T^G$,

$V(T^G) = V(T) \cup A$ and $B_x \cap T = \{x\}$. Any monotone loop based at x whose image turns once around B_x will be denoted by β_x. Given $x, y \in T$, any monotone path from x to y will be denoted by σ_{xy}, and the trivial loop based at x will be denoted by σ_x.

Let (T, A) be a pointed tree. We choose one point from each discrete component of (T, A), which will be called the *base point* of the discrete component (note that we do not assume that the base points are pairwise different). This choice of base points will be called an *orientation* for (T, A), which will be called an *oriented* pointed tree. Orienting (T, A) allows us to fix a free system of generators of $\pi(T^G, A)$. Indeed, let $K_1, K_2, \ldots, K_n$ be the discrete components of (T, A), with respective base points $z_1, z_2, \ldots, z_n$. We define

$$\mathcal{L}(T, A) = \bigcup_{i=1}^{n} \bigcup_{x \in K_i \setminus \{z_i\}} [\sigma_{z_i x}], \quad \mathcal{T}(T, A) = \bigcup_{x \in A} [\sigma_x], \quad \mathcal{C}(T, A) = \bigcup_{x \in A} [\beta_x],$$

where $[\gamma]$ denotes the equivalence class of a path γ on the quotient space $\pi(T^G, A)$. Then, from Proposition 2.3 of [2] and its proof, it follows that $\mathcal{T}(T, A) \cup \mathcal{L}(T, A) \cup \mathcal{C}(T, A)$ is a free system of generators of $\pi(T^G, A)$.

Let (T, A) and (S, P) be oriented pointed trees such that there exists a bijection $\theta \colon A \longrightarrow P$ which preserves discrete components. We will say that the orientations of (T, A) and (S, P) are *θ-compatible* if, for each discrete component K of (T, A), z is the base point of K if and only if $\theta(z)$ is the base point of $\theta(K)$. In this case a natural isomorphism

$$\theta^\star : \pi(T^G, A) \longrightarrow \pi(S^G, P)$$

is defined as follows (on the generators of $\pi(T^G, A)$). If $[\sigma_x] \in \mathcal{T}(T, A)$ then $\theta^\star([\sigma_x]) = [\sigma_{\theta(x)}] \in \mathcal{T}(S, P)$. If $[\sigma_{xy}] \in \mathcal{L}(T, A)$ then $\theta^\star([\sigma_{xy}]) = [\sigma_{\theta(x)\theta(y)}] \in \mathcal{L}(S, P)$. Finally, if $[\beta_x] \in \mathcal{C}(T, A)$ then $\theta^\star([\beta_x]) = [\beta_{\theta(x)}] \in \mathcal{C}(S, P)$.

Lemma 2.8.7. *Let (T, A, f) and (S, P, g) be two models with the same pattern and let $\theta : A \longrightarrow P$ be a bijection which preserves discrete components. Then there are homotopy equivalences $(T^G, A) \xrightarrow{r} (S^G, P) \xrightarrow{s} (T^G, A)$ such that $r^\star = \theta^\star$ and $s^\star = (\theta^\star)^{-1}$.*

Proof. We choose θ-compatible orientations for (T, A) and (S, P). It is enough to take r and s in such a way that $r|_A = \theta$, $s|_P = \theta^{-1}$, r is a homeomorphism between B_z and $B_{\theta(z)}$ for each $z \in A$, and, for each discrete component K of (T, A), r and s are homotopy equivalences between $\langle K \rangle_T$ and $\langle \theta(K) \rangle_S$. ■

The notion of a completion is accompanied by the following notation, which will be used in the rest of this section:

Standing notation. Let (T, A, f) be a periodic y-expansive canonical model and let $K_i = \{z_1^i, z_2^i, \dots, z_{k_i}^i\}$ with $i = 1, 2, \dots, n$ be the discrete components of (T, A) (for convenience we also set $k_0 = 0$). For $1 \le i \le n$ and $1 \le j \le k_i$, we define t_j^i, x_j^i and a_j^i as follows:

- $t_j^i = \sum_{m=0}^{i-1} k_m + j$.
- x_j^i is the unique point from $A^y \cap \langle K_i \rangle_T$ such that $(z_j^i, x_j^i) \cap A^y = \emptyset$.
- a_j^i is any point from (z_j^i, x_j^i) (when $|K_i| = 2$ then $\langle K_i \rangle_T = [z_1^i, z_2^i]$ and we additionally impose $z_1^i < a_1^i < a_2^i < z_2^i$).

We also set $\mathsf{B}_A = \cup_{x \in A} B_x$, $\mathsf{L}_A = \cup_{i,j} [a_j^i, z_j^i]$ and $\mathsf{M}_A = \mathsf{L}_A \cup \mathsf{B}_A$.

A graph model $(T^G, A, \mathbf{f})$ will be called a *completion* of (T, A, f) if the following conditions hold:

(C 1) $\mathbf{f}|_{A^y} = f|_{A^y}$, and if $x, z \in A$ with $[x, z] \cap A = \{x, z\}$ then $\mathbf{f}$ is monotone on $[x, z]$.

(C 2) $\mathbf{f}(a_j^i) = \mathbf{f}(z_j^i)$ and $\mathbf{f} \circ \sigma_{a_j^i z_j^i}$ is a monotone increasing loop which turns t_j^i times around $B_{\mathbf{f}(z_j^i)}$.

(C 3) Let $t = \max\{t_j^i\} + 1$. Given any $z \in A$, $\mathbf{f}$ is monotone on B_z and the $\mathbf{f}$-image of B_z turns t times around $B_{\mathbf{f}(z)}$.

(C 4) $\mathbf{f}$ coincides with f on $T \setminus \cup_{i,j} (z_j^i, x_j^i)$ and, for each $1 \le i \le n$ and $1 \le j \le k_i$, $\mathbf{f}$ maps linearly $[a_j^i, x_j^i]$ onto $[\mathbf{f}(z_j^i), \mathbf{f}(x_j^i)]$.

Sometimes we will simply speak about a *completion* $(T^G, A, \mathbf{f})$, without specifying the original canonical model. $\square$

The next lemma, which states some basic properties of a completion, follows immediately from the definitions.

Lemma 2.8.8. *Let $(T^G, A, \mathbf{f})$ be a completion of a y-expansive canonical model. Then the following statements hold:*

(a) If $\{x, z\} \subset T \setminus \mathsf{M}_A$ and $[x, y] \cap A = \emptyset$ then $[x, y] \subset T \setminus \mathsf{M}_A$.

(b) If $[x, z] \subset T$ and $(x, z) \cap A = \emptyset$ then $\mathbf{f}$ is monotone on $[x, z]$. If in addition $(x, z) \subset T \setminus \mathsf{M}_A$ then $[\mathbf{f}(x), \mathbf{f}(z)] \subset T$.

(c) $\mathbf{f}(\mathsf{B}_A) \subset \mathsf{B}_A$ and $\mathbf{f}(\mathsf{L}_A) \subset \mathsf{B}_A$. Thus M_A is $\mathbf{f}$-invariant.

(d) There are no periodic orbits of $\mathbf{f}$ in $\mathsf{L}_A \setminus A$. Therefore, all the periodic orbits of $\mathbf{f}$ contained in $\mathsf{M}_A \setminus A$ are contained in B_A.

(e) $(T^G, A^y, \mathbf{f})$ is a Markov model.

The following remark and example tell us why properties (C 2–3) of the definition of a completion have been chosen in such a particular way.

Remark 2.8.9. The exact values of the numbers t^i_j and t of the definition of a completion are not important. They have been chosen to satisfy the following properties: $t > \max\{t^i_j\}_{i,j}$ and $t^i_j \neq t^l_m$ when (i,j) and (l,m) are two different ordered couples. In this case it follows that $t^i_j + nt - t^l_m \neq 0$ for each $n \in \mathbb{Z}$. This fact has a consequence which will be crucial in the proof of Proposition 2.8.15 (and thus in the Stage 3 of the proof of Theorem 2.5.1): let $x \in A$ which belongs to two different discrete components K_i and K_l. Then $x = z^i_j = z^l_m$ for some $j \in \{1,2,\dots,k_i\}$ and $m \in \{1,2,\dots,k_l\}$. Now let $a \in K_i \setminus \{x\}$ and $b \in K_l \setminus \{x\}$, and let γ be a monotone path from a to b. The path γ can be written as $\sigma_{ax}(\beta_x)^n\sigma_{xb}$ for some $n \in \mathbb{Z}$. Then $\mathbf{f}(\gamma) = \sigma\beta_{f(x)}^{t^i_j+nt-t^l_m}\sigma'$ for some paths σ and σ' whose images contain $[f(a), f(x)]$ and $[f(x), f(b)]$ respectively. Since $t^i_j + nt - t^l_m \neq 0$, we have proved that *given any monotone path γ which passes through a point $x \in A$ it follows that $\mathbf{f} \circ \gamma$ turns at least once around $B_{\mathbf{f}(x)}$*. $\square$

Example 2.8.10. Consider the model (T, A, f) shown in the left side of Figure 2.4. The pointed tree (T, A) has 2 discrete components $K_1 = \{a, b, c\}$ and $K_2 = \{c, d\}$, and A is a 4-periodic orbit of f with $f(a) = b$, $f(b) = c$, $f(c) = d$ and $f(d) = a$. Set $a = z^1_1$, $b = z^1_2$, $c = z^1_3 = z^2_1$ and $d = z^2_2$. We have $t^1_1 = 1$, $t^1_2 = 2$, $t^1_3 = 3$, $t^2_1 = 4$, $t^2_2 = 5$ and $t = 6$. Then $\mathbf{f}$ acts as follows on the next monotone paths:

- $\mathbf{f} \circ \sigma_{ab} = (\beta_b)^{-1}\sigma_{bc}(\beta_c)^2$.
- $\mathbf{f} \circ \sigma_{ac} = (\beta_b)^{-1}\sigma_{bd}(\beta_d)^3$.
- $\mathbf{f} \circ \sigma_{cd} = (\beta_d)^{-4}\sigma_{da}(\beta_a)^5$.
- $\mathbf{f} \circ \beta_a = (\beta_b)^6$, $\mathbf{f} \circ \beta_b = (\beta_c)^6$, $\mathbf{f} \circ \beta_c = (\beta_d)^6$, $\mathbf{f} \circ \beta_d = (\beta_a)^6$.

Therefore, if we consider the path $\gamma = \sigma_{ad}$, which passes through the point c, we get $\mathbf{f} \circ \gamma = \mathbf{f} \circ \sigma_{ac} \circ \mathbf{f} \circ \sigma_{cd} = (\beta_d)^{-1}\sigma_{bd}(\beta_d)^{3-4}\sigma_{da}(\beta_a)^5$, a path which passes through the point d turning once around B_d. $\square$

Now we are ready to state and prove Proposition 2.8.11, which is the main result of this subsection.

Proposition 2.8.11. *Let (S, D, h) be a Markov linear model such that D contains a periodic orbit P with $\mathrm{En}(S) \subset P$. Let (T, A, f) be a y-expansive canonical model of $[S, P, h]$. Then there exist graph models $(T^G, A, \mathbf{f})$ and $(S^G, D, \mathbf{h})$ verifying:*

(a) $(T^G, A, \mathbf{f})$ is a completion of (T, A, f).

(b) $(S^G, D, \mathbf{h})$ is Markov and $\mathbf{h}|_D = h|_D$. Moreover, $\mathbf{h}$ is linear on each interval $I \subset S \setminus D$ such that $\mathbf{h}(I) \subset S$.

(c) $\mathbf{f} \simeq_A s \circ \mathbf{h} \circ r$, where $(T^G, A) \xrightarrow{r} (S^G, P) \xrightarrow{s} (T^G, A)$ are the homotopy equivalences given by Lemma 2.8.7.

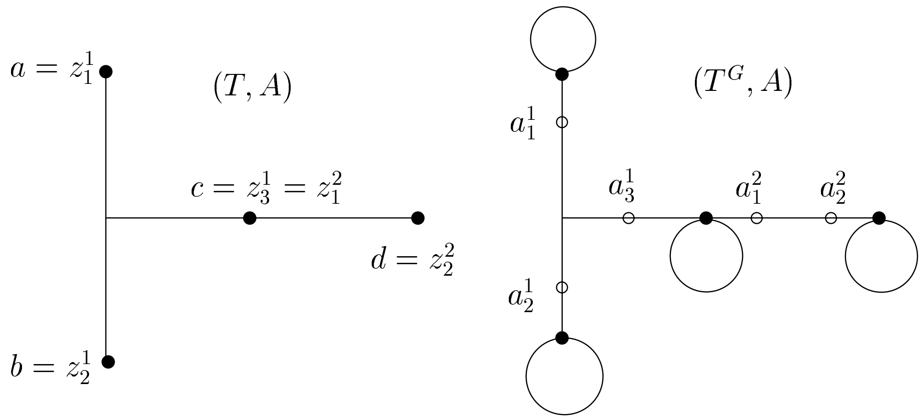

Figure 2.4: The model (T, A, f) of Example 2.8.10 and a completion.

Proof. Since (T, A, f) and (S, P, h) have the same pattern, there exists a bijection $\theta : A \longrightarrow P$ which preserves discrete components. Then by Lemma 2.8.7 there exist homotopy equivalences $(T^G, A) \xrightarrow{r} (S^G, P) \xrightarrow{s} (T^G, A)$ such that $r^\star = \theta^\star$ and $s^\star = (\theta^\star)^{-1}$.

Statement (a) holds simply by choosing appropriate points a_j^i in the definition of a completion.

Next we define the model $(S^G, D, \mathbf{h})$ in order that (b) and (c) hold. To this end, we essentially repeat the construction of a completion (which is defined only for canonical models) in this setting (Markov models). Note that $\theta(K_1), \theta(K_2), \dots, \theta(K_n)$ are the discrete components of (S, P) and $D \supset V(S) \cup P = V(S) \cup (\cup_{i=1}^n \theta(K_i)$. For each $1 \le i \le n$ and $1 \le j \le k_i$, we take $y_j^i \in D \cap \langle \theta(K_i) \rangle_S$ such that $(\theta(z_j^i), y_j^i) \cap D = \emptyset$ and $b_j^i \in (\theta(z_j^i), y_j^i)$. We define $\mathbf{h}$ to coincide with h on $S \setminus \cup_{i,j} (\theta(z_j^i), y_j^i)$. Also, for each $1 \le i \le n$ and $1 \le j \le k_i$, we define $\mathbf{h}$ to map linearly $[b_j^i, y_j^i]$ onto $[h(\theta(z_j^i)), h(y_j^i)]$. Moreover, $\mathbf{h}$ is defined to be monotone increasing on $(b_j^i, \theta(z_j^i))$, its image turning t_j^i times around $B_{h(\theta(z_j^i))}$. Finally, if $z \in A$ then $\mathbf{h}$ is defined to be monotone increasing on $B_{\theta(z)}$, its image turning t times around $B_{h(\theta(z))}$. Then, it is not difficult to see that (b) and (c) hold by construction. ∎

The following remarks, which concern Proposition 2.8.11, will be used in Stage 4 of the proof of Theorem 2.5.1.

Remark 2.8.12. In the hypotheses of Proposition 2.8.11, since $\mathbf{h}(B_a) = B_{\mathbf{h}(a)}$ for each $a \in P$, it easily follows that the period of each $\mathbf{h}$-periodic point contained in B_a is a multiple of $|P|$. Moreover, since the $\mathbf{h}$-image of B_a turns increasingly t times around $B_{\mathbf{h}(a)}$ for each $a \in P$, it follows that for each $k \in \mathbb{N}$ there are exactly $t^{k|P|}$ isolated fixed points of $\mathbf{h}^{k|P|}$ contained in B_a, and $\mathbf{h}^{k|P|}$ is monotone increasing on each of these points. □

Let (G, Q, F) be a Markov graph model. As usual we can consider the *Markov F-graph of Q*, whose vertices are the closures of the connected components of $G \backslash Q$ and there is an arrow from the vertex I to the vertex J if and only if there is a subinterval K of I such that $F(K) = J$. Let $(T^G, A, \mathbf{f})$ and $(S^G, D, \mathbf{h})$ be the Markov graph models given by Proposition 2.8.11. Consider the set $\mathcal{K}$ of all the closures of the connected components of $T^G \setminus A^y$ which are contained in T. This set coincides with the set of all the A^y-basic intervals of the tree T. By identifying both sets and using the properties of $\mathbf{f}$, it is not difficult to see that the subgraph $\mathcal{G}$ of the Markov $\mathbf{f}$-graph of A^y whose set of vertices is $\mathcal{K}$ is isomorphic to the Markov f-graph of A^y. From now on, $\mathcal{G}$ will be called the *Markov $\mathbf{f}$-subgraph of A^y*. Analogously, we construct the *Markov $\mathbf{h}$-subgraph of D*, which is isomorphic to the Markov h-graph of D.

Remark 2.8.13. It can be seen that Lemmas 1.3.2, 1.3.3 and 1.3.5 (and their proofs) remain valid in this setting. $\square$

2.8.5 Stage 3: n is a period of the completion of $(S, \overline{P}, \overline{g})$

In this subsection we state and prove Proposition 2.8.15, which is the main result used in Stage 3 of the proof of Theorem 2.5.1. To do it, we first introduce the notion of a *simplification* of paths and prove a technical result on this notion (Lemma 2.8.14).

Let $(T^G, A, \mathbf{f})$ be a completion. Given points $w \in T \setminus \mathsf{L}_A$ and $z \in T^G$, and a path γ in T^G from w to z, a *simplification of* γ is any monotone path which is equivalent to γ. If $z \in T$ and $[w, z] \cap A = \emptyset$ then we define the *A-length of* γ, denoted by $l(\gamma)$, to be 0. Otherwise, any simplification of γ can be written as a product of paths $\alpha_0 \beta_1^{n_1} \alpha_1 \beta_2^{n_2} \alpha_2 \dots \alpha_{k-1} \beta_k^{n_k} \alpha_k$, where $k \geq 1$ and:

(PS 1) $n_i \in \mathbb{Z}$ for $1 \leq i \leq k$.

(PS 2) For each $1 \leq i \leq k$, there exist $x_i \in A$ such that $\beta_i = \beta_{x_i}$.

(PS 3) Set $x_0 = w$. Then $\alpha_i = \sigma_{x_i x_{i+1}}$ and $\alpha_i((0,1)) \cap A = \emptyset$ for each $0 \leq i < k$. Note that $\alpha_i([0,1]) \subset T$.

(PS 4) α_k is a trivial path if and only if $z \in A$. If $z \in T^G \setminus T$ (equivalently, if $z \in B_{x_k} \setminus \{x_k\}$), then $\alpha_k((0,1))$ is one of the two connected components of $B_{x_k} \setminus \{x_k, z\}$ (determined by the fact that a simplification of γ is equivalent to γ).

(PS 5) No subexpression of the simplification corresponds to a path being equivalent to a trivial path (except, perhaps, α_0, α_k, and $\beta_i^{n_i}$ when $n_i = 0$).

In this case we define the *A-length of* γ to be $l(\gamma) = k$.

As a consequence of (PS 5), the expression of a simplification of γ as a product of paths is essentially unique (up to equivalence classes of paths). Observe that two equivalent paths have a common simplification and thus the same A-length.

Lemma 2.8.14. *Let $(T^G, A, \mathbf{f})$ be a completion and let γ be a path from w to z, with $z \in T^G$ and $\mathrm{Orb}(w) \subset T \setminus \mathsf{L}_A$. Then the following statements hold:*

(a) $l(\mathbf{f}^n \circ \gamma) \geq l(\gamma)$ for each $n \geq 0$.

(b) If $l(\gamma) > 0$ and $\alpha_0 \beta_1^{n_1} \alpha_1 \dots \alpha_{k-1} \beta_k^{n_k} \alpha_k$ is a simplification of γ, then $\beta_i(0) \neq \beta_{i+1}(0)$ for each $1 \leq i < k$. Moreover, for each $n \geq 0$ we have $l(\mathbf{f}^n \circ \gamma) = l(\gamma)$ if and only if $\mathbf{f}^n(\alpha_i([0,1])) \cap A = \{\mathbf{f}^n(\alpha_i(0)), \mathbf{f}^n(\alpha_i(1))\}$ for each $1 \leq i < k$ and $\mathbf{f}^n(\alpha_0([0,1])) \cap A = \{\mathbf{f}(\alpha_0(1))\}$.

Proof. Since when $l(\gamma) = 0$ there is nothing to prove, from now on we assume that $l(\gamma) > 0$. Let $\alpha_0 \beta_1^{n_1} \alpha_1 \dots \alpha_{k-1} \beta_k^{n_k} \alpha_k$ be a simplification of γ.

First we claim that $\beta_i(0) \neq \beta_{i+1}(0)$ for each $1 \leq i < k$. Indeed, assume that there exists $1 \leq j < k$ such that $\beta_j(0) = \beta_{j+1}(0)$. Then, from (PS 3) it follows that α_j is a path whose image is contained in a tree and does not intersect A except at $\alpha_j(0)$, which in addition coincides with $\alpha_j(1)$. Therefore α_j is a path with a trivial homotopy. This contradicts (PS 5), and thus the claim follows.

Now we prove (a) and (b). Observe that $l(\mathbf{f}^n \circ \gamma)$ is well defined for each $n \geq 0$ because $\mathrm{Orb}(w) \subset T \setminus \mathsf{L}_A$ by assumption. In order to reduce the number of cases to be considered, we will assume that $z \in T^G \setminus T$ (equivalently, $z \in B_{x_k} \setminus \{x_k\}$). The case $z \in T$ is analogous, even simpler. Observe that, since $(T^G, A, \mathbf{f})$ is a completion, it satisfies the properties (C 1–4) (see page 102). For each $1 \leq i \leq k$, let $x_i \in A$ be such that $\beta_i = \beta_{x_i}$. Set $x_0 = w$ and $\widetilde{\alpha}_i = \sigma_{\mathbf{f}(x_i)\mathbf{f}(x_{i+1})}$ for $0 \leq i < k$. Then the following statements hold:

1. For $1 \leq i \leq k$, from (C 1–2) it follows that $\mathbf{f} \circ \alpha_i = \beta_{\mathbf{f}(x_i)}^{p_i} \widetilde{\alpha}_i \beta_{\mathbf{f}(x_{i+1})}^{l_{i+1}}$ for some integers $p_i < 0$ and $l_{i+1} > 0$, and $\mathbf{f} \circ \alpha_0 = \widetilde{\alpha}_0 \beta_{\mathbf{f}(x_1)}^{l_1}$ for some integer $l_1 > 0$.
2. Set $p_k = 0$. For each $1 \leq i \leq k$, from (C 3) it follows that $\mathbf{f} \circ \beta_i^{n_i} = \beta_{\mathbf{f}(x_i)}^{tn_i}$ for an integer t such that $|t| > \max\{|l_i|, |p_i|\}$. Since $l_i > 0$ and $p_i < 0$, it follows that $|t| \geq |l_i + p_i|$.
3. $\mathbf{f} \circ \alpha_k = \beta_{\mathbf{f}(x_k)}^m \widetilde{\alpha}_k$ for some integer m with $|m| \leq t$ and a path $\widetilde{\alpha}_k$ such that either $\mathbf{f}(z) = \mathbf{f}(x_k)$ and $\widetilde{\alpha}_k$ is trivial, or $\mathbf{f}(z) \neq \mathbf{f}(x_k)$ and $\widetilde{\alpha}_k((0,1))$ is one of the two connected components of $B_{\mathbf{f}(x_k)} \setminus \{\mathbf{f}(z), \mathbf{f}(x_k)\}$.

Summarizing, we have that

$$\mathbf{f} \circ \gamma = \widetilde{\alpha}_0 \beta_{\mathbf{f}(x_1)}^{l_1 + tn_1 + p_1} \widetilde{\alpha}_1 \beta_{\mathbf{f}(x_2)}^{l_2 + tn_2 + p_2} \widetilde{\alpha}_2 \dots \widetilde{\alpha}_{k-1} \beta_{\mathbf{f}(x_k)}^{l_k + tn_k + p_k + m} \widetilde{\alpha}_k \tag{2.6}$$

and $l_i + tn_i + p_i \neq 0$ holds for each $1 \leq i < k$. Moreover, $\widetilde{\alpha}_i$ is injective and, by the claim above, $\beta_{\mathbf{f}(x_i)}(0) \neq \beta_{\mathbf{f}(x_{i+1})}(0)$. Finally, $\widetilde{\alpha}_{k-1} \neq \widetilde{\alpha}_k^{-1}$ since $\widetilde{\alpha}_{k-1}((0,1)) \subset T$ and $\widetilde{\alpha}_k((0,1)) \subset T^G \setminus T$. Therefore, no subexpression of (2.6) is equivalent to a trivial path. If $\mathbf{f}(\alpha_i([0,1])) \cap A = \{\mathbf{f}(\alpha_i(0)), \mathbf{f}(\alpha_i(1))\}$ for $1 \leq i < k$ and $\mathbf{f}(\alpha_0([0,1])) \cap A = \{\mathbf{f}(\alpha_0(1))\}$ then $\widetilde{\alpha}_i((0,1)) \cap A = \emptyset$ for $0 \leq i < k$. Hence, (2.6) is the expression of a simplification of $\mathbf{f} \circ \gamma$ and consequently $l(\mathbf{f} \circ \gamma) = l(\gamma)$. Otherwise, $l(\mathbf{f} \circ \gamma) > l(\gamma)$. Thus (a) and (b) hold for $n = 1$.

Observe that $\mathbf{f} \circ \gamma$ is a path starting at $\mathbf{f}(w)$, which belongs to $T \setminus \mathsf{L}_A$ by assumption. Moreover, when $l(\mathbf{f} \circ \gamma) = l(\gamma)$ the right hand side of (2.6) is a simplification of $\mathbf{f} \circ \gamma$. Thus we can repeat the above arguments with $\mathbf{f} \circ \gamma$ instead of γ, and hence the proof for any $n > 1$ follows analogously by induction. ∎

Now we are ready to state and prove Proposition 2.8.15.

Proposition 2.8.15. *Let (T, A, f) be a non-twist y-expansive canonical model and let $(T^G, A, \mathbf{f})$ be a completion of (T, A, f). Let $x \in T \setminus A^y$ be an n-periodic point of f associated to a unique simple negative loop β in the Markov f-graph of A^y, with $|A| \neq n = |\beta|$. Then there exists a periodic orbit Q of $\mathbf{f}$ such that:*

(a) Q is associated to β and $|Q| = n$.

(b) $Q \cap (\mathsf{M}_A \cup V(T)) = \emptyset$.

(c) For each $w \in Q$ we have $[w, \mathbf{f}^n] = \{w\}$ and $\operatorname{ind}[w; \mathbf{f}^n] = -1$.

(d) The graph pattern $[T^G, Q, \mathbf{f}]$ is irreducible.

Proof. β is also a loop in the Markov $\mathbf{f}$-subgraph of A^y (which is isomorphic to the Markov f-graph of A^y). By Lemma 1.3.3, there exists a periodic orbit $Q \subset T$ of $\mathbf{f}$ such that Q and β are associated. Since β is simple and negative, from Lemma 1.3.5(b) it follows that Q is an n-periodic orbit and $Q \cap A^y = \emptyset$. In particular, $Q \cap \mathsf{B}_A = \emptyset$. Thus, by Lemma 2.8.8(d) we have that $Q \cap (\mathsf{M}_A \cup V(T)) = \emptyset$. Therefore, (a) and (b) hold.

Now we prove (c). Let $w \in Q$. Since (b) holds, $w \notin \mathsf{M}_A \cup V(T) \supset V(T^G)$. Hence, w belongs to the interior of a $V(T^G)$-basic interval I. Since β is negative, by Remark 1.3.4 there is a closed interval $K \subset I$ such that $\mathbf{f}^n|_K$ is monotone decreasing and w is the unique fixed point of $\mathbf{f}^n$ in K. Thus $\mathbf{f}^n$ is decreasing at w. It follows that the index of w as a fixed point of $\mathbf{f}^n$ equals -1 (see Chapter 1 of [33] on how to compute the index of a fixed point; as it has been said before, we follow [2], where the considered index is minus that defined in [33]).

To end the proof of (c) we must show that w is the unique element of its class of fixed points of $\mathbf{f}^n$. Assume the contrary: let $z \in \operatorname{Fix}(\mathbf{f}^n)$ with $w \neq z$

and let γ be a path from w to z such that $\mathbf{f}^n \circ \gamma \backsim \gamma$. Note that, for each $i \geq 0$, $\mathbf{f}^i \circ \gamma$ is a path from $\mathbf{f}^i(w)$ to $\mathbf{f}^i(z)$. Since, by (b), $\mathbf{f}^i(w) \notin \mathsf{M}_A$, the A-length $l(\mathbf{f}^i \circ \gamma)$ is defined for each $i \geq 0$. Since $l(\mathbf{f}^n \circ \gamma) = l(\gamma)$, from Lemma 2.8.14(a) it follows that $l(\mathbf{f}^i \circ \gamma) = l(\gamma)$ for each $1 \leq i \leq n$. Indeed, if $l(\mathbf{f}^j \circ \gamma) > l(\gamma)$ for some $1 \leq j < n$ then $l(\gamma) = l(\mathbf{f}^n \circ \gamma) \geq l(\mathbf{f}^{n-1} \circ \gamma) \geq \ldots \geq l(\mathbf{f}^j \circ \gamma) > l(\gamma)$, a contradiction.

Assume that $l(\gamma) = 0$. Thus $[w, z] \cap A = \emptyset$. In particular, since $w \in Q \subset T$ then $z \notin \mathsf{B}_A$. Moreover, $z \in T \setminus \mathsf{M}_A$ by Lemma 2.8.8(d). From (a) and (b) of Lemma 2.8.8, it follows that $[w, z] \subset T \setminus \mathsf{M}_A$ and $\mathbf{f}$ is monotone on $[w, z]$. Observe that, since $l(\mathbf{f} \circ \gamma) = 0$, we can repeat this argument with $\mathbf{f} \circ \gamma$, $\mathbf{f}(w)$ and $\mathbf{f}(z)$ instead of γ, w and z, to obtain that $\mathbf{f}$ is monotone on $[\mathbf{f}(w), \mathbf{f}(z)]$, and thus $\mathbf{f}^2$ is monotone on $[w, z]$. By using iteratively this argument we finally get that $\mathbf{f}^n$ is monotone on $[w, z]$, in contradiction with the fact that $\mathbf{f}^n$ is decreasing at w and $\mathbf{f}^n(w) = w$. This proves (c) in the case $l(\gamma) = 0$.

Assume now that $l(\gamma) \geq 1$. Set $k = l(\gamma)$ and let $\alpha_0 \beta_1^{n_1} \alpha_1 \ldots \alpha_{k-1} \beta_k^{n_k} \alpha_k$ be a simplification of γ. Let $x' \in A$ be such that $\alpha_0(1) = x'$. Since $l(\mathbf{f}^n \circ \gamma) = l(\gamma)$, by Lemma 2.8.14(b) we have $\mathbf{f}^n(\alpha_0([0,1])) \cap A = \{\mathbf{f}^n(\alpha_0(1))\} = \{\mathbf{f}^n(x')\}$. Since $\mathbf{f}^n \circ \gamma \backsim \gamma$, it easily follows that $\mathbf{f}^n(x') = x'$. Therefore, $\mathbf{f}^n$ is monotone on $[w, x']$, in contradiction with the fact that $\mathbf{f}^n$ is decreasing at w and $\mathbf{f}^n(w) = w$. This proves (c) in the case $l(\gamma) > 0$.

Finally we prove (d). Let $w \in Q$. Assume that $[T^G, Q, \mathbf{f}]$ is reducible. Then there exist $m < |Q|$ and $q > 1$ with $|Q| = qm$ and a path γ from w to $\mathbf{f}^m(w)$ such that

$$[\gamma \circ \mathbf{f}^m \circ \gamma \cdots \mathbf{f}^{(q-1)m} \circ \gamma] = [\sigma_w].$$

It follows that $[\gamma]^{-1} = [\mathbf{f}^m \circ \gamma \cdots \mathbf{f}^{(q-1)m} \circ \gamma]$. Also, by mapping m times $\mathbf{f}^\star$ on both sides of the equality, it follows that $[\mathbf{f}^m \circ \gamma \cdots \mathbf{f}^{(q-1)m} \circ \gamma \circ \mathbf{f}^{qm} \circ \gamma] = [\sigma_{\mathbf{f}^m(w)}]$. Thus, $[\gamma]^{-1}[\mathbf{f}^{qm} \circ \gamma] = [\sigma_{\mathbf{f}^m(w)}]$. In other words, $[\mathbf{f}^{qm} \circ \gamma] = [\gamma]$. Hence w and $\mathbf{f}^m(w)$ are Nielsen equivalent fixed points of $\mathbf{f}^{qm}$, in contradiction with (c). Thus (d) is proved. ∎

2.8.6 Proof of Theorem 2.5.1

Now we are ready to perform the four Stages of the programme proposed in subsection 2.8.2 in order to prove Theorem 2.5.1.

Proof of Theorem 2.5.1. Set $N = \max\{M(S), |P|\}$. By Theorem 2.7.7, the statement holds when $n > N \geq M(S)$. So, from now on we assume that $n \leq N$.

Let β be a simple loop in the Markov f-graph of A^y associated to x (this loop exists and contains strictly a typical loop, since x satisfies the $(\star)$-

property). By Lemma 2.7.6, we are done if β is positive. So, we assume that β is negative.

Let $r : S \longrightarrow \langle P \rangle_S$ be the natural retraction. Consider the map $r \circ g : \langle P \rangle_S \longrightarrow \langle P \rangle_S$. It is well known (see Corollary 4.2 of [16]) that $\text{Per}(r \circ g) \subset \text{Per}(g)$. Thus it is enough to prove that $n \in \text{Per}(r \circ g)$. This holds trivially when $n = |A|$ because P is a periodic orbit of $r \circ g$ and $|P| = |A|$. From now on we assume that $n \neq |A|$.

Observe that $(\langle P \rangle_S, P, r \circ g)$ is a non-twist model and $M(\langle P \rangle_S) \leq M(S)$. Hence, without loss of generality, from now on we will use S and g to denote $\langle P \rangle_S$ and $r \circ g$ respectively.

Summarizing, we have to prove that $n \in \text{Per}(g)$ when β is negative, $\text{En}(S) \subset P$, $n \neq |P|$ and $n \leq N$.

(Stage 1) Since $|P| \leq N$, by Proposition 2.8.5 there exists a Markov linear model $(S, \overline{P}, \overline{g})$ such that $\text{Per}(\overline{g}) \cap \{1, 2, \ldots, N\} = \text{Per}(g) \cap \{1, 2, \ldots, N\}$, $P \subset \overline{P}$ and $\overline{g}|_{P \cup V(S)} = g|_{P \cup V(S)}$. Since $n \leq N$, it is enough to show that $n \in \text{Per}(\overline{g})$.

(Stage 2) Since $\text{En}(S) \subset P$ and the patterns $[S, P, \overline{g}]$ and $[S, P, g]$ coincide, (T, A, f) is a y-expansive canonical model of the pattern $[S, P, \overline{g}]$. Since $(S, \overline{P}, \overline{g})$ is a Markov linear model, Proposition 2.8.11 (with D and h replaced, respectively, by $\overline{P}$ and $\overline{g}$) gives us a completion $(T^G, A, \mathbf{f})$ of (T, A, f) and a Markov graph model $(S^G, \overline{P}, \mathbf{g})$ such that $\mathbf{g}$ and $\overline{g}$ coincide on $\overline{P}$ and $\mathbf{f} \simeq_A s \circ \mathbf{g} \circ r$, where $(T^G, A) \xrightarrow{r} (S^G, P) \xrightarrow{s} (T^G, A)$ are homotopy equivalences.

(Stage 3) Since P is non-twist, from Lemma 2.4.1 it follows that (T, A, f) is also non-twist. Then, by Proposition 2.8.15, there exists a periodic orbit Q of $\mathbf{f}$ with index -1 and period n. Thus, Q is an essential periodic orbit. Since $\mathbf{f} \simeq_A s \circ \mathbf{g} \circ r$, from Theorem 2.8.3 it follows that there exists a periodic orbit R of $\mathbf{g}$ with index -1 such that $|R|$ divides $|Q|$ and the graph pattern $[S^G, R, \mathbf{g}]$ either coincides with $[T^G, Q, \mathbf{f}]$ or it is a reduction of $[T^G, Q, \mathbf{f}]$.

By (d) of Proposition 2.8.15, $[T^G, Q, \mathbf{f}]$ is irreducible. Hence, $[S^G, R, \mathbf{g}] = [T^G, Q, \mathbf{f}]$ and, in particular, $|R| = |Q| = n$.

(Stage 4) Let $\mathcal{H}$ be the Nielsen class of a point from R with respect to $\mathbf{g}^n$. Observe that if $z \in \mathcal{H}$ then z is a point of period n of $\mathbf{g}$. Otherwise, by Theorem 3.4 of [2] the graph pattern $[S^G, \text{Orb}(z), \mathbf{g}]$ is a reduction of $[S^G, R, \mathbf{g}]$, in contradiction with the fact that $[S^G, R, \mathbf{g}] = [T^G, Q, \mathbf{f}]$ and $[T^G, Q, \mathbf{f}]$ is irreducible.

Finally we prove that there exists an n-periodic orbit of $\overline{g}$. We consider two cases.

Case 1. $\mathrm{Orb}_{\mathbf{g}}(z) \cap \overline{P} \neq \emptyset$ *for some* $z \in \mathcal{H}$.

Then $\mathrm{Orb}_{\mathbf{g}}(z) \subset \overline{P}$. Since $\mathbf{g}|_{\overline{P}} = \overline{g}|_{\overline{P}}$, then $\mathrm{Orb}_{\mathbf{g}}(z)$ is also a periodic orbit of $\overline{g}$. Therefore, $|\,\mathrm{Orb}_{\mathbf{g}}(z)| = n$. This ends the proof of the theorem in this case.

Case 2. $\mathrm{Orb}_{\mathbf{g}}(z) \cap \overline{P} = \emptyset$ *for each* $z \in \mathcal{H}$.

Since $V(T) \subset \overline{P}$, it follows that $z \notin V(T)$ and thus the index of any point in the class $\mathcal{H}$ is 0,1 or -1. Since $\mathrm{ind}[\mathcal{H}, \mathbf{g}^n] = -1$, there exists some $z \in \mathcal{H}$ whose index is -1. As stated above, z is an n-periodic point of $\mathbf{g}$.

Let us see that $z \in S$. On the contrary, $z \in B_a$ for some $a \in P$. By Remark 2.8.12, $\mathbf{g}^n$ is monotone increasing on z. Since $z \notin V(T)$, it follows that the index of z as a fixed point of $\mathbf{g}^n$ is either 0 or 1, a contradiction. Hence, $z \in S$ and therefore $\mathrm{Orb}_{\mathbf{g}}(z) \subset S$.

Let I_0 be the $\overline{P}$-basic interval such that $z \in \mathrm{Int}(I_0)$. By Lemma 1.3.2, there is a unique loop $I_0 \to I_1 \to \ldots \to I_{n-1} \to I_n = I_0$ in the Markov $\mathbf{g}$-subgraph of $\overline{P}$ (see page 105), which we call α, such that $\mathrm{Orb}_{\mathbf{g}}(z)$ and α are associated.

Now we claim that α is simple and negative. Since the index of z as a fixed point of $\mathbf{g}^n$ is -1, $\mathbf{g}^n$ is decreasing on a neighborhood of z and, thus, α is negative. It is not difficult (see Lemma 1.3.3 or Lemma 1.2.6 of [8]) to construct inductively closed intervals $K_n \subset K_{n-1} \subset \ldots \subset K_1 \subset I_0$ satisfying $\mathbf{g}^i(K_j) \subset I_i$ for each $1 \leq i < j$, and $\mathbf{g}^j(K_j) = I_j$. Moreover, $\mathbf{g}^j|_{K_j}$ is a homeomorphism, which is linear by Proposition 2.8.11(b). Since α is negative, $\mathbf{g}^n|_{K_n}$ is decreasing and z is the unique point of $\mathrm{Fix}(\mathbf{g}^n) \cap K_n$. Now suppose that α is the k-repetition (with $k > 1$ odd) of a simple negative loop $I_0 \to I_1 \to \ldots \to I_{n/k-1} \to I_0$, which we call δ. By Lemma 1.3.3, there exists a fixed point w of $\mathbf{g}^{n/k}$ in $K_{n/k}$. In fact, w is unique because $\mathbf{g}^{n/k}$ is decreasing and linear on $K_{n/k}$. Since δ is simple and negative, from Lemma 1.3.5(b) it follows that the period of w is n/k. Since $\mathbf{g}^{n/k}$ is linear and decreasing, w is a repelling fixed point of $\mathbf{g}^{n/k}$. Therefore, there are no points $w' \in K_{n/k} \setminus \{y\}$ satisfying $(\mathbf{g}^{n/k})^i(w') \in I_0$ for $2 \leq i < k$ and $\mathbf{g}^n(w') = w'$. Since $K_n \subset K_{n/k}$, $z = w$. But z is n-periodic and w is n/k-periodic, a contradiction which ends the proof of the claim.

Since $\mathrm{Orb}_{\mathbf{g}}(z) \subset S$, it follows that α is also a simple and negative loop in the Markov $\overline{g}$-subgraph of $\overline{P}$ (see page 105). By Remark 2.8.13 we can use Lemmas 1.3.3 and 1.3.5 and obtain an n-periodic orbit of $\overline{g}$ associated to α. This completes the proof. ∎

2.9 Proof of Theorem D

This section is devoted to prove Theorem D. Before doing it, we will state and prove four technical lemmas, which will be used to construct tree maps with a prescribed set of periods.

Lemma 2.9.1. *Let $f\colon T \longrightarrow T$ be a tree map and let $\mathcal{F} \subset \mathbb{N}$ be finite. Then there exist a tree $S \supset T$ and a tree map $g\colon S \longrightarrow S$ such that* $\operatorname{Per}(g) = \operatorname{Per}(f) \cup \mathcal{F}$.

Proof. It is enough to prove that, for any $n \in \mathcal{F}$, there exists a tree $S \supset T$ and a tree map $g\colon S \longrightarrow S$ such that $\operatorname{Per}(g) = \operatorname{Per}(f) \cup \{n\}$.

Let $n \in \mathcal{F}$ and let $y \in \operatorname{Fix}(f)$. Consider a tree X containing T such that $\operatorname{Cl}(X \setminus T)$ is an n-star whose central point is y, and denote its branches by $[y, b_i]$ for $1 \le i \le n$. Now consider a map $g\colon S \longrightarrow S$ such that $g|_T = f$, $g(b_i) = b_{i+1 \bmod n}$ and g maps linearly $[y, b_i]$ onto $[y, b_{i+1 \bmod n}]$ for each $1 \le i \le n$. It easily follows that $\operatorname{Per}(g) = \operatorname{Per}(f) \cup \{n\}$. ■

Lemma 2.9.2. *Let $p \in \mathbb{N}$, $\lambda \in \mathbb{N} \cup \{0\}$ and $n \in \mathbb{N} \cup \{p2^\infty\}$ such that $\lambda = 0$ when $\mathcal{S}_p(n) \subsetneq \{1\} \cup p\mathbb{N}$. Then there exist a tree S with $|\operatorname{En}(S)| \ge \max\{3, p\}$ and a tree map $g\colon S \longrightarrow S$ such that* $\operatorname{Per}(g) = \mathcal{S}_p(n) \setminus p\{2, 3, \ldots, \lambda\}$.

Proof. Assume first that $\lambda \le 1$. In this case, $p\{2, 3, \ldots, \lambda\} = \emptyset$. By Baldwin's theorem, there exists a $(\max\{3, p\})$-star map such that $\operatorname{Per}(g) = \mathcal{S}_p(n)$. Hence, the lemma follows in this case.

From now on we assume that $\lambda > 1$. Then, since $\mathcal{S}_p(n)$ is an initial segment, from the hypotheses we get that one of the following cases occurs:

(i) either $\mathcal{S}_p(n) = \{1\} \cup p\mathbb{N}$ (then $n = 3p$)

(ii) or $p\mathbb{N} \subsetneq \mathcal{S}_p(n)$ (then $n \notin p\mathbb{N}$).

Assume that $p = 1$. In this case, (ii) does not hold and thus (i) occurs. Therefore, we must construct a tree map $g\colon S \longrightarrow S$ such that $\operatorname{Per}(g) = \mathcal{S}_1(3) \setminus \{2, 3, \ldots, \lambda\} = \mathbb{N} \setminus \{2, 3, \ldots, \lambda\}$. From the definition of Baldwin's orderings, it is not difficult to see that $\mathbb{N} \setminus \{2, 3, \ldots, \lambda\}$ coincides with the following finite union of initial segments of the ${}_{\lambda+1}\!\ge$ ordering:

$$\mathbb{N} \setminus \{2, 3, \ldots, \lambda\} = \bigcup_{1 \le n \le \lambda} \mathcal{S}_{\lambda+1}(\lambda + 1 + n).$$

Then we choose S to be a $(\lambda + 1)$-star and, by Baldwin's theorem, there exists a tree map $g\colon S \longrightarrow S$ such that $\operatorname{Per}(g) = \mathbb{N} \setminus \{2, 3, \ldots, \lambda\}$. Moreover, since $\lambda > 1$ we have $|\operatorname{En}(S)| \ge 3 = \max\{3, p\}$ and the lemma follows in this case.

From now on we assume that $p > 1$. Let l be the least nonnegative integer such that $n + lp > (\lambda + 1)p$. Set $r = n + lp$. We claim that there exist a tree T and a tree map $f\colon T \longrightarrow T$ such that $\operatorname{Per}(f) = \mathcal{S}_p^*(r) \setminus \{2, 3, \dots, \lambda\}$.

Let us prove the claim. Let (T, A, f) be an (r, p, q)-model (see page 48 for a definition) with $q = 0$. The construction of this model corresponds to Case 2 of Proposition 1.9.3. In the notation used there (see paragraph "Definition of T") we have $r = sp + k$, with $s = \lambda + 1$ and $0 < k \leq p$ (recall that $(\lambda + 1)p < r \leq (\lambda + 1)p + p$). From the construction of the tree T it follows that $|\operatorname{En}(T)| = \lambda p + k$. Therefore, $|\operatorname{En}(T)| \geq 2p + 1 \geq \max\{3, p\}$. Moreover, from (1.22) we easily get (see paragraph "Images of the vertices (d)") that $\operatorname{Per}(f) = \mathcal{S}_p^*(r) \setminus \{2, 3, \dots, \lambda\}$ and thus the claim follows.

Set $\mathcal{F} = \mathcal{S}_p(n) \setminus \mathcal{S}_p^*(r)$. When (i) occurs, $r = 3p + lp \in p\mathbb{N}$ and therefore $\mathcal{S}_p^*(r) = \{1\} \cup p\mathbb{N} = \mathcal{S}_p(n)$. On the other hand, if (ii) holds then $\mathcal{S}_p^*(r) = \mathcal{S}_p(r)$ and, by Remark 1.2.3, $\mathcal{F} = \{n\} \cup (\{in + jp : 1 \leq i < p/\text{g.c.d}(p, n), 1 \leq j \leq li\} \setminus \{r\})$. In any case, $\mathcal{F}$ is either empty or finite. Then by Lemma 2.9.1 there exist a tree $S \supset T$ and a tree map $g\colon S \longrightarrow S$ such that $\operatorname{Per}(g) = \operatorname{Per}(f) \cup \mathcal{F} = \mathcal{S}_p^*(r) \cup \mathcal{F} = \mathcal{S}_p(n)$. ∎

Lemma 2.9.3. *Let $g\colon S \longrightarrow S$ be a tree map and let $p \geq 2$ be an integer. Then there exist a tree T such that $T \sqsupset pS$ and a tree map $f\colon T \longrightarrow T$ such that* $\operatorname{Per}(f) = \{1\} \cup p\operatorname{Per}(g)$.

Proof. Let X be a p-star and let y be the central point of X. Choose one endpoint e of S. Let T be the tree obtained by attaching p-many disjoint copies of S to X, each copy attached by identifying e with an endpoint of X. It follows that $T \sqsupset pS$. It is easy to define a map $f\colon T \longrightarrow T$ such that $f(y) = y$ and $\operatorname{Per}(f) = \{1\} \cup p\operatorname{Per}(g)$. The detailed construction has been done –when g is monotone– in the proof of Proposition 1.9.2 (see also Figure 1.3), and it can be directly extended to the case of a continuous map. ∎

Lemma 2.9.4. *Let $\{f_i\colon T_i \longrightarrow T_i\}_{i \in R}$ be a finite set of tree maps such that $T_i \cap T_j = \emptyset$ for $i \neq j$. Then there exist a tree T such that $T \supset T_i$ for each $i \in R$ and a tree map $f\colon T \longrightarrow T$ such that* $\operatorname{Per}(f) = \bigcup_{i \in R} \operatorname{Per}(f_i)$.

Proof. For each $i \in R$, we choose $y_i \in \operatorname{Fix}(f_i)$. Set $X = \bigcup_{i \in R} T_i$. Consider the following equivalence relation $\sim$ on X: for $w, z \in X$, we let $w \sim z$ if either $w = z$ or there exist $i, j \in R$ such that $w = y_i$ and $z = y_j$. Let T be the resulting quotient space. Observe that T can be thought as the tree obtained by gluing together all T_i at the points y_i. Thus we can assume that $T_i \subset T$ for each $i \in R$.

Let us denote by y the only point of T which corresponds to the $\sim$-class of all y_i. Now let $f\colon T \longrightarrow T$ be the only map such that $f|_{T_i} = f_i$ for each

$i \in R$. Observe that f is continuous and well defined. Moreover, $f(y) = y$ and $\operatorname{Per}(f) = \bigcup_{i \in R} \operatorname{Per}(f_i)$. ■

Now we are ready to prove Theorem D.

Proof of Theorem D. We claim that there exist a family $\{T_{\underline{s}}\}_{\underline{s} \in \mathsf{S}}$ of pairwise disjoint trees and, for each $\underline{s} \in \mathsf{S}$, a tree map $f_{\underline{s}} \colon T_{\underline{s}} \longrightarrow T_{\underline{s}}$ verifying:

(I) $\underline{s} \in \Sigma_{T_{\underline{s}}}$

(II) $\operatorname{Per}(f_{\underline{s}}) = \mathcal{K}_{\underline{s}} \cup (\mathcal{I}_{\underline{s}} \setminus \lceil \underline{s} \rceil \{2, 3, \dots, \lambda_{\underline{s}}\})$.

Let us see that the theorem follows from the above claim. Indeed: since S is finite, by Lemma 2.9.4 there exists a tree map $g \colon S \longrightarrow S$ such that

$$\operatorname{Per}(g) = \bigcup_{\underline{s} \in \mathsf{S}} \operatorname{Per}(f_{\underline{s}}) = \bigcup_{\underline{s} \in \mathsf{S}} (\mathcal{K}_{\underline{s}} \cup (\mathcal{I}_{\underline{s}} \setminus \lceil \underline{s} \rceil \{2, 3, \dots, \lambda_{\underline{s}}\}))$$

and $S \supset T_{\underline{s}}$ for each $\underline{s} \in \mathsf{S}$. By (I), $\underline{s}$ verifies (S 1–2) (with $T_{\underline{s}}$ instead of S) for each $\underline{s} \in \mathsf{S}$. And, since $S \supset T_{\underline{s}}$, (S 1) (and obviously (S 2)) remains true when one replaces $T_{\underline{s}}$ by S. Therefore, $\underline{s} \in \Sigma_S$ for each $\underline{s} \in \mathsf{S}$ and the theorem follows.

Let us prove the above claim. Fix $\underline{s} = (p_1, p_2, \dots, p_m) \in \mathsf{S}$. We have $p_i \geq 2$ for $1 \leq i < m$ and $\mathcal{K}_{\underline{s}} = \{p_1, p_1 p_2, \dots, p_1 p_2 \cdots p_{m-1}\}$. Set $k = p_1 p_2 \cdots p_{m-1}$ (or $k = 1$ when $m = 1$) and $p = p_m$. Let M be the set of ${}_{\lceil \underline{s} \rceil}\!\geq$-maximal elements n of $\mathcal{I}_{\underline{s}}$ such that $n \neq 1$. Observe that M is either empty or finite, and $M = \emptyset$ if and only if $\mathcal{I}_{\underline{s}} = \{1\}$. We have

$$\mathcal{I}_{\underline{s}} = \{1\} \cup \bigcup_{n \in M} \mathcal{S}_{\lceil \underline{s} \rceil}(n). \tag{2.7}$$

Moreover, from the hypotheses and the definition of the ${}_{\lceil \underline{s} \rceil}\!\geq$ ordering, we easily get that

(i) $n \in k(\mathbb{N} \cup 2^\infty)$ for each $n \in M$

(ii) for each $n \in M$ it follows that if $\mathcal{S}_{\lceil \underline{s} \rceil}(n) \subsetneq \{1\} \cup \lceil \underline{s} \rceil \mathbb{N}$ then $\lambda_{\underline{s}} = 0$.

Since $\mathcal{S}_{\lceil \underline{s} \rceil}(n) = \{1\} \cup k \mathcal{S}_p(n/k)$ for each $n \in k\mathbb{N}$ (see, for instance, Remark 1.2.2, where this fact is proved when $n/k \notin p\mathbb{N}$), from (ii) we get

(ii') for each $n \in M$ it follows that if $\mathcal{S}_p(n/k) \subsetneq \{1\} \cup p\mathbb{N}$ then $\lambda_{\underline{s}} = 0$.

Moreover, from (2.7) we easily get that

$$\begin{gathered} \mathcal{K}_{\underline{s}} \cup (\mathcal{I}_{\underline{s}} \setminus \lceil \underline{s} \rceil \{2, 3, \dots, \lambda_{\underline{s}}\}) = \\ \{1\} \cup \mathcal{K}_{\underline{s}} \cup k \bigcup_{n \in M} \left(\mathcal{S}_p(n/k) \setminus p\{2, 3, \dots, \lambda_{\underline{s}}\}\right). \end{gathered} \tag{2.8}$$

Fix $n \in M$. Since $n \neq 1$ and (ii') holds, we can use Lemma 2.9.2 to obtain a tree K_n with $|\operatorname{En}(K_n)| \geq \max\{3, p\}$ and a tree map $h_n\colon K_n \longrightarrow K_n$ such that $\operatorname{Per}(h_n) = \mathcal{S}_p(n/k) \setminus p\{2, 3, \ldots, \lambda_{\underline{s}}\}$. Thus from (2.8) we get that

$$\mathcal{K}_{\underline{s}} \cup (\mathcal{I}_{\underline{s}} \setminus \lceil \underline{s} \rceil \{2, 3, \ldots, \lambda_{\underline{s}}\}) = \{1\} \cup \mathcal{K}_{\underline{s}} \cup k \bigcup_{n \in M} \operatorname{Per}(h_n). \tag{2.9}$$

Without loss of generality we can assume that $K_n \cap K_l = \emptyset$ when $n \neq l$. Since M is finite, by Lemma 2.9.4 there exist a tree, which we denote by S_m, such that $K_n \subset S_m$ for each $n \in M$, and a tree map $g_m\colon S_m \longrightarrow S_m$ such that $\operatorname{Per}(g_m) = \bigcup_{n \in M} \operatorname{Per}(h_n)$. Therefore, from (2.9) we have

$$\mathcal{K}_{\underline{s}} \cup (\mathcal{I}_{\underline{s}} \setminus \lceil \underline{s} \rceil \{2, 3, \ldots, \lambda_{\underline{s}}\}) = \{1\} \cup \mathcal{K}_{\underline{s}} \cup k \operatorname{Per}(g_m). \tag{2.10}$$

Note that S_m is not an interval, since K_n is not an interval for $n \in M$. Moreover, since $p \leq \max\{3, p\} \leq |\operatorname{En}(K_n)|$ for each $n \in M$, then

(1) $p \leq |\operatorname{En}(S_m)|$.

Since $p_i \geq 2$ for $1 \leq i < m$, by using $m - 1$ times Lemma 2.9.3 by backwards induction we obtain, for $1 \leq i < m$, tree maps $g_i\colon S_i \longrightarrow S_i$ such that

(2) $S_i \sqsupset p_i S_{i+1}$

(3) $\operatorname{Per}(g_i) = \{1\} \cup p_i \operatorname{Per}(g_{i+1})$.

Since S_m is not an interval, from (2) it easily follows that

(4) S_i is not a star for $1 \leq i < m$.

We define $T_{\underline{s}} = S_1$ and $f_{\underline{s}} = g_1$. From (3) we easily get that $\operatorname{Per}(f_{\underline{s}}) = \{1\} \cup \mathcal{K}_{\underline{s}} \cup k \operatorname{Per}(g_m)$. Consequently, (II) follows from (2.10). On the other hand, (I) holds by (1–2) and (4). This ends the proof of the claim. ■

Chapter 3

Computer experiments

3.1 Introduction

In Section 2.2 we have summarized the dynamic minimality properties of canonical models studied in [3]. If $\mathcal{P}$ is a pattern and (T, A, f) is a canonical model of $\mathcal{P}$ (which is essentially unique), then f minimizes the topological entropy within the class of maps exhibiting $\mathcal{P}$ (Theorem 2.2.1). Moreover, if g is any map which exhibits $\mathcal{P}$ then the set of periods of f is *essentially* contained (up to f-monotone equivalence and period-doubling) on the set of periods of g (see comments on Theorems 2.2.3 and 2.2.4 in page 67). In this setting, several important problems remain open. Among them:

1. Is it always true that $\text{Per}(f) \subset \text{Per}(g)$? In the negative, the periods of what sort of orbits of f can fail to be inherited by g? (for instance, orbits necessarily contained in $V(T)$).
2. In the theory of interval maps, a pattern P is said to *force* a pattern Q if and only if each interval map exhibiting P also exhibits Q. It is a known fact (Theorem 2.6.13 of [8]) that P forces Q if and only if the minimal (or “connect-the-dots”) map f_P corresponding to the pattern P exhibits an invariant set whose pattern is Q. This is not true in the setting of tree maps (see [6]), as the following example shows. Consider a pattern $([T, A], [\theta])$ with $A = \{x_1, x_2, x_3, x_4\}$ and $\theta(x_i) = x_{i+1 \bmod 4}$ for $1 \leq i \leq 4$ such that (T, A) consists of two discrete components $\{x_1, x_2\}$ and $\{x_2, x_3, x_4\}$. On the left side of Figure 3.1 the canonical model (T, A, f) of this pattern is shown. On the right side, a Markov model (T, A, g) exhibiting the same pattern with $g(v) = x_1$ is shown. Both f and g are monotone on each $(A \cup V(T))$-basic interval. Note that f has a periodic orbit R of period 3 contained in $[x_1, x_2] \cup [v, x_3] \cup [v, x_4]$, while each periodic orbit of g is contained in $[x_1, x_3]$. Thus if R' is a

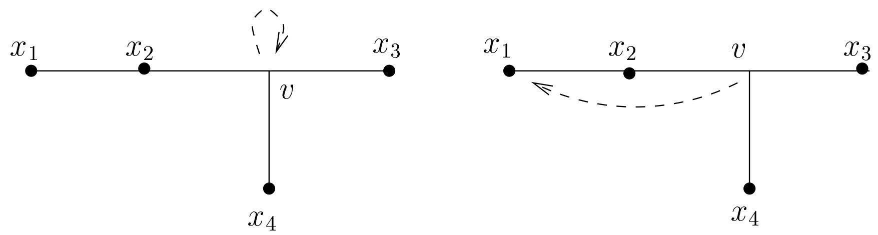

Figure 3.1: Two models exhibiting the pattern $([T, A], [\theta])$ with $A = \{x_1, x_2, x_3, x_4\}$ and $\theta(x_i) = x_{i+1 \bmod 4}$ for $1 \leq i \leq 4$. On the left figure, the canonical model (T, A, f), for which $f(v) = v$. On the right figure, a Markov model (T, A, g) with $g(v) = x_1$.

3-periodic orbit of g then the patterns $([T, R], [f|_R])$ and $([T, R'], [g|_{R'}])$ do not coincide.

Thus the following question arises: what is the appropriate definition of forcing for patterns of tree maps in order that the same property as in the interval case holds?

Partial answers to these questions, as well as potential counterexamples, can be suggested by computer exploration. In this spirit, our main interest have been focused on question 1. We have constructed some computer software in order to explore a wide range of patterns and find potential counterexamples to the relation $\mathrm{Per}(f) \subset \mathrm{Per}(g)$ (in the notation introduced at the beginning of the chapter).

In a spirit of modular programming, we have constructed a lot of self-contained functions which can be used to design a wide variety of several-purpose software. The efficient programming of a part of this machinery needs an important theoretical background. In this chapter we list and explain the source code (written in language C) of the most important functions. When required, we also state and prove some results which have been used either to construct the algorithms or to optimize the execution time. The code of other minor functions, which are not interesting from a mathematical point of view, has been listed in the Appendix.

This chapter is organized as follows. In Section 3.2 we analyze in detail the kernel and the auxiliary tree-managing functions which take part in the algorithm of construction of the canonical model of an arbitrary pattern provided by a user. In Section 3.3 we describe the functions for extracting all the simple loops of a given length from a Markov transition matrix, and we also establish and prove some theoretical tools for the efficient symbolic manipulation of chains and loops. In Section 3.4 we describe the calculus of the Markov transition matrix of a given Markov model. Finally in Section 3.5

we report some experimental results.

3.2 The program "TREES"

3.2.1 Aims and source code of the main program

The input of the program "TREES" is a pattern $(\mathcal{T}, [\theta])$ provided by the user, either from the keyboard or from a disk file. The program calculates a pointed tree (T, A) and an A-monotone tree map $f : T \longrightarrow T$ exhibiting the pattern $(\mathcal{T}, [\theta])$. This notation is fixed from now on until the end of Section 3.2. The output can either be shown or stored on a disk file. The main algorithm implements the recursive construction of the pointed tree (T, A) and the map f given by Theorem 5.1 of [3]. The main source code of the program is listed below:

```
#include "head.h"   /* HEAD FILE */

/* GLOBAL VARIABLES: */

/* INPUT (variables containing the user-defined pattern): */
unsigned int **c,ndc,endpat;
struct pair **orb;

/* OUTPUT (variables containing the calculus of T and f): */
unsigned int *im;
struct pair *namechange;
struct tree MZ;

void main(int argc,char *argv[])
{
 void user(void);
 void ReadPattern(char *);
 void Megazero(void);
 void results(void);

 if(argc>1) ReadPattern(argv[1]); else user();

 Megazero();
 results();
}
```

The function **user** is described in Subsection 3.2.4 and it allows to enter the pattern according to the conventions established in Subsection 3.2.3. Alternatively, the input can be read from a disk file by means of the function **ReadPattern**, which is described in Section A.4 in the Appendix. Next the function **Megazero** computes the pointed tree (T, A) and the map f. This function is described in Subsections 3.2.5 and 3.2.7. Finally the function **results**, described in Subsection 3.2.8, prints out these results or, alternatively, stores them in a disk file.

3.2.2 Global data structures

The head file contains the following source code:

```
#include "stdlib.h"
#include "stdio.h"
#include "alloc.h"
#include "math.h"

struct pair {
 unsigned int e1;
 unsigned int e2;
};

struct tree {
 unsigned int nedges;
 struct pair *edges;
};

void error(int,char**);
```

The `pair` structure consists of a couple of unsigned integers. Variables of type `struct pair` have been used mainly to store two sorts of different objects:

- an edge of a tree whose vertices are labeled by unsigned integers.
- the binary representation of a point of the pattern $\mathcal{P}$ (see Subsection 3.2.3).

The main structure is `tree`, which consists of two fields: an unsigned integer which can be thought as the number of edges of the tree, and a pointer to (or a vector of) `pair` structures, which stores the set of edges of the tree. Observe that no field of the structure is reserved to store the number of vertices. Since the Euler characteristic of a tree is 1, the amount of vertices of any tree equals the amount of edges plus 1. Therefore, we establish the following basic convention: the vertices of each tree will be numbered from 1 to the amount of edges plus 1.

The function **error** is an error message management function, which is frequently called from other functions. It is described in Section A.5 in the Appendix.

3.2.3 Algebraic representation of a pattern

A point of the pattern can be thought as an ordered pair of unsigned integers $\{n, k\}$, where n is a discrete component which contains the point and k is the number of the point inside the n-th component. Obviously this representation, which will be called *binary representation*, is not unique.

The representation of the pattern $(\mathcal{T}, [\theta])$ uses three global variables: `ndc`, `**c` and `**orb`. The unsigned integer `ndc` contains the number of discrete components of the pattern. These discrete components are numbered from 1 to `ndc`. The global variable `**c` is an array of unsigned integers of size $(\mathtt{ndc}+1) \times (\mathtt{ndc}+1)$. For each $1 \leq n \leq \mathtt{ndc}$, `c[0][n]` stores the number of points of the n-th discrete component. The points of this discrete component are numbered from 1 to `c[0][n]`. Given $1 \leq m, n \leq \mathtt{ndc}$ $(n \neq m)$, we have $\mathtt{c}[\mathtt{n}][\mathtt{m}] = k$ $\ (1 \leq k \leq c[0][n])$ if and only if the n-th component intersects the m-th component at the k-th point (of the n-th component). And $\mathtt{c}[\mathtt{n}][\mathtt{m}] = 0$ if and only if the n-th and m-th components have no intersection. Moreover, $\mathtt{c}[0][0] = \max\{\mathtt{c}[0][1], \mathtt{c}[0][2], \ldots, \mathtt{c}[0][\mathtt{ndc}]\}$ and `c[n][0]` stores the number of discrete components intersecting the n-th component. Finally, the diagonal of `**c` does not store any information relative to the pattern: positions `c[n][n]` $(1 \leq n \leq \mathtt{ndc})$ are used as auxiliary variables in some processes, as the one performed by the function **reduction** (see page 140).

The θ-image of each point of the pattern is codified by means of the global variable `**orb`, which is an array of `pair` structures. It consists of $(\mathtt{ndc}+1)$–many vectors $*\mathtt{orb}[\mathtt{n}]$ $\ (0 \leq n \leq \mathtt{ndc})$. The length of the vector $*\mathtt{orb}[\mathtt{n}]$ $\ (1 \leq n \leq \mathtt{ndc})$ is $\mathtt{c}[0][\mathtt{n}] + 1$, while $*\mathtt{orb}[0] = \mathtt{NULL}$. The coding is: $\mathtt{orb}[\mathtt{n}][\mathtt{k}] = \{m, l\}$ if and only if the θ-image of the point $\{n, k\}$ is the point $\{m, l\}$. Positions `orb[n][0]` are free.

The global variable `endpat`, an unsigned integer, stores the amount of points of the pattern $(\mathcal{T}, [\theta])$ plus 1. Note that this number is computed *a posteriori*, since the user does not specify the number of points of the pattern.

The global variable `MZ` is a `tree` structure which contains the output of the calculus of the tree T. Its vertices will be labeled from 1 to $\mathtt{MZ.nedges} + 1$, with the additional assumption that the vertices corresponding to the set A are the ones labeled from 1 to $\mathtt{endpat} - 1$.

Finally, the global variable `*namechange` is a vector of `pair` structures in such a way that if $1 \leq p < \mathtt{endpat}$ then $\mathtt{namechange}[p] = \{n, k\}$ means that the p-th vertex corresponds to the point $\{n, k\}$ in the binary representation of the pattern (the k-th point inside the n-th discrete component).

3.2.4 Pattern input

The function **user** gets the pattern either from the keyboard or from a disk file. It uses extensively all the functions described in Section A.4 in the Apendix. If keyboard input is chosen, it must be taken into account that `ndc` $= 1$ is not accepted (trivial case: T is a star). The function performs several checks of data input coherency (see the comments inside code list). If succeed, then a short function named **reduction** is called in order to test whether the input data correspond to a pattern embedded on a tree (see page 141).

```
#include "conio.h"
#include "io.h"
#include "string.h"
#define INFINIT 32767  /* Upper bound for integers. */
static char *errors[] = {"Inconsistent data",
                         "This is not a tree"};

void user(void)
{
 unsigned int i,j,k,n; char r,fitxer[81];
 int reserve_matrix(unsigned int ***,
                    unsigned int,unsigned int);
 int reserve_pointers_sp(struct pair ***,unsigned int);
 int reserve_vector_sp(struct pair **,unsigned int);
 unsigned int reduction(void);
 int getnum(int,int);
 void ReadPattern(char *);
 void WritePattern(char *);
 char getanswer(char *, char *);

 clrscr();
 r = getanswer("MTP","Manual input (M), read tree (T)
  or pattern (P)");
 if(r != 'M'){
  printf("\nFile name: "); scanf("%s",fitxer);
  ReadPattern(fitxer);
  if(r=='P') return;
  else goto IMAGES;
 }
```

```
printf("Amount of discrete components ? ");
ndc=getnum(2,INFINIT);

if(!reserve_matrix(&c,ndc+1,ndc+1)
|| !reserve_pointers_sp(&orb,ndc+1)) error(3,NULL);

c[0][0]=0;

for(i=1;i<=ndc;i++) {
 printf("\nAmount of points of the %hu-th component ? ",i);
 c[0][i] = getnum(2,INFINIT);
 if(c[0][i]>c[0][0]) c[0][0]=c[0][i]; /* Refreshing c[0][0] */
}

/* Right now, c[0][0]=maximum number of points in a component. */

printf("\nMATRIX OF COMPONENTS PAIRWISE CONNECTION\n");
for(i=1;i<=ndc;i++) {
 c[i][0]=0;
 printf("\n");
 for(j=1;j<=ndc;j++) { int cc;

  /* We set c[i][i]=1 because of the conventions assumed by the
     function REDUCTION, which will be called later: */

  if(i==j) {c[i][j]=1;continue;};

  /* If we know that the j-th component does not intersect the
     i-th component, it is unnecessary to ask for c[i][j]: */

  if((i>j)&&(!c[j][i])) {c[i][j]=0;continue;};

  printf("Point of the %hu-th d.c. contained in the %hu-th c.d. ?
  (0=do not intersect,-1=jump) ",i,j);
  cc = getnum(-1,c[0][i]);
  if(cc == -1){ int s,ss;
   printf("   Jump to: %hu-%hu ? ",j+1,ndc+1);
   s = getnum(j+1,ndc+1);
   for(ss=j;ss<s;ss++){c[i][ss] = 0; if( i>ss && c[ss][i])
    error(1,errors); }
   j = s-1;
  } else if(!cc) { c[i][j] = 0; if( i>j && c[j][i])
   error(1,errors); }
    else { c[i][j] = cc; if(cc) c[i][0]++;
```

```
      /* Refreshing c[i][0] */

   /* We check whether the input matrix is symmetric with
      respect the property for the elements of "being either
      zero or different from zero": */

    if((i>j)&&(!((c[i][j]&&c[j][i])||((!c[i][j])&&(!c[j][i])))))
     error(1,errors);
   }
  }

 /* We check whether some row of the input matrix equals zero
    (a discrete component which does not intersect anything): */

  if(!c[i][0]) error(2,errors);
 }

 /* We check the following pathology ("some discrete component
    contains a point with two different names"): if the i-th and
    j-th components do not intersect, there cannot be a component
    k such that c[k][i] = c[k][j]:  */

 for(i=1;i<=ndc-1;i++) {
  for(j=i+1;j<=ndc;j++) {
   if(c[i][j]) continue;
   for(k=1;k<=ndc;k++) if(c[k][i]&&(c[k][i]==c[k][j]))
    error(1,errors);
  }
 }

 /* Now we call REDUCTION, which returns the amount of components
    of the tree after retracting. If there is no such components,
    it follows that the input matrix **c actually corresponds to
    a tree rather than a graph. */

 if (reduction()) error(2,errors);

 for(i=1;i<=ndc;i++) {
  if(!reserve_vector_sp(&(orb[i]),c[0][i]+1)) error(3,NULL);
 }

IMAGES:;

 printf("\nIMAGES OF THE POINTS OF THE PATTERN:\n");
```

```
 for(i=1;i<=ndc;i++) {
  for(j=1;j<=c[0][i];j++) {

   /* If the point {i,j} has another binary representation
      {k,c[k][i]} with k<i, the image of this point is already
      known and it is unnecessary to get it: */

   n=0;
   for(k=1;k<i;k++) {
    if(c[i][k]==j) {n=1;orb[i][j].e1=orb[k][c[k][i]].e1;
        orb[i][j].e2=orb[k][c[k][i]].e2; break;}
   }
   /* Otherwise, we proceed prompting {i,j}: */

   if(!n) {
    printf("\nOn which component is mapped {%hu,%hu} ? ",i,j);
    orb[i][j].e1=getnum(1,ndc);
    printf("On which point inside %hu ? ",orb[i][j].e1);
    orb[i][j].e2=getnum(1,c[0][orb[i][j].e1]);
   }
  }
 }
 r = getanswer("YN","Do you want to save the pattern on a
 disk file? (Y/N)");
 if(r != 'Y') return;
 printf("\nFile name (extension .PAT recommended) : ");
 scanf("%s",fitxer);
 WritePattern(fitxer);
}
```

The function **ReadPattern**, whose source code can be found in Section A.4 in the Appendix, reads a pattern from a file using the above conventions. It is assumed that the pattern file was created by the function **WritePattern** inside of **user**, and thus the pattern contained in the file *must* be correct. Hence, **ReadPattern** does not perform any checking among the ones performed by **user**.

3.2.5 Construction of the canonical model

In this subsection we explain in detail how the function **Megazero** works in order to construct the canonical model (T, A, f) of the pattern $\mathcal{P}$ provided by the user. The key point of the construction of the tree T is a function called **treeDC**, which is important enough to be analyzed in a special subsection (see Subsection 3.2.6). **Megazero** computes also the A-monotone map f. This task is performed by **TranslateAndImages** (which is called from **Megazero**). This function is described in Subsection 3.2.7.

This subsection is organized in three separated sub-subsections. In *Support functions* we briefly list and explain some short several-purpose routines. In *Nomenclature and description of the procedure* we recall the definitions and results from Section 5.1 of [3]. Finally in *The function "Megazero"* we list the main source code and comment the general purpose of **Megazero**.

Support functions

Here we list the source code of some short functions which are extensively used from all the tree-managing functions described in the sequel.

The function **SamePoints** returns 1 when the binary representations {`i1,i2`} and {`i3,i4`} correspond to the same point of the pattern, and 0 otherwise:

```
char SamePoints(unsigned int i1,unsigned int i2,
                unsigned int i3,unsigned int i4)
{
 if((!i1)||(!i2)||(!i3)||(!i4)) return 0;
 if(i1==i3) return (i2==i4);
 return ((c[i1][i3]==i2)&&(c[i3][i1]==i4));
}
```

The function **val** returns the valence of the vertex `n` in `tree`, with respect to the subtree given by the set of edges m such that `eactiu[m]` $\neq 0$:

```
unsigned int val(unsigned int n,struct tree arbre,int *eactiu) {
 unsigned int i,j=0;
 for(i=1;i<=arbre.nedges;i++) {
  if(!eactiu[i]) continue;
  if(n==arbre.edges[i].e1 || n==arbre.edges[i].e2) j++;
 }
 return j;
}
```

The function **IsRelativeEnd** admits the following parameters: an integer `v`, a vector of `pair` structures `*edges`, an integer `nedges` and two vectors of integers `*nactiu`, `*eactiu`. The couple `nedges`,`*edges` represents a tree (amount of edges and the set of edges, respectively). The couple `*nactiu`, `*eactiu` indicates that we are considering a subtree of the tree {nedges, *edges}, in such a way that: the vertex n ($1 \leq n \leq$ nedges $+ 1$) is deleted if and only if nactiu[n] $= 0$; and the edge n ($1 \leq n \leq$ nedges) is deleted if and only if eactiu[n] $= 0$. The function returns 0 either when v is not an endpoint relative to the considered subtree, or when it is already deleted (nactiu[v] $= 0$). If v is a relative endpoint, then the function returns an integer e, where e is the edge containing v.

```
int IsRelativeEnd(int v,struct pair *edges,int nedges,
                  int *nactiu,int *eactiu)
{
 int e=0,m;

 if(!nactiu[v]) return 0; /* Output 0 if v is deleted. */
 for(m=1;m<=nedges;m++) {
  if(!eactiu[m]) continue;
  if((edges[m].e1==v)||(edges[m].e2==v)) { if(e) return 0; e=m; }
 }
 return e;
} /* IsRelativeEnd */
```

The function **ConvexHull** computes the convex hull of a finite set $Q = P \cup E$, where P is a subset of $A \cup V(T)$ and E is a subset of the set of $(A \cup V(T))$-basic intervals (which here we informally call *edges*). The points of P and the edges of E are called *unerasable*. The rest of edges and the rest of points of $A \cup V(T)$ are called *erasable*. The parameter `tree` stores the tree T on which the function works. There are two possible input configurations for the rest of the parameters:

1. If `n[0]`>0 then the unerasable points of `tree` are exactly `n[1]`, `n[2]`, …, `n[n[0]]`. And, by convention, all the edges are erasable.
2. If `n[0]`$= 0$ then a point k is unerasable if `nactiu[k]`$= 1$, and erasable if `nactiu[k]`$= -1$ (analogously with `eactiu` and the set of edges).

In both cases, the function deletes all the erasable vertices and edges which are not contained in the convex hull of the unerasable ones. The vectors `*nactiu` and `*eactiu` are returned, updated according to the following code: `nactiu[k]`$= 0$ if and only if the k-th vertex is deleted (that is, it does not belong to the convex hull of Q), and analogously with `*eactiu` and the set of edges.

The function iteratively uses **IsRelativeEnd** to search an erasable point x which is an endpoint relative to the tree defined by non-deleted points and edges. When found, x and the unique edge containing x are deleted by updating `*eactiu` and `*eactiu`. The iterative procedure stops when a cycle is completed without deleting anything.

```
void ConvexHull(struct tree arbre,
                int *nactiu,int *eactiu,unsigned int *n)
{
 int IsRelativeEnd(int,struct pair *,int,int *,int *);
 int m,npoints,v,va,e;

 npoints=arbre.nedges+1;

 /* npoints = number of points. Note that it is > 1. */

/* Initializing vectors nactiu and eactiu.
   Code: 1: Unerasable node or edge
         0: Deleted node or edge (non active)
        -1: Erasable node or edge */

 if(n[0]) {
  for(m=1;m<=npoints;m++) nactiu[m]=eactiu[m]=-1;
  for(m=1;m<=n[0];m++) nactiu[n[m]]=1;
 }

/* Reduction process. We delete all the relative ends (and
   their edges) until a cycle is completed without deleting
   anything. */

 v = va = 1;
 do {
   if(nactiu[v]==-1) {
    e=IsRelativeEnd(v,arbre.edges,arbre.nedges,nactiu,eactiu);
    if(e) { nactiu[v]=eactiu[e]=0; va=v; }
   }
   if(++v > npoints) v=1;
 } while(v!=va); /* End of "do" */
} /* ConvexHull */
```

Nomenclature and description of the procedure

The key point of the calculus of the tree T is an iterative procedure that associates, to each discrete component Q of the pattern $(\mathcal{T}, [\theta])$, a tree $T(Q)$ such that $Q = \mathrm{En}(T(Q))$. This tree will be called the *minimal tree* associated to the discrete component Q. Finally, we will obtain T by gluing together all the trees $T(Q)$ by the corresponding intersection points of the discrete components. To describe the algorithm which computes the minimal tree $T(Q)$ we need some notions and results from Section 5.1 of [3] in order to construct the tree T. We start by summarizing them.

Let $g : B \longrightarrow S$ be a map from a finite set B to a tree S. Let S' be the tree obtained from S by adding a new edge $\langle g(a), e_a \rangle$ for each point $a \in B$ so that if $a \neq b$ but $g(a) = g(b)$ then $e_a \neq e_b$. We define a map $g' : B \longrightarrow S'$ such that $g'(a) = e_a$. Finally, let S_g be the tree $\langle g'(B) \rangle_{S'}$. Note that there is a bijection from the elements of B to the endpoints of the tree S_g. In fact, in order to simplify the notation, we will consider that $B = \mathrm{En}(S_g)$. This procedure to obtaining S_g is uniquely determined up to homeomorphisms, and will be called *g-extension*.

Let (S, B) be a pointed tree. From now on, we will not necessarily assume that $B \subset S$, and (S, B) will stand for $(S, B \cap S)$. The set of discrete components of (S, B) will be denoted by $\mathcal{D}(S, B)$. A subset Q of a discrete component of (S, B) such that either $|Q| = |B \cap S| = 1$ or $|Q| \geq 2$ will be called *discrete subcomponent of* (S, B).

Let $([S, B], [\tau])$ be a pattern and let Q be a discrete subcomponent of (S, B). The set of discrete components of $(\langle \tau(Q) \rangle_S, B)$ will be denoted by $\mathcal{S}(Q)$. Note that this definition is independent from the chosen representative (S, B). If there exists $Q' \in \mathcal{S}(Q)$ such that $|Q'| = 1$ then $\mathcal{S}(Q) = \{Q'\}$. Moreover, for each $Q' \in \mathcal{S}(Q)$ with $|Q'| > 1$ there is a unique $P \in \mathcal{D}(S, B)$ such that $Q' = \langle \tau(Q) \rangle_S \cap P$. In particular, Q' is a discrete subcomponent of (S, B). We define a *Q-sequence of the pattern* $([S, B], [\tau])$ to be an infinite sequence $Q_0 Q_1 Q_2 \ldots$ such that:

a) $Q_0 = Q$ is a discrete subcomponent of (S, B)

b) $Q_n \in \mathcal{S}(Q_{n-1})$ for each $n \in \mathbb{N}$.

Now we list the main properties of Q-sequences, extracted from Lemma 5.5 of [3]:

Lemma 3.2.1. *Let $Q_0 Q_1 Q_2 \ldots$ be a Q-sequence of a pattern. Then*

a) $|Q_{n-1}| \geq |Q_n|$ for each $n \in \mathbb{N}$.

b) There is $m \in \mathbb{N} \cup \{0\}$ such that $|Q_m| = |Q_k|$ for all $k \geq m$.

c) If $|Q_{n-1}| = |Q_n|$ for some $n \in \mathbb{N}$ and $Q_0 Q_1 Q_2 \ldots Q_{n-1} Q'_n \ldots$ is another Q-sequence such that $Q'_n \neq Q_n$ then $|Q'_n| = 2$.

Let $\mathbf{Q} = Q_0 Q_1 Q_2 \ldots$ be a Q-sequence of a pattern. We define the *depth* $\delta(\mathbf{Q})$ of $\mathbf{Q}$ to be the least $n \in \mathbb{N} \cup \{0\}$ such that there exists a Q-sequence $Q_0 Q_1 Q_2 \ldots Q_n Q'_{n+1} Q'_{n+2} \ldots$ with $|Q'_j| = |Q_n|$ for all $j > n$. It is well defined by Lemma 3.2.1 (b). The next result corresponds to Lemma 5.6 of [3]:

Lemma 3.2.2. *The following statements hold:*

a) Let $\mathbf{Q} = Q_0 Q_1 Q_2 \ldots$ and $\mathbf{Q}' = Q_0 Q_1 Q_2 \ldots Q_{\delta(\mathbf{Q})-1} Q'_{\delta(\mathbf{Q})} \ldots$ be two Q-sequences. Then $\delta(\mathbf{Q}') \geq \delta(\mathbf{Q})$.

b) The set $\{\delta(\mathbf{Q}) : \mathbf{Q}$ *is a Q-sequence*$\}$ *is finite.*

Let $\underline{Q} = Q_0 Q_1 \ldots Q_n$ and $\underline{P} = P_0 P_1 \ldots P_m$ be sequences of discrete subcomponents. The sequence $Q_0 Q_1 \ldots Q_n P_0 P_1 \ldots P_m$ will be denoted by $\underline{Q}\underline{P}$. The number n will be called the *length* of $\underline{Q}$, and will be denoted by $||\underline{Q}||$. We say that $\underline{Q}$ is *admissible* if there exists a Q_0-sequence starting with $\underline{Q}$ whose depth is larger or equal to $||\underline{Q}|| - 1$. If $\underline{Q}$ is admissible then the number $\delta(\underline{Q}) = \max\{\delta(\mathbf{Q}) : \mathbf{Q} = \underline{Q}\ldots\}$ will be called the *depth* of $\underline{Q}$. It is well defined by Lemma 3.2.2(b). By definition, $\delta(\underline{Q}) \geq ||\underline{Q}|| - 1$.

Let Q be a discrete component of the pattern $(\mathcal{T}, [\theta])$. We construct the minimal tree $T(Q)$ by induction as follows:

Step 0: For each admissible sequence of discrete subcomponents $\underline{Q} = Q Q_1 Q_2 \ldots Q_n$ for which $\delta(\underline{Q}) = ||\underline{Q}|| - 1$, we define $T(\underline{Q})$ to be a $|Q_n|$-star whose endpoints are Q_n (recall that in page 48 a 1-star was defined to be a point).

Step k: Let $k \geq 0$. Assume by induction that, for all admissible sequences $\underline{Q} = Q Q_1 Q_2 \ldots Q_m$ with $\delta(\underline{Q}) \leq ||\underline{Q}|| - 1 + k$, we have defined a tree $T(\underline{Q})$ such that $\mathrm{En}(T(\underline{Q})) = Q_m$. By Step 0, this induction hypothesis is satisfied for $k = 0$. Let $\underline{Q}' = Q Q_1 Q_2 \ldots Q_n$ be an admissible sequence such that $\delta(\underline{Q}') = ||\underline{Q}'|| + k$. For each $Q^* \in \mathcal{S}(Q_n)$, the sequence $\underline{Q}'Q^*$ is admissible and it satisfies $\delta(\underline{Q}'Q^*) \leq \delta(\underline{Q}') = ||\underline{Q}'|| + k = ||\underline{Q}'Q^*|| - 1 + k$. By the induction hypothesis, the tree $T(\underline{Q}'Q^*)$ is defined and $\mathrm{En}(T(\underline{Q}'Q^*)) = Q^*$ holds.

Let $R(\underline{Q}') = \cup_{Q^* \in \mathcal{S}(Q_n)} T(\underline{Q}'Q^*)$, where the union is obtained as the disjoint union of the trees $T(\underline{Q}'Q^*)$ under identification of the common points of the elements of $\mathcal{S}(Q_n)$. Then $(R(\underline{Q}'), B)$ is a pointed tree whose discrete components are $\mathcal{S}(Q_n)$. Now we apply the g-extension construction on $g : Q_n \longrightarrow R(\underline{Q}')$, where $g = \theta|_{Q_n}$, and we define $T(\underline{Q}') = S_g$, which is a tree whose endpoints are Q_n (after doing the corresponding identification). This completes Step k of the induction process.

Remark 3.2.3. If for some admissible sequence $\underline{Q} = Q Q_1 \ldots Q_n$ we have $|Q_n| < 4$, then $T(\underline{Q})$ is a $|Q_n|$-star. This follows from the fact that each tree with less than 4 endpoints is a star. □

The function "Megazero"

A variable `*mz` of type pointer to `tree` structures is declared locally for all functions described in Subsection 3.2.5. For each $1 \leq n \leq$ `ndc`, the tree `mz[n]` is the minimal tree constructed according to the above inductive process, corresponding to the n-th discrete component of the pattern $(\mathcal{T}, [\theta])$ (see comments after the function code):

```
static struct tree *mz

void Megazero(void)
{
  unsigned int *dc,i;
  struct tree treeDC(unsigned int);
  void Sort2(unsigned int, unsigned int *, unsigned int *);
  void TranslateAndImages(void);

  /* Reserve memory space: */

  mz = malloc(sizeof(struct tree)*(ndc+1));if(!mz)
   error(2,NULL);
  dc = malloc((ndc+1)*sizeof(unsigned int));if(!dc)
   error(2,NULL);

  {
    /* The vectors dc and npdc ("discrete component" and "number
       of points per discrete component", resp.) are send to the
       function "Sort2" to sort dc with respect to npdc: */

    unsigned int *npdc;
    npdc = malloc((ndc+1)*sizeof(unsigned int));if(!npdc)
     error(2,NULL);
    for(i=1;i<=ndc;i++) { dc[i]=i;npdc[i]=c[0][i];
                          mz[i].nedges=0; };
                          /* initialization */
    Sort2(ndc,npdc,dc);
    free(npdc);
  }

  /* We successively call "treeDC" in the order
     induced by dc[i] (0 < i < ndc+1)
     (see comments below): */

  for(i=1;i<=ndc;i++)  mz[dc[i]] = treeDC(dc[i]);
  free(dc);

  /* We compute an upper bound for the amount of edges
     which belong to the convex hull of any pair of
     vertices of the minimal tree T, and we store it on
     c[1][1]. This value will be stored on a disk file
     together with the tree T and the map f. It relates
     to the optimization of the row length of the Markov
```

```
    matrix (to be used in further programs). */

  c[1][1]=0;
  for(i=1;i<=ndc;i++) c[1][1]+=mz[i].nedges-c[0][i]+2;

  TranslateAndImages();
  free(mz);
} /* Megazero */
```

The function which calculates the minimal tree associated to a discrete component is called **treeDC**, and it is described in the next subsection. The function **Sort2** (see Section A.3) is used to filter the calls to **treeDC** in increasing order of amount of points per component, since we consider it more efficient. This is due to:

- Essentially, the execution time for the calculus of the minimal tree $T(Q)$ associated to a discrete component Q is a decreasing function of the size of Q.
- The execution time for the calculus of $T(Q)$ can be optimized by using certain information about the previously calculated minimal trees associated to other discrete components different from Q (the exact procedure is given by the criterion 3 of the function **Stop**, see page 140). Therefore, the execution time for the calculus of $T(Q)$ is a decreasing function of the number of minimal trees previously calculated associated to other components different from Q.

Remark 3.2.4. First of all, a value 0 is given to `mz[n].nedges` for all $1 \leq n \leq$ `ndc`. After calling the function **treeDC** with argument n, the tree `mz[n]` is defined and in particular `mz[n].nedges` > 0. Hence, `mz[n].nedges` $= 0$ if and only if `mz[n]` has not been calculated yet. □

After filling the vector `*mz`, the function **TranslateAndImages** (see Section 3.2.7) glues together all the trees `mz[n]` ($1 \leq n \leq$ `ndc`) to obtain the minimal tree T, which is stored on the global variable `MZ`.

3.2.6 The function "treeDC"

This function is the key point of **Megazero**. It uses three important routines, named **Stop**, **NextNodes** and **MakeTree**, which are analyzed in three respective so-called sub-subsections.

Let Q be a discrete component of the pattern $(\mathcal{T}, [\theta])$. We can consider that the set of all Q-sequences is arranged according to a *tree of Q-sequences.* It is an abstract tree, whose vertices, which will be called *nodes*, are discrete subcomponents of the pattern $(\mathcal{T}, [\theta])$. The *root node* (or "first node") of the tree of Q-sequences is the discrete component Q. Finitely many *arrows* (or directed edges) start at Q, each of them ending at a node which is an element of $\mathcal{S}(Q)$. Given one of these nodes Q', finitely many arrows start at Q', each of them ending at a node which is an element of $\mathcal{S}(Q')$. And so on. Proceeding in this way we obtain an "infinite tree" which expands from the root node Q. Then any Q-sequence is an infinite sequence $Q_0Q_1Q_2\ldots$ such that Q_n is a node of the tree of Q-sequences for all $n \geq 0$, $Q_0 = Q$ and, for each $n \geq 0$, there is an arrow from Q_n to Q_{n+1} in the tree of Q-sequences.

With these definitions, we can do a complete description of the computer implementation of the algorithm introduced in page 128. Previously to all the functions of Subsection 3.2.5, we have this piece of code:

```
struct node {
 unsigned int dc;
 unsigned int *points;
 struct node **arrows;
 unsigned int narrows;
 int arrow;
 struct tree local;
 struct node *back;
} *start, *actual;
```

It is the declaration of a structure `node`, which is local to the set of all the functions of Subsection 3.2.5. This structure is suitable for the computer representation of a node of the tree of Q-sequences. Next we analyze in detail the contents of each field.

Let Q' be a node of the tree of Q-sequences. In particular, Q' is a discrete subcomponent of the pattern $(\mathcal{T}, [\theta])$.

1. `dc` is an integer between 1 and `ndc`, which stores the discrete component of which Q' is a discrete subcomponent.
2. `*points` is a vector of integers such that $Q' = \{\texttt{points}[1], \texttt{points}[2], \ldots, \texttt{points}[\texttt{points}[0]]\}$. That is, it stores the subset of points of the discrete component `dc` conforming the subcomponent Q', labeled in terms of the internal ordering of the points of the discrete component `dc`. In particular, $1 \leq \texttt{points}[\texttt{n}] \leq \texttt{c}[0][\texttt{dc}]$ for $0 \leq n \leq \texttt{points}[0]$ (see Subsection 3.2.3).
3. `narrows` is an unsigned integer which equals the amount of arrows starting from the node Q' in the tree of Q-sequences.
4. `**arrows` is a vector of pointers to `node` structures, of length `narrows` (from `*arrows[0]` to `*arrows[narrows-1]`), which can be thought as

the set of arrows starting from the node Q' in the tree of Q-sequences. Each of these pointers stores the memory address which allocates a `node` structure corresponding to an element of $\mathcal{S}(Q')$.

5. `arrow` is an integer in the range [0,`narrows`] which, during the construction of the tree of Q-sequences, marks our situation on the node Q'. A value `NULL` for `arrows` indicates that we arrive to Q' for the first time and, if it is possible, we must generate the Q-subsequent nodes and then move to the node whose address is `arrows[0]`. A value $0 \leq$ `arrow` $<$ `narrows` $- 1$ indicates that we already have moved to the node whose address is `arrows[arrow]` and that the next "open problem" is to compute the minimal tree corresponding to the node whose address is `arrows[arrow+1]`. Finally, a value `arrow` $=$ `narrows` $- 1$ indicates that we have completed the generation of all the subtree whose root node is Q'. The generation of the tree of Q-sequences is described in detail at page 133.
6. `local` is a `tree` structure containing (when it is known) the minimal tree $T(\underline{Q})$ (defined at page 128) associated to the admissible sequence $\underline{Q} = \overline{Q \ldots} Q'$ formed by all predecessors of Q' in the tree of Q-sequences. We recall that the set of endpoints of this tree is exactly Q'. By construction, the tree `local` will be *consistent*. This means that: the vertices will be numbered from 1 to `local.nedges` $+ 1$, the vertices between 1 and `points[0]` will be exactly the endpoints and they will we sorted in such a way that, if $1 \leq n \leq$ `points`[0], then the vertex n represents the point `points[`n`]` of the discrete component `dc`.
7. `*back` is a pointer to a `node` structure containing the address of the node Q'' of the tree of Q-sequences such that there is an arrow from Q'' to Q'.

Also a pair of variables of type pointer to `node` structure, `*start` and `*actual`, are declared. They are local to all the functions of Subsection 3.2.5 and they are the key point of the computer version of the algorithm of page 128. The purpose of the algorithm is to generate the tree of Q-sequences, that is: given a node Q', we want to calculate the tree $T(Q \ldots Q')$. The address of the root node of the tree of Q-sequences is stored at `*start`, while the address of the node Q' is stored at `*actual`. The problem of computing the tree $T(Q \ldots Q')$ may be:

- Already solvable if and only if the trees $T(Q \ldots Q'Q'')$ are calculated for each node $Q'' \in \mathcal{S}(Q')$. In this case, we compute the tree $T(Q \ldots Q')$ and next we free the memory space used by the nodes of $\mathcal{S}(Q')$, which are not yet necessary.
- Unsolvable yet if:

– either the nodes of $\mathcal{S}(Q')$ are still unknown. In this case, we must generate them.
– or the nodes of $\mathcal{S}(Q')$ are known but we do not know the trees $T(Q \dots Q'Q'')$ for each $Q'' \in \mathcal{S}(Q')$. In this case, we must compute them.

This is a general idea about the "local" decisions the function **TreeDC** must take to calculate the tree $T(Q \dots Q')$. Each decision implies a further movement to a node different from Q'. This is why we speak of an "algorithm of generation and wandering on the tree of Q-sequences". These movements are detailed in the sequel.

The function **treeDC** implements the algorithm of generation and wandering on the tree of Q-sequences associated to a $1 \leq$ `component` $\leq$ `ndc` which will be called Q. It returns a `tree` structure containing the minimal tree $T(Q)$. The key lines have been numbered:

```
struct tree treeDC(unsigned int component) {
  int Stop(struct node **),i;
  void MakeTree(struct node **);
  void NextNodes(struct node *);
  void Star(unsigned int, struct tree *);
  struct tree auxiliar;

  /* Trivial case: if Q has less than 4 points,
     T(Q) is a star: */
  if(c[0][component] < 4) {
    Star(c[0][component],&auxiliar);
    return auxiliar;
  }

  /* Initializations corresponding to the root node: */
  start = malloc(sizeof(struct node *)); if(!start)
   error(1,NULL);
  start->dc = component;
  start->points = malloc((c[0][component]+1)
   *sizeof(unsigned int)); if(!start->points)
    error(2,NULL);
  start->points[0] = c[0][component];
  start->arrows = NULL;
  start->back = NULL;
  start->local.edges = NULL;
  for(i=1;i<=start->points[0];i++) start->points[i] = i;
```

```
  /* We move to the root node: */
  actual = start;

  /* And we start the generation and wandering on the
     tree of Q-sequences: */

  while(actual) {                                          /*1*/
    if(actual->arrows) {                                   /*2*/
      (actual->arrow)++;                                   /*3*/
      if(actual->arrow < actual->narrows)
       actual = actual->arrows[actual->arrow];             /*4*/
      else {MakeTree(&actual); actual=actual->back;}       /*5*/
    }
    else {                                                 /*6*/
      if(Stop(&actual)) actual = actual->back;             /*7*/
      else {                                               /*8*/
        NextNodes(actual);                                 /*9*/
        actual->arrow = 0; actual = actual->arrows[0];     /*10*/
      }
    }
  }

  /* Tree T(Q) computed. We copy it into auxiliar and clean: */

  auxiliar.nedges = start->local.nedges;
  auxiliar.edges = start->local.edges;
  free(start->points); free(start);
  return auxiliar;
} /* treeDC */
```

A variable `auxiliar` of type `tree` structure is declared. Its value will be returned as the output of the function. First of all we check that the `component` Q has more than 3 points. Otherwise, by Remark 3.2.3 the calculus is trivial: the function **Star** (see page 137) loads a $|Q|$-star on `auxiliar` and we are done.

Next the function initializes the values of the fields of the `node` structure whose address is contained in the variable `start`, corresponding to the root node of the tree of Q-sequences. Observe that `start->arrows` and `start->back` are given a NULL value. This is essential, as we shall see. The variable `actual` contains the address of the node on which we are placed at each step of the wandering on the tree of Q-sequences. At the beginning, `actual` $=$ `start`.

Assume that we are placed in a node Q' of the tree of Q-sequences (that is, the variable `actual` points to the node describing Q). We define *moving forward* as doing `actual=actual->arrows[narrow]`. We define *moving backward* as doing `actual=actual->back`. Then:

i) Case `actual->arrows=NULL` (line 6 of code). In this case, the nodes subsequent to the node `actual` have not been yet generated. If the function **Stop** detects that the tree $T(Q \dots Q')$ can be directly calculated and it is unnecessary to compute the nodes (see page 137), then it constructs the tree and stores it at `actual->local`. Then we move backward (line 7 of code). Otherwise (line 8), **NextNodes** generates the nodes, then we set `actual->arrow` to 0 and move forward to the first node successor of the current one (line 10).

ii) Case `actual->arrows`$\neq$`NULL` (line 2). In this case, the nodes $0, 1, \dots,$ `narrows-1` subsequent to the node `actual` have already been generated. We increment the counter `actual->arrow` (line 3). If it is smaller than the number of arrows starting at node `actual`, then we move forward to the node `actual->arrows[actual->arrow]` (line 4). Otherwise (line 5), the local trees of all the nodes subsequent to `actual` are defined and thus the function **MakeTree** can generate the tree $T(Q \dots Q')$. Next we move back.

We repeat this procedure until `actual=NULL` (line 1). Note that this will eventually happen since `start->back=NULL` and this value is no longer modified. This algorithm works (i.e., calculates the tree $T(Q)$ spending a finite time) due essentially to two facts:

1. The function **NextNodes** (see page 140) sets the field `arrows` of all generated `node` structures to `NULL`.
2. The function **Stop** detects, in particular, admissible (finite) sequences whose depth equals its length minus 1 (see Step 0 of the inductive process described at page 128). Then Lemma 3.2.2(b) warrants the finiteness of the process.

To have a full view of the algorithm we still need to describe the key functions **Stop**, **NextNodes** and **Maketree**. This is the aim of the following three sub-subsections.

The function "Stop"

Before analyzing the source code of **Stop**, we describe the utility function **OptimizeIfStar** which allows us to minimize the used memory space and optimize the execution time. This function is called from most of the functions invoked during the process of generation and wandering on the tree of

Q-sequences. It is based on Lemma 3.2.1(c). Let Q' and Q'' be nodes such that $Q'' \in \mathcal{S}(Q')$. Assume that the tree $T(Q \ldots Q'Q'')$ is known to be a $|Q''|$-star, and that $|Q'| = |Q''|$. Then from Lemma 3.2.1(c) it follows that the other elements of $\mathcal{S}(Q')$ have cardinality 2. Therefore, when constructing the tree $T(Q \ldots Q')$ by a process of g-extension and identification of endpoints, we obtain a tree which is homeomorphic to $T(Q \ldots Q'Q'')$, and so a $|Q''|$-star.

The function **OptimizeIfStar** uses iteratively this procedure in order to move as back as possible in the tree of Q-sequences. First of all it checks whether `*actual->local` is a star (note that the used criterion is that the amount of edges of the tree `(*actual)->local` equals $|Q''|$). If this is not the case we abandon the optimization. Otherwise, the value of `*actual` is updated until the subcomponent associated to the node `(*actual)->back` has not the same number of points as the one associated to the node `*actual` (observe the extra level of indirection). Before returning, the function **Star** loads a star on the tree `(*actual)->local`, and finally frees the memory space used by the subtree of sequences starting at node `*actual`.

We note that, in particular, the function **OptimizeIfStar** detects admissible sequences whose depth equals the length minus 1. In fact, it translates the node `actual` to the node R of the sequence $Q \ldots R \ldots Q''$ such that $\delta(Q \ldots R) = ||Q \ldots R|| - 1$.

```
void OptimizeIfStar(struct node **actual) {
  if((*actual)->local.nedges != (*actual)->points[0]) return;

  /* We know it is an n-star. Can we continue going back
     by using Lemma 3.2.1 ? : */
  while(((*actual)->back)->points[0] == (*actual)->points[0])
   (*actual) = (*actual)->back;

  /* Here we "prune" from node *actual : */

  if((*actual)->arrows){ struct node *aux;
    aux=*actual; aux->arrow=0;
    for(;;) {
      while(aux->arrow < aux->narrows){
        while(aux->arrows){
          aux = aux->arrows[aux->arrow];
          free(aux->points);
          aux->arrow = 0;
        } /* Here the forward "while" ends */
        if(aux->local.edges) free(aux->local.edges);
        aux = aux->back;
```

```
        free(aux->arrows[aux->arrow]);
        (aux->arrow)++;
      } /* Here the number-of-arrows "while" ends.
           All the successors of *actual have bee pruned. */
      free(aux->arrows); aux->arrows = NULL;
       if(aux == *actual) break;
      aux = aux->back;
      free(aux->arrows[aux->arrow]);
      (aux->arrow)++;
    } /* Here the "for" ends */
  } /* End of "if" */

  /* Now we hang the n-star if it is not there from actual node
     and return: */

  if((*actual)->local.edges) return;
  Star((*actual)->points[0],&((*actual)->local));
  return;
} /* OptimizeIfStar */
```

Star admits two parameters: an unsigned integer `np` and a pointer to `tree` structure `*tree`. It loads an `np`-star on the `tree` structure starting at memory address `*tree`, doing it with a consistent numbering of edges and vertices (recall the definition of consistency given in page 132):

```
void Star(unsigned int np, struct tree *arbre) {
  int i,b;

  arbre->nedges = np - (np == 2);
  arbre->edges = malloc((arbre->nedges+1)*sizeof(struct pair));
  b=arbre->nedges+1;
  for(i=1;i<=arbre->nedges;i++) {
    (arbre->edges[i]).e1 = i;
    (arbre->edges[i]).e2 = b;
  }
} /* Star */
```

The function **Stop** is called if and only if the nodes subsequent to the node `actual` (we call it Q') are not defined yet. It uses four criteria in order to determine whether it is possible to halt the generation of the tree of Q-sequences because the tree $T(Q \ldots Q')$ can be easily calculated. If one of the four criteria holds, the function calculates $T(Q \ldots Q')$ and returns 1.

Otherwise it returns 0. This functions allows us to save time and it warrants the finiteness of the algorithm of generation and wandering on the tree of Q-sequences. **Stop** assumes the parameter `**actual` with an extra level of indirection, since the value of `*actual` can be updated to a previous node.

```
int Stop(struct node **actual) {
 void OptimizeIfStar(struct node **);
 void Star(unsigned int,struct tree *);
 void Subtree(struct node *);
 char SamePoints(unsigned int,unsigned int,
                 unsigned int,unsigned int);
 int i,m,n;
 struct node *aux;

       /******** CRITERION 1: AMOUNT OF POINTS ********/

 if((*actual)->points[0] < 4) {
  Star((*actual)->points[0],&((*actual)->local));
  return 1;
 }

     /***** CRITERION 2: THE IMAGE CONSISTS OF ONE POINT *****/

 m=orb[(*actual)->dc][(*actual)->points[1]].e1;
 n=orb[(*actual)->dc][(*actual)->points[1]].e2;
 for(i=2;i<=(*actual)->points[0];i++) {
  if(SamePoints(m,n,
   orb[(*actual)->dc][(*actual)->points[i]].e1,
   orb[(*actual)->dc][(*actual)->points[i]].e2))
    goto Criteri3;
 }
 (*actual)->local.nedges = (*actual)->points[0];
 OptimizeIfStar(actual);
 return 1;

    /******** CRITERION 3: THE COMPATIBILITY ********/

 Criteri3:;

 if(mz[(*actual)->dc].nedges) {
  Subtree(*actual);
  OptimizeIfStar(actual);
  return 1;
```

```
}

      /******** CRITERION 4: THE STABILITY ********/

  /* Fist of all we look for a cicle */

  aux = *actual;
  n = (*actual)->dc;
  m = (*actual)->points[0];
  while((aux->back)->points[0] == m)
   /* (the number of points in back is not smaller
      than the number of points in actual) */
  {
    aux = aux->back;
    if(aux->dc != n) continue;
    for(i=1;i<=m;i++) if(aux->points[i] !=
     (*actual)->points[i]) goto AnemProvant;

  /* Start of a cycle found. We optimize
     (and hang an n-star) and return */

    *actual = aux;
    (*actual)->local.nedges = (*actual)->points[0];
    OptimizeIfStar(actual);
    return 1;

  AnemProvant:;
  } /* Here "while" ends */

  /* No halt detected */

  return 0;
} /* Stop */
```

- Criterion 1 detects that $|Q'| < 4$ and, by using Remark 3.2.3, equals the tree $T(Q \dots Q')$ to a star. Let Q'' be the node previous to Q'. Note that we do not spend time calling the function **OptimizeIfStar**, since the condition checked by this function ($|Q''| = |Q'|$) is not satisfied. Indeed, the construction of the algorithm implies that the node Q'' does not hold any criteria of the function **Stop**. In particular, it does not hold criterion 1, and hence $|Q''| \geq 4$.
- Criterion 2 detects that the image of all points in Q' is a unique point. In this case $T(Q \dots Q')$ is clearly a star. Then the amount of edges

of the tree `(*actual)->local` is set to $|Q'|$ and **OptimizeIfStar** is called.

- Let P be the discrete component of the pattern $(\mathcal{T}, [\theta])$ such that $Q' \subset P$. By using Remark 3.2.4, criterion 3 determines whether the tree $T(P)$ have been already calculated. In the affirmative, Section 5.3 of [3] shows that $T(Q \dots Q')$ is exactly the convex hull of the points of $Q' \subset P$ inside the tree $T(P)$ (up to homeomorphisms). The convex hull is generated by calling **Subtree** (see page 151). Finally **OptimizeIfStar** is called.
- Criterion 4, in particular, detects admissible sequences whose depth equals the length minus 1. If the sequence ending at the node `actual` is denoted by $QQ_1Q_2 \dots Q_nQ'$, then the criterion looks for some $1 \leq k \leq n$ such that $Q_k = Q'$. If found, it moves to Q_k and loads a star on it. Finally **OptimizeIfStar** is called.

Observe that there are two classes of criteria in the function **Stop**: criteria 1 and 4 are necessary in order to warrant the finiteness of the calculus, and thus are not dispensable. On the other hand, criteria 2 and 3 are purely optimizing.

The function "NextNodes"

NextNodes generates the nodes subsequent to the node `*actual`, which is sent as a parameter. From the construction of the algorithm it follows that this function is called with the input condition `actual->arrows=NULL`, and the node `actual` does not hold any criteria of the function **Stop** (see page 133).

Let Q be the discrete subcomponent loaded at node `*actual`. The subsequent nodes are discrete subcomponents of the convex hull of the image $\theta(Q)$. The first stage of the procedure carried out in **NextNodes** consists of marking the discrete components contained in the convex hull of the discrete components which contain points of $\theta(Q)$. This task is carried out by a short function called **reduction**. Now we list its source code and briefly explain how it works.

```
unsigned int reduction(void) {
 unsigned int i=1,p,j,l,ii=1;

 l=ndc-1; /* (Amount of non-deleted components)-1;
             output parameter */
 do {
```

```
  if(c[i][i]==1) { /* We only work with erasable components */
   p=0;            /* Contact point with the rest of the tree */
   for(j=1;j<=ndc;j++) {   /* For each discrete component */
    if(i==j) continue;
    if(c[j][j]&&c[i][j]) { /* non-deleted we search contact */
     if(p) {if(p!=c[i][j]) goto noui;} else p=c[i][j]; /* unique */
    }
   } if(!p) return l;   /* Component with no contact */
   c[i][i]=0;l--;ii=i;  /* The contact was unique. We delete
                           and update ii */
  }
  noui: if(++i>ndc) i=1;  /* Next component */
 } while (l && (i!=ii));     /* Stop */
 return l;
}
```

This function assumes that the diagonal of the matrix `**c` is marked according to the following code: $\mathtt{c}[\mathtt{n}][\mathtt{n}] \in \{0,1,2\}$ for each $1 \leq n \leq \mathtt{ndc}$. We say that a discrete component n is *erasable* if $\mathtt{c}[\mathtt{n}][\mathtt{n}] = 1$, and we call it *unerasable* if $\mathtt{c}[\mathtt{n}][\mathtt{n}] = 2$. The discrete component n is *deleted* if $\mathtt{c}[\mathtt{n}][\mathtt{n}] = 0$. The exact purpose of **reduction** is to delete all the erasable components which are not contained in the convex hull of the unerasable components.

The function iteratively searches erasable components which are connected to the rest of the tree by exactly one point (terminal components) and it deletes them. The iterative procedure stops when either there is only one remaining non-deleted component ($\mathtt{l} = 1$ in the source code) or when a complete checking of all components does not detect any component to be deleted ($\mathtt{i} = \mathtt{ii}$ in the source code).

The function, of type unsigned integer, returns the amount of components which have not been deleted yet, minus 1. Moreover it modifies the contents of the diagonal of the matrix `**c`, in such a way that: if it has been called with some components marked as unerasable, then $\mathtt{c}[\mathtt{n}][\mathtt{n}] \neq 0$ if and only if the n-th component belongs to the convex hull of the unerasable components; otherwise, the contents of the diagonal of `**c` are not determined.

Incidentally, note that if we call **reduction** with all the components marked as erasable, then it returns 0 if and only if the pattern can be retracted to a point. This fact is used by the function **user** in order to test whether the pattern defined by the user is embedded in a tree.

```
void NextNodes(struct node *actual) {
 unsigned int i,j,m=0,**image,*ord;
 int reserve_matrix(unsigned int ***,
```

```
                   unsigned int,unsigned int);
void free_matrix(unsigned int ***,unsigned int);
void Sort1(unsigned int,unsigned int *);
unsigned int belongs(unsigned int,unsigned int *);
unsigned int reduction(void);

if(!reserve_matrix(&image,ndc+1,c[0][0]+1))
 error(3,NULL);

/* image[i][0] = amount of image points which belong
   to the i-th component. These points are image[i][1],...,
   ,...,image[i][image[i][0]]. We mark all components with
   1, and with 2 (unerasable) the ones containing some image
   point of the starting subcomponent. Note that the image
   contains more than one point by Criterion 2 of Stop. */

for(i=1;i<=ndc;i++) { c[i][i]=1; image[i][0]=0; }

for(i=1;i<=actual->points[0];i++) { int k,l;
  k=orb[actual->dc][actual->points[i]].e1;
  l=orb[actual->dc][actual->points[i]].e2;
  c[k][k]=2; if(!image[k][0]) m++;
  if(!belongs(l,image[k])) { image[k][++image[k][0]]=l; }
}

/* We cannot call REDUCTION with exactly one marked component: */

if(m==1) { actual->narrows=1; goto FILL; }
reduction();

/* Note that after REDUCTION there can be non-relevant components
   (i.e., containing a unique image point that in addition is a
   contact point with other components of the convex hull of the
   image). Now we detect and delete them (labeled with 0). Note
   also that the remaining components contain an image point if
   and only if they are marked with 2. Thus, if a component is
   non-relevant then it is marked with 2.  */

for(i=1;i<=ndc;i++) { int jj=0;
 if(c[i][i]!=2 || image[i][0]>1) continue;
 for(j=1;j<=ndc;j++) { /* i = possible non-relevant component */
  if(c[j][j] && j!=i && c[i][j]) {
   jj=j; if(c[i][j]!=image[i][1]) goto DONOT_DELETE;
  }
```

```
 } if(!jj) error(4,NULL); /* i-th component is, indeed,
                             non-relevant. */
 /* Now we must write the only image point of the i-th
    component on the list of image points of the jj-th
    component (which is known to intersect i-th) and
    finally delete i-th */

 if(!belongs(c[jj][i],image[jj]))
  image[jj][++image[jj][0]]=c[jj][i];
 c[i][i]=image[i][0]=0;
 DONOT_DELETE:;
}

/* Finally, for each component we add to the list of image
   points the set of connection points (with components of
   type 1 or 2) which are not yet in the list because are
   not image of anything. In addition we calculate the amount
   of arrows starting from actual. Thus image[i] becomes the
   set of all points in the i-th discrete component belonging
   to the convex hull of the image \theta(actual) */

actual->narrows=0;
for(i=1;i<=ndc;i++) { if(!c[i][i]) continue;
 actual->narrows++;
 for(j=1;j<=ndc;j++) {
  if(!c[j][j] || j==i || !c[i][j]) continue;
  if(!belongs(c[i][j],image[i]))
   image[i][++image[i][0]]=c[i][j];
 }
}

FILL:;

/* We sort increasingly image[i] for each i: */

for(i=1;i<=ndc;i++) if(image[i][0])
 {c[i][i]=1;Sort1(image[i][0],image[i]);}

/* We fill the arrows starting at actual SORTED IN A SUITABLE
   WAY: i.e., in such a way that the discrete subcomponent
   associated to a node arrow[i] intersects the subcomponent
   associated to a node arrow[j] for some j<i (convention of
   "MakeTree"). This order is stored in the vector ord: */
```

```
 ord=malloc((actual->narrows)*sizeof(unsigned int));
 if(!ord) error(2,NULL);

 /* The first component, i, containing image points is joined
    to the vector ord, and marked with c[i][i]=3: */

 for(i=1;i<=ndc;i++) if(image[i][0])
                      {ord[0]=i;c[i][i]=3;break;}

 /* As above, the first component, j, containing image points
    and intersecting component ord[0] is sent to ord[1], and
    marked with c[j][j]=3. And successively...  */

 j=0;i=1;
  while(i<(actual->narrows)) {
   if(++j>ndc) j=1;
   if(c[j][j]&&(c[j][j]!=3)) { int k;
    for(k=0;k<i;k++) if(c[j][ord[k]])
                      {ord[i]=j;c[j][j]=3;i++;break;}
   }
  };

 /* Reserve memory space and fill: */


 actual->arrows=malloc(actual->narrows*sizeof(struct node *));
  if(!actual->arrows) error(3,NULL);

 for(i=0;i<actual->narrows;i++) {
   actual->arrows[i]=malloc(sizeof(struct node));
    if(!actual->arrows[i]) error(1,NULL);
   if(!reserve_vector(&actual->arrows[i]->points,
                         image[ord[i]][0]+1))
    error(2,NULL);
   actual->arrows[i]->dc=ord[i];
   for(j=0;j<=image[ord[i]][0];j++)
    actual->arrows[i]->points[j]=image[ord[i]][j];
   actual->arrows[i]->back=actual;
   actual->arrows[i]->arrows=NULL;
   actual->arrows[i]->local.edges = NULL;
 }
 free_matrix(&image,ndc+1);
 free(ord);
} /* NextNodes */
```

The function“MakeTree”

This function assumes some input conventions: the parameter `**actual` corresponds to a node Q' of the tree of Q-sequences which does not satisfies the four criteria of the function **Stop** (see page 139) and such that the trees $T(Q\dots Q'Q'')$ are defined for each $Q'' \in \mathcal{S}(Q')$. **MakeTree** calculates the tree $T(Q\dots Q')$ and loads it on the tree structure `(*actual)->local`. The extra level of reference is necessary because before returning we call the function **OptimizeIfStar**, which can modify the value of `*actual` (see page 135).

The function**MakeTree** reserves room for a vector `*edgesl` of pair structures with an upper bound of memory necessary to contain $T(Q\dots Q')$. This bound is obtained by adding the amount of edges of all trees $T(Q\dots Q'Q'')$ (since these trees will be glued together) and adding $|Q'|+2$ (each point of the subcomponent Q' will give rise to a new edge as a consequence of the process of g-extension, and 2 edges are added for safety).

In the function **MakeTree** (as well as in its utility functions **glue** and **gExtension**) a tree is represented by two pairs {`nedgesl`,`*edgesl`} and {`nends`,`*ends`}, with the following conventions: the vertices of the tree {`nedgesl`,`*edgesl`} are labeled from 1 to `nedgesl`+1; `*ends` is a vector of pair structures containing `nends` points of the pattern $(\mathcal{T},[\theta])$ (from `ends[1]` to `ends[nends]`). Both vectors are compatible, that is, `nends` $<$ `nedges` $+1$ and, for each $1 \le n \le$ `nends`, the n-th vertex of the tree {`nedgesl`,`*edgesl`} coincides with the `ends[n]`-th point of the pattern.

The tree {`nedgesl`,`*edgesl`} is initially set to `(*actual)->arrows[0]`, and **glue** is called `(*actual)->narrows`-1 times, with parameters {`nedgesl`, `*edgesl`} and `(*actual)->arrows[n]` for $1 \le n <$`(*actual)->narrows`. Note that between two consecutive calls it is not necessary to update the parameters, since **glue** gives an updated output which can be immediately used as a new input (see page 146). Finally we call **gExtension** (described at page 148) and free the memory space of the fields of the node structures `(*actual)->arrows[n]`, which are no long used.

```
void MakeTree(struct node **actual) {
 int i,j=0,k=0,nends,nedgesl;
 struct pair *ends,*edgesl;
 void glue(int *,struct pair *,int *,
           struct pair *,struct node *);
 void gExtension(struct node *,int,struct pair *,
                 int,struct pair *);
 void OptimizeIfStar(struct node **);
```

```
 for(i=0;i<(*actual)->narrows;i++) {
  j += (*actual)->arrows[i]->points[0];
  k += (*actual)->arrows[i]->local.nedges;
 }
 k += (*actual)->points[0]+2;j++;
 ends=malloc(j*sizeof(struct pair));
 if(!ends) error(2,NULL);
 edgesl=malloc(k*sizeof(struct pair));
 if(!edgesl) error(2,NULL);

 nends=(*actual)->arrows[0]->points[0];
 for(i=1;i<=nends;i++) {
  ends[i].e1=(*actual)->arrows[0]->dc;
  ends[i].e2=(*actual)->arrows[0]->points[i];
 }
 nedgesl=(*actual)->arrows[0]->local.nedges;
 for(i=1;i<=nedgesl;i++) {
  edgesl[i].e1=(*actual)->arrows[0]->local.edges[i].e1;
  edgesl[i].e2=(*actual)->arrows[0]->local.edges[i].e2;
 }
 for(i=1;i<(*actual)->narrows;i++)
 glue(&nends,ends,&nedgesl,edgesl,(*actual)->arrows[i]);

 /* Cleaning : */

 for(i=0;i<(*actual)->narrows;i++) {
  free((*actual)->arrows[i]->points);
  free((*actual)->arrows[i]->local.edges);
  free((*actual)->arrows[i]);
 }
 free((*actual)->arrows); (*actual)->arrows = NULL;

 gExtension(*actual,nends,ends,nedgesl,edgesl);
 free(ends); free(edgesl);

OptimizeIfStar(actual);

} /*MakeTree /*
```

Next we describe the tree-management routines which are called from **MakeTree**.

The function **glue** uses {`nedgesl`,`*edgesl`} and {`nends`,`*ends`} as parameters, with the conventions described in page 145. Moreover, `*nod` is the address of a node (we call it Q'') which has the tree `nod->local` consistently defined (recall the definition in page 132). In particular, its endpoints are the vertices labeled from 1 to `nod->points[0]`. It is assumed that there is exactly one point of the vector `*ends` which is also a point of Q''. The purpose is to **glue** together both trees {`nedgesl`,`*edgesl`} and `nod->local` by this common point, storing the resulting tree at {`nedgesl`,`*edgesl`} and updating the vector `*ends` in such a way that it is still compatible with the obtained tree.

Remark 3.2.5. After calling the function **glue**, one of the points of the vector `*ends` is not an endpoint of the obtained tree {`nedgesl`,`*edgesl`} (inside the source code this point corresponds to `ends[i]`). □

```
void glue(int *nends,struct pair *ends,int *nedgesl,
          struct pair *edgesl,struct node *nod)
{
 unsigned int i,j,l,m,n;
 char SamePoints(unsigned int,unsigned int,
                 unsigned int,unsigned int);

 /* We look for i,j such that ends[i] = nod->points[j]
    (i,j are unique by hypothesis): */

 for(i=1;i<=*nends;i++) {
  for(j=1;j<=nod->points[0];j++)
  if(SamePoints(ends[i].e1,ends[i].e2,nod->dc,nod->points[j]))
   goto OK;
 } error(4,NULL);
 OK:;

 /* We look for the only edge l of nod->local containing j; If
    necessary we re-arrange it in such a way that edges[l].e2=j: */

 for(l=1;l<=nod->local.nedges;l++) {
   if(nod->local.edges[l].e1==j) {
   nod->local.edges[l].e1=nod->local.edges[l].e2;
   nod->local.edges[l].e1=j;
   break;
  }
  if(nod->local.edges[l].e2==j) break;
 }

 /* Updating ends (one of them is not an endpoint...): */
```

```
 for(m=1;m<j;m++) {
  ends[*nends+m].e1=nod->dc;ends[*nends+m].e2=nod->points[m];
 }
 n=*nends-1;
 for(m=j+1;m<=nod->points[0];m++) {
  ends[n+m].e1=nod->dc;ends[n+m].e2=nod->points[m];
 }

 /* We re-label the vertices of *edgesl so that the first *nends
    vertices still correspond to the points of the discrete
    component:  */

 for(m=1;m<=*nedgesl;m++) {
  if(edgesl[m].e1>*nends) edgesl[m].e1+=(nod->local.nedges);
  if(edgesl[m].e2>*nends) edgesl[m].e2+=(nod->local.nedges);
 }

 /* We define the new edges *edgesl. */

 for(m=1;m<=nod->local.nedges;m++) {
  n=nod->local.edges[m].e1;
  edgesl[*nedgesl+m].e1=n+*nends-(n>j);
  n=nod->local.edges[m].e2;
  edgesl[*nedgesl+m].e2=n+*nends-(n>j);
 }
 edgesl[*nedgesl+l].e2=i;

 /* Updating sizes: */

 *nends += nod->points[0]-1;
 *nedgesl += nod->local.nedges;
} /* glue */
```

The function **gExtension** will be called if and only if the tree obtained by gluing together all the trees $T(Q\dots Q'Q'')$ for $Q''\in\mathcal{S}(Q')$ has been calculated and loaded on the variables {`nedgesl`,`*edgesl`}. We assume that this has been done in such a way that (see page 145) when calling **gExtension** we have:

- The vector `*ends` contains `nends` points of the pattern $(\mathcal{T},[\theta])$.
- The n-th vertex of the tree {`nedgesl`,`*edgesl`} is, as a point of the pattern, `ends[n]`, for $1\le n\le$ `nends`.

- The set of endpoints of the tree {nedgesl,*edgesl} is contained in the set of vertices $\{1, 2, \ldots,$ nends$\}$. Note that the inclusion is strict: some points of *ends are not endpoints of the tree, since $T(Q \ldots Q'Q'')$ have been glued by identifying some points of *ends: see Remark 3.2.5 (those unwanted vertices will be finally removed by the function **Clean-Valence2**).

gExtension calculates the g-extension of the tree {nedgesl,*edgesl}, with $g = \theta$ and $B = Q'$ (here we are using the notation of page 127). The points of Q' are the points of the vector actual->points. During the process of g-extension we work with the tree {nedgesl,*edgesl}, which, externally (see function **MakeTree** in page 145), has been given enough memory space to contain the new edges. Finally, we allocate the vector actual->local.edges and store on it the obtained g-extension {nedgesl,*edgesl}.

```
void gExtension(struct node *actual,int nends,struct pair *ends,
                int nedgesl,struct pair *edgesl)
{
 int i,j,l,m;
 char SamePoints(unsigned int,unsigned int,
                 unsigned int,unsigned int);
 void CleanValence2(struct pair *,int *,int,int);

 /* Reserve room for the points[0]-many points of actual: */
 for(m=1;m<=nedgesl;m++) {
  edgesl[m].e1 += actual->points[0];
  edgesl[m].e2 += actual->points[0];
 }

 /* Define the points[0]-many new edges: */
 for(m=1;m<=actual->points[0];m++) {
  i=orb[actual->dc][actual->points[m]].e1;
  j=orb[actual->dc][actual->points[m]].e2;
  for(l=1;l<=nends;l++) {
   if(SamePoints(ends[l].e1,ends[l].e2,i,j)) break;
  }
  edgesl[nedgesl+m].e1=m;edgesl[nedgesl+m].e2=l+actual->points[0];
 }
 nedgesl += actual->points[0];

 /* We clean the vertices of valence 2 (note that it is not
   necessary to update ends: now ends is actual->points[0]). */

 CleanValence2(edgesl,&nedgesl,actual->points[0],
```

```
                nends+actual->points[0]);

 /* The g-extension is done. We load it on actual->local : */

 actual->local.nedges=nedgesl;
 actual->local.edges=malloc((nedgesl+1)*sizeof(struct pair));
 if(!actual->local.edges) error(2,NULL);
 for(m=1;m<=nedgesl;m++) {
  actual->local.edges[m].e1=edgesl[m].e1;
  actual->local.edges[m].e2=edgesl[m].e2;
 }
} /* gExtension */
```

The function **CleanValence2** assumes the following input parameters: {`*edgesl`,`*nedgesl`} represent a tree with consistent numbering, in such a way that the vertices are labeled from 1 to `*nedgesl`+1, the ones numbered from 1 to `nends` being exactly the endpoints. Moreover, it is assumed that each vertex labeled from `final`+1 to `*nedgesl`+1 belongs to at least 3 edges of the edge vector `*edgesl`. From `nends`+1 to `final`, there can be vertices of valence 2. The function delete these vertices, modifying at the same time the contents of `*edgesl` and `*nedgesl` to assure the consistence of the returned tree.

```
void CleanValence2(struct pair *edgesl,int *nedgesl,
                     int nends,int final)
{
 int n;

 for(n=nends+1;n<=final;n++) { /* n = vertex to process */
   int v,m,l[2]; /* local variables:
                    v is the valence of n;
                    m is the edge counter;
                    l[0],l[1] are the edges containing n
                    if indeed it has valence 2. */
   v=0;
   for(m=1;m<=*nedgesl;m++) {
    if(edgesl[m].e1==n) {
     if(v==2) goto NEXTVERTEX;
     l[v]=m;v++;continue;
    }
    if(edgesl[m].e2==n) {
     if(v==2) goto NEXTVERTEX;
```

```
      l[v]=m;v++;
      /* By convention, if the edge containing n is
         {n',n}, we reverse it and set {n,n'}: */
      edgesl[m].e2=edgesl[m].e1;edgesl[m].e1=n;
     }
    } /* Here we have found a vertex of valence 2. */

    /* We glue together edges l[0], l[1] obtaining one edge l[0]:
       by the convention, l[0]={n,n'} and l[1]={n,n''} for some
       vertices n' and n''. Then we set l[0]={n'',n'}: */
    edgesl[l[0]].e1=edgesl[l[1]].e2;

    /* Finally we delete edge l[1], updating the names of the
       vertices (subtracting 1 to the indices greater than n): */

    for(m=l[1];m<*nedgesl;m++) {
     edgesl[m].e1=edgesl[m+1].e1-(edgesl[m+1].e1>n);
     edgesl[m].e2=edgesl[m+1].e2-(edgesl[m+1].e2>n);
    }
    /* and update the names of the vertices (with indices greater
       than n) of the remaining edges (from edge 1 to l[1]): */
    for(m=1;m<l[1];m++) {
     if(edgesl[m].e1>n) (edgesl[m].e1)--;
     if(edgesl[m].e2>n) (edgesl[m].e2)--;
    }
    (*nedgesl)--;n--;final--; */ Updating to make 'for' */
    NEXTVERTEX:;              */ working again */
  } /* End of the initial "for"  */
} /* CleanValence2 */
```

We call the function **Subtree** when criterion 3 in the function **Stop** holds (see page 139). In this case, the tree $T(Q \ldots Q')$ corresponding to the node Q' (address `actual`) can be directly extracted from the minimal tree corresponding to the discrete component P such that $Q' \subset P$, which we assume to be defined. More precisely, $T(Q \ldots Q')$ will be the convex hull of the points of Q' inside the tree contained in `mz[actual->dc]`. The function **Subtree** performs this calculus. The process is the same as in **ConvexHull** (described in page 125), and the source code of both functions is almost identical, except that here we can use the fact that `mz[actual->dc]` is a consistent tree: before starting the reduction process, the vertices labeled from 1 to $|P|$ (which are exactly the endpoints) and the edges containing them are marked ("unerasable" when belong to the subcomponent Q', "erasable" otherwise). Thus, during the process of reduction it is not necessary to revise the vertices numbered from 1 to $|P|$, which are automatically updated.

```
void Subtree(struct node *actual) {
 void CleanValence2(struct pair *,int *,int,int);
 int IsRelativeEnd(int,struct pair *,int,int *,int *);
 int *nactiu,*eactiu,m,k,npoints,v,va,e,kk;
 struct tree inici;
 struct pair *edgesl;

/* Easy case: the subtree is all the tree */

 if(actual->points[0]==c[0][actual->dc]) {
  actual->local.nedges=mz[actual->dc].nedges;
  actual->local.edges=malloc(sizeof(struct pair)*
   (actual->local.nedges+1));
  if(!actual->local.edges) error(2,NULL);
  for(m=1;m<=actual->local.nedges;m++){
    actual->local.edges[m].e1=mz[actual->dc].edges[m].e1;
    actual->local.edges[m].e2=mz[actual->dc].edges[m].e2;
  }
  return;
 }

/* initializing and memory allocation */

 inici=mz[actual->dc]; npoints=inici.nedges+1;

 nactiu=malloc(sizeof(int)*(inici.nedges+2));
 eactiu=malloc(sizeof(int)*(inici.nedges+2));

/* Initializing nactiu and eactiu.
   Codi: 1: Unerasable node or edge
         0: Deleted node or edge (non active)
        -1: Erasable node or edge */

 for(m=1;m<=c[0][actual->dc];m++) { nactiu[m]=0;eactiu[m]=-1; }
 for(;m<=npoints;m++) { nactiu[m]=-1;eactiu[m]=-1; }
 for(m=1;m<=actual->points[0];m++) nactiu[actual->points[m]]=1;

 /* Here we mark the edges containing endpoints, as its
    endpoints: */

 for(m=1;m<=c[0][actual->dc];m++) {
```

```
   for(k=1;k<=inici.nedges;k++) {
    if((inici.edges[k].e1==m)||(inici.edges[k].e2==m)) {
     eactiu[k]=nactiu[m];
     break;  /* An endpoint belongs to a unique edge */
    }
   }
  }

/* Process of reduction. We delete the relative endpoints
   (and the edges containing them) until a cycle is
   completed without deleting anything.  */

 v = va = c[0][actual->dc]+1;
 do {
   if(nactiu[v]==-1) {
    e=IsRelativeEnd(v,inici.edges,inici.nedges,nactiu,eactiu);
    if(e) { nactiu[v]=0; eactiu[e]=0; va=v; }
   }
   if(++v > npoints) v=c[0][actual->dc]+1;
 } while(v!=va); /* Final of "do" */

/* Extracting the subtree:

   Step 1: Labeling non deleted nodes. At the end k = amount of
   non deleted nodes. We reserve memory to store the subtree. */

 k=0; for(m=1;m<=npoints;m++) { if(nactiu[m]) {nactiu[m]=++k;} }

/* Step 2: Loading the subtree on edgesl. */

 edgesl=malloc(sizeof(struct pair)*(k+1));
 if(!edgesl) error(2,NULL);
 kk=k; k=0;
 for(m=1;m<=inici.nedges;m++) {
  if(eactiu[m]) {
   edgesl[++k].e1=nactiu[inici.edges[m].e1];
   edgesl[k].e2=nactiu[inici.edges[m].e2];
  }
 } if(kk != k+1) error(4,NULL);

/* Step 3: Cleaning vertices of valence 2. */

 CleanValence2(edgesl,&k,actual->points[0],k+1);
```

```
/* Step 4: Loading the tree on actual->local. */

 actual->local.nedges=k;
 actual->local.edges=malloc(sizeof(struct pair)*(k+1));
 if(!actual->local.edges) error(2,NULL);
 for(m=1;m<=k;m++) {
  actual->local.edges[m].e1=edgesl[m].e1;
  actual->local.edges[m].e2=edgesl[m].e2;
 }
 free(edgesl);free(nactiu);free(eactiu);
} /* Subtree */
```

3.2.7 Calculus of the A-monotone map f

The function **TranslateAndImages** performs two tasks. First it translates the vector of trees `*mz` (`[mz[n]` is the minimal tree corresponding to the n-th discrete component of the pattern) into a unique `tree` structure (global variable `MZ`), in such a way that, if the pattern has k-many points, then the vertices of `MZ` labeled from 1 to k are exactly the points of the pattern. The global variable `endpat` is given the value $k+1$. Thus, inside the tree `MZ` the vertices which belong to A are the ones labeled from 1 to `endpat` -1. To do this, the following intermediate variables are defined and memory-allocated: a matrix of `pair` structures `**name1` and three vectors of unsigned integers `*w`, `*ww` and `*vector`. Then the process of translation and re-labeling is performed in two steps:

1. Given a point x of the pattern with binary representation $\{i,j\}$ (which is not unique), then `name1[i][j]={k,l}`, where k is the least discrete component containing x, and l is the label of x in terms of the (internal) order of points in the k-th component. Note that `name1[i][j]` is another binary representation of the point x.
2. `w[j]` is the accumulated sum of the amount of points of the i-th discrete component, from $1 \le i \le j-1$; `ww[j]` is the accumulated sum of the amount of vertices of the minimal tree corresponding to the i-th discrete component, from $1 \le i \le j-1$. Then we fill `*vector` with 0's and use a short technical function called **name2** which, by using `w`, `ww` and `name1`, transforms the binary representation of a point $\{i,j\}$ into a positive integer which marks in which position of `*vector` we will put a 1. The listing of **name2** is

```
unsigned int name2(unsigned int dc,unsigned int p,
                   unsigned int *w,unsigned int *ww,
```

```
                        struct pair **name1)
{
 return (p<=c[0][dc]) ? (w[name1[dc][p].e1]+name1[dc][p].e2) :
                         (w[ndc+1]+ww[dc]+p);
}
```

Next we do a drag sum of `*vector` (i.e. `vector[i]+=vector[i-1]`) and the label finally assigned to a point with binary representation $\{i,j\}$ is `vector[name2(i,j,w,ww,name1)]`.

Also the global variable `*namechange` is reserved and filled, in such a way that $\texttt{namechange}[m] = \{n,l\}$ if and only if m is the label, as a vertex of the new tree MZ, of a point whose binary representation was $\{n,l\}$ (that is, `*namechange` gives the inverse of the above assignment).

The second purpose of this function is to calculate the f-image of all the vertices of the tree MZ for an A-monotone map f. The function fills the vector of integers `*im` in such a way that `im[n]=m` if and only if the f-image of the n-th vertex is the m-th vertex ($1 \leq n, m \leq \texttt{MZ.nedges} + 1$). The images of the vertices labeled from 1 to `endpat-1` are a part of the information provided by the user (see Subsection 3.2.3), and we only have to translate the binary representations of the matrix `**orb` into the new numbering. The function uses Remark 1.2.1 in order to calculate the images of the remaining vertices.

Now we list the source code of the function **TranslateAndImages**.

```
void TranslateAndImages(void) {
 extern struct pair *namechange;
 unsigned int belongs(unsigned int,unsigned int *);
 struct pair **name1;
 unsigned int i,j,k,l,*w,*ww,*vector;
 unsigned int val(unsigned int,struct tree,int *);
 unsigned int name2(unsigned int,unsigned int,unsigned int *,
                  unsigned int *,struct pair **);
 int *nactiu,*eactiu,
     reserve_vector(unsigned int **,unsigned int);
 void ConvexHull(struct tree,int *,int *,unsigned int *);

 /* Allocating the tree we want to create, MZ : */

 MZ.nedges=0;for(i=1;i<=ndc;i++) MZ.nedges+=mz[i].nedges;
 MZ.edges=malloc(sizeof(struct pair)*(MZ.nedges+1));
 if(!MZ.edges) error(2,NULL);

 /* Table of pairs **name1: ndc-many rows. k-th row has
    length = amount of points of k-th component.
```

```
   name1[i][j].e1 = least component containing the point {i,j}.
   name1[i][j].e2 = point inside that component.    */

name1=malloc(sizeof(struct pair *)*(ndc+1));
if(!name1) error(3,NULL);
for(i=1;i<=ndc;i++) {
 name1[i]=malloc(sizeof(struct pair)*(c[0][i]+1));
 if(!name1[i]) error(2,NULL);
}

/* We check wheter it is necessary to change the name of
   the point {i,j} (component i,point j). How?: if component
   i intersects any other component l (with l<i) by the point
   j, we rename it {l,c[l][i]}. Otherwise we keep the name. */

for(j=1;j<=c[0][1];j++) {name1[1][j].e1=1;
                         name1[1][j].e2=j;}  /* obvious. */
for(i=2;i<=ndc;i++) {
 for(j=1;j<=c[0][i];j++) {
  c[i][0]=i-1;   /* length of the vector sent to BELONGS */
  l=belongs(j,c[i]);
  if(!l) {name1[i][j].e1=i; name1[i][j].e2=j; continue;}
  name1[i][j].e1=l; name1[i][j].e2=c[l][i];
 }
}

/* w[i]=accumulated sum of the points of the pattern until
   component i-1.
   ww[i]=accumulated sum of vertices of the tree until
   component i-1.
   *vector=translation vector, initially filled with 0's.
   name2(i,j,w,ww,name1) = function which, by using w,ww,name1,
   transforms the name of a point {i,j} into an integer which
   marks in which position of *vector we will put a 1 (see
   function NAME2).  */

if(!reserve_vector(&w,ndc+2)) error(2,NULL);
if(!reserve_vector(&ww,ndc+2)) error(2,NULL);
w[1]=ww[1]=0;
for(i=2;i<=ndc+1;i++) { w[i]=w[i-1]+c[0][i-1];
                        ww[i]=ww[i-1]+mz[i-1].nedges+1; }
k=w[ndc+1]+ww[ndc+1]+1;
if(!reserve_vector(&vector,k)) error(2,NULL);
```

```
for(j=1;j<=k;j++) vector[j]=0;
for(i=1;i<=ndc;i++) {
 for(j=1;j<=mz[i].nedges+1;j++) vector[name2(i,j,w,ww,name1)]=1;
}

/* Finally, we do a drag sum of *vector : */

for(i=2;i<=k;i++) vector[i]+=vector[i-1];

/* We want vector[name2(point)] to be the final
   name of a point: */

k=endpat=0;
for(i=1;i<=ndc;i++) {
 for(j=1;j<=mz[i].nedges;j++) {
  MZ.edges[++k].e1=l=vector[name2(i,mz[i].edges[j].e1,
  w,ww,name1)];
  if(mz[i].edges[j].e1<=c[0][i] && l>endpat) endpat=l;
  MZ.edges[k].e2=l=vector[name2(i,mz[i].edges[j].e2,
  w,ww,name1)];
  if(mz[i].edges[j].e2<=c[0][i] && l>endpat) endpat=l;
 }
}

endpat++;

/* Allocate and fill *namechange: */

namechange=malloc(sizeof(struct pair)*(endpat));
if(!namechange) error(2,NULL);
for(i=1;i<=ndc;i++) {
 for(j=1;j<=c[0][i];j++) {
  k=vector[name2(i,j,w,ww,name1)];
  namechange[k].e1=i; namechange[k].e2=j;
 }
}

/* Allocate *im and translate the images of the points
   of the pattern: */

if(!reserve_vector(&im,MZ.nedges+2)) error(2,NULL);
for(i=1;i<=ndc;i++) {
 for(j=1;j<=c[0][i];j++) {
  if(name1[i][j].e1==i) im[vector[name2(i,j,w,ww,name1)]]=
```

```
  vector[name2(orb[i][j].e1,orb[i][j].e2,w,ww,name1)];
 }
}

free(w);free(ww);free(vector);
for(i=1;i<=ndc;i++) free(name1[i]); free(name1);

/* Calculate the images of the remaining points
   by using Remark 1.2.1 */

nactiu=malloc(sizeof(int)*(MZ.nedges+2));
if(!nactiu) error(2,NULL);
eactiu=malloc(sizeof(int)*(MZ.nedges+2));
if(!eactiu) error(2,NULL);

for(i=endpat;i<=MZ.nedges+1;i++) { unsigned int v[4];

/* v[1], v[2] and v[3] will be the images of the 3 points of the
   pattern which determine the image of our point i. */

/* k = initial edge (the first one we find contains our point
       of input to do/while.
   kk = edge during do/while
   jj = point during do/while  */

 v[0]=0;
 for(k=1;k<=MZ.nedges;k++) { int kk,jj;
   if(MZ.edges[k].e1!=i && MZ.edges[k].e2!=i) continue;
   jj = (MZ.edges[k].e1==i) ? MZ.edges[k].e2 : MZ.edges[k].e1;
   kk=k;
   do {
     if(MZ.edges[kk].e1==jj || MZ.edges[kk].e2==jj)
     jj=(MZ.edges[kk].e1==jj)?MZ.edges[kk].e2:MZ.edges[kk].e1;
     if(++kk>MZ.nedges) kk=1;
    } while(jj>=endpat);
   v[++v[0]]=im[jj];
   if(v[0]==3) break;
 }                /* Points v[1], v[2], v[3] found. */

/* If two points coincide, the intersection of the convex hulls
   is trivially determined (see Remark 1.2.1):  */

 if(v[1]==v[2] || v[1]==v[3]) { im[i]=v[1]; continue; }
 if(v[2]==v[3]) { im[i]=v[2]; continue; }
```

```
 /* Now we can assume the 3 points are different. If one of
    them has valence 2 in the convex hull, it is the desired
    intersection point. Otherwise, the intersection point
    consists of the unique point of valence 3 in the
    convex hull. */

  ConvexHull(MZ,nactiu,eactiu,v);

  for(j=1;j<4;j++) if(val(v[j],MZ,eactiu)==2)
                    {im[i]=v[j]; continue;}
  for(j=1;j<4;j++) nactiu[v[j]]=0;
  for(j=1;j<=MZ.nedges+1;j++) {
   if(!nactiu[j]) continue;
   if(val(j,MZ,eactiu)==3) { im[i]=j; break; }
  }
 }

 free(nactiu);free(eactiu);
} /* TranslateAndImages */
```

3.2.8 Output results

The function **results** is the ending function of the program: it prints the contents of the global variables `MZ` and `*im` (i.e., the tree T and the A-monotone map f respectively), and it reports the name change of the points of the set A (as vertices of the tree `MZ`) with respect to the binary representation provided by the user, by exhibiting the contents of the global variable `namechange`. All this information (except the contents of `namechange`) can be saved on a disk file, for further purposes.

```
#include "conio.h"
#include "io.h"
#include "string.h"

char *err_mgz[]={
  "opening write file of the tree",
  "of tree writing",
  };

void results(void) {
 int getnum(int,int),reserve_vector(unsigned int **,
                                    unsigned int);
```

```
char getanswer(char *,char *),fitxer[81];
int i,pos(int,int *);

if(getanswer("YN","\nDo you want to see the tree T?
  (Y/N)")=='Y') {
 printf("\n");
 for(i=1;i<=MZ.nedges;i++) printf("%d: {%d,%d} /
 ",i,MZ.edges[i].e1,MZ.edges[i].e2);
 if(getanswer("YN","\nDo you want translation to the original
    numbering? (Y/N)")=='Y') {
   printf("\n");
   for(i=1;i<endpat;i++) printf("%d -> {%d,%d} /
   ",i,namechange[i].e1,namechange[i].e2);
 }
 if(getanswer("YN","\nDo you want the images of the points of A?
  (Y/N)")=='Y') {
  printf("\n");
  for(i=1;i<=MZ.nedges+1;i++) printf("f(%d)=%d,",i,im[i]);
 }
}

free(namechange); /* Purely informative variable:
                     not written on the disk file. */

/* Order of writing of the information in the disk file:
   - MZ.nedges
   - Edges ( (MZ.nedges)-many pair structures )
   - endpat (unsigned int indicating that the first endpat-1
     vertices of T belong to the original pattern).
   - Images of the vertices of T (MZ.nedges+1 unsigned int).
   - c[1][1] (unsigned int which has been computed in the function
     MEGAZERO, containing an upper bound for the length of a row
     of the further Markov matrix). Since this value depends on
     variables which are local on Phase 1 (and thus unknown on
     Phase 2), we must write it on disk. */

if(getanswer("YN","\nDo you want to save it on a disk file ?
 (Y/N)")=='Y') {
 FILE *fp;
 printf("\nFile name (extension .MGZ recommended) :");
 scanf("%s",fitxer);
 if(!access(fitxer,0) &&
   (getanswer("YN","The file already exists. Proceed? (Y/N)")
    != 'Y')) exit(0);
```

```
  if((fp=fopen(fitxer,"wb")) == NULL) error(1,err_mgz);
  if(fwrite(&MZ.nedges,sizeof(unsigned int),1,fp)!=1)
   error(2,err_mgz);
  for(i=1;i<=MZ.nedges;i++) {
   if(fwrite(&MZ.edges[i],sizeof(struct pair),1,fp)!=1)
    error(2,err_mgz);
  }
  if(fwrite(&endpat,sizeof(unsigned int),1,fp)!=1)
   error(2,err_mgz);
  for(i=1;i<=MZ.nedges+1;i++) {
   if(fwrite(&im[i],sizeof(unsigned int),1,fp)!=1)
    error(2,err_mgz);
  }
  if(fwrite(&c[1][1],sizeof(unsigned int),1,fp)!=1)
   error(2,err_mgz);
  fclose(fp);
 }
}
```

3.3 Extraction of simple loops from Markov transition matrices. Symbolic manipulation of chains

Let (T, Q, f) be a Markov model (that is, f is Q-monotone and $V(T) \subset Q$). Let $\mathcal{K} = \{K_1, K_2, \dots, K_l\}$ be the set of Q-basic intervals. Given $n \in \mathbb{N}$, we want to construct all the simple loops in the Markov f-graph of Q having length n. The aim of this section is to explain the algorithms we have used to perform this task. Before discussing the computer implementation of the algorithms, we proceed to establish some concepts and theoretic tools. From now on, *Markov loop* will stand for "loop in the Markov f-graph of Q".

To the Markov f-graph of Q we associate an "equivalent" object, the *Markov transition matrix* of (T, Q, f), which is a matrix `A` of non-negative integers, with l-many variable-length rows, such that:

1. `A[i][0]` is the number of Q-basic intervals which are f-covered by K_i.
2. The Q-basic intervals f-covered by a Q-basic interval K_i are K_j, where `j=A[i][k]` for $1 \leq k \leq$`A[i][0]`.

We say that a Q-basic interval K is *relevant* if there is some simple Markov loop starting at K. Otherwise we say that K is *irrelevant*. Observe that K is irrelevant if and only if no Markov loop passes through K. Let $\mathcal{K}' = \{L_1, L_2, \dots, L_m\}$ be the set of relevant Q-basic intervals. Obviously

$m \leq l$. This notation is fixed for the rest of this section. We define the *reduced Markov matrix* of (T, Q, f) as a matrix `A` of positive integers, with m-many variable-length rows, such that:

1. `A[i][0]` is the number of relevant Q-basic intervals which are f-covered by L_i.
2. The relevant Q-basic intervals f-covered by a relevant Q-basic interval L_i are L_j, where `j=A[i][k]` for $1 \leq k \leq$`A[i][0]`.
3. `A[i][j]`$<$`A[i][k]` when $j < k$.

In fact, the algorithm of extraction of simple Markov loops works using the reduced Markov matrix rather than the Markov transition matrix.

We define a *chain of length* $n > 1$ to be an ordered set of n natural numbers. We consider the set of chains of length n endowed with a natural lexicographic order. That is, $(I_0, I_1, \ldots, I_{n-1}) < (J_0, J_1, \ldots, J_{n-1})$ if and only if:

(i) either $I_0 < J_0$

(ii) or there exists $0 < j < n$ with $I_i = J_i$ for $0 \leq i \leq j-1$ and $I_j < J_j$.

For any chain $\alpha = (I_0, I_1, \ldots, I_{n-1})$ we define $\sigma(\alpha)$ to be the chain $(I_1, I_2, \ldots, I_{n-1}, I_0)$. The *concatenation* of two chains $\alpha = (I_0, I_1, \ldots, I_{n-1})$ and $\beta = (J_0, J_1, \ldots, J_{k-1})$ is defined to be the chain $\alpha\beta = (I_0, I_1, \ldots, I_{n-1}, J_0, J_1, \ldots, J_{k-1})$, of length $n+k$. The length of a chain α will be denoted by $|\alpha|$. If $k \in \mathbb{N}$ then the concatenation $\alpha\alpha\ldots\alpha$ (k times) will be denoted by α^k. We will say that a chain α is *repetitive* if there is a chain β and an integer $k > 1$ such that $\alpha = \beta^k$. Otherwise we will say that α is *simple*.

Lemma 3.3.1. *Let α and β be chains such that $\alpha\beta = \beta\alpha$ and $|\alpha| < |\beta|$. Then there exist two chains α' and β' such that:*

a) $\beta = \alpha'\beta' = \beta'\alpha'$

b) $\alpha \in \{\alpha', \beta'\}$

c) $|\alpha'| \leq |\beta'|$ *and* $\max\{|\alpha'|, |\beta'|\} < \max\{|\alpha|, |\beta|\}$

Proof. By comparing each element of the chains $\alpha\beta$ and $\beta\alpha$, since $|\alpha| < |\beta|$ it follows that $\beta = \alpha\gamma$ for a chain γ of length $|\beta| - |\alpha|$. Moreover, $\alpha\gamma\alpha = \beta\alpha = \alpha\beta = \alpha\alpha\gamma$, and thus $\beta = \gamma\alpha = \alpha\gamma$. Consider two chains α' and β' such that $\{\alpha', \beta'\} = \{\alpha, \gamma\}$ and $|\alpha'| \leq |\beta'|$. Clearly, (a), (b) and the first part of (c) hold. Moreover, $\max\{|\alpha'|, |\beta'|\} = \max\{|\alpha|, |\gamma|\} = \max\{|\alpha|, |\beta| - |\alpha|\} < \max\{|\alpha|, |\beta|\}$. ■

Lemma 3.3.2. *Let α be a chain of length n. Then α is repetitive if and only if there exists $0 < k < n$ such that $\alpha = \sigma^k(\alpha)$.*

Proof. It is obvious that if α is repetitive then there exists such a k. Assume that there is $0 < k < n$ with $\alpha = \sigma^k(\alpha)$. Now we prove that α is repetitive.

Consider two chains α_0 and β_0 such that $|\alpha_0| = \min\{k, n-k\}$, $|\beta_0| = \max\{k, n-k\}$ and either $\alpha = \alpha_0\beta_0$ (when $|\alpha_0| = k$) or $\alpha = \beta_0\alpha_0$ (when $|\alpha_0| = n-k$). In both cases, by using the fact that $\alpha = \sigma^k(\alpha)$, we have that $\alpha_0\beta_0 = \beta_0\alpha_0$.

If $|\alpha_0| = |\beta_0|$ then we are done. Assume that $|\alpha_0| \neq |\beta_0|$. By Lemma 3.3.1, there exist two chains α_1 and β_1 such that

a) $\beta_0 = \alpha_1\beta_1 = \beta_1\alpha_1$
b) $\alpha_0 \in \{\alpha_1, \beta_1\}$
c) $|\alpha_1| \leq |\beta_1|$ and $\max\{|\alpha_1|, |\beta_1|\} < \max\{|\alpha_0|, |\beta_0|\}$.

Since (a) and (c) hold, we can use again Lemma 3.3.1. By repeating this procedure we obtain a sequence of pairs of chains α_i, β_i such that:

a) $\beta_i = \alpha_{i+1}\beta_{i+1} = \beta_{i+1}\alpha_{i+1}$
b) $\alpha_i \in \{\alpha_{i+1}, \beta_{i+1}\}$
c) $|\alpha_{i+1}| \leq |\beta_{i+1}|$ and $\max\{|\alpha_{i+1}|, |\beta_{i+1}|\} < \max\{|\alpha_i|, |\beta_i|\}$

for each $0 \leq i < l$, where l is a natural such that $|\alpha_l| = |\beta_l|$. Since $\alpha_l\beta_l = \beta_l\alpha_l$, we have that $\beta_l = \alpha_l$ and hence $\beta_{l-1} = \alpha_l\alpha_l$. From (b) we get that $\alpha_{l-1} = \alpha_l$. Therefore, $\beta_{l-2} = \alpha_{l-1}\beta_{l-1} = \alpha_l^3$. Proceeding in this way, we can use properties (a–c) by backwards induction and we obtain $\alpha = \alpha_l^k$ for some $k > 1$. ∎

We say that a chain α of length n is *minimal* if and only if $\alpha \leq \sigma^k(\alpha)$ for $k = 1, 2, \ldots, n-1$. And we will say that α is *strongly minimal* if $\alpha < \sigma^k(\alpha)$ for $k = 1, 2, \ldots, n-1$. The following result is a direct consequence of Lemma 3.3.2.

Lemma 3.3.3. *A chain α is strongly minimal if and only if it is minimal and simple.*

Finally, a chain $\alpha = (I_0, I_1, \ldots, I_{n-1})$ will be called *minimal of first order* if and only if $I_0 \leq I_i$ for $i = 0, 1, \ldots, n-1$. Observe that if a chain is not minimal of first order then it is not minimal.

The following function tests whether a chain `loop` of length `n`, assumed to be minimal of first order, is in addition strongly minimal. Thus, by Lemma 3.3.3, this function is a test of both minimality and simplicity.

```
int is_strongly_minimal(unsigned int length,unsigned int *loop) {
 unsigned int i,j,v;

 for(i=2;i<=length;i++) {
```

```
  if(loop[i]>loop[1]) continue;
  for(j=1;j<=length-i;j++) {
   if(loop[i+j]<loop[j+1]) return 0;
   if(loop[i+j]>loop[j+1]) goto NEWI;
  }
  v=length-i+1;
  for(j=1;j<i;j++) {
   if(loop[j]<loop[v+j]) return 0;
   if(loop[j]>loop[v+j]) goto NEWI;
  }
  return 0;
  NEWI:;
 }
 return 1;
} /* is_strongly_minimal */
```

To each Markov loop we can uniquely associate a chain as follows. Observe that there is a natural one-to-one correspondence τ between the set $\mathcal{K}' = \{L_1, L_2, \dots, L_m\}$ of relevant Q-basic intervals and the set $\{1, 2, \dots, m\}$ (we simply set $\tau(L_i) = i$ for $1 \leq i \leq m$). From now on, given a Markov loop $I_0 \to I_1 \to \dots \to I_{n-1} \to I_0$ we will identify it with the chain $(\tau(I_0), \tau(I_1), \dots, \tau(I_{n-1}))$. Therefore, in the computer implementation of the algorithms each Markov loop will be represented as a vector of unsigned integers. Note that a Markov loop is repetitive (respectively simple) if and only if the corresponding chain is repetitive (respectively simple). Note also that a Markov loop has length n if and only if the corresponding chain has length n.

We want to generate all simple Markov loops of a given `length` which can be extracted from the data contained in a reduced Markov `matrix`. Now we explain the general strategy we use to do it. All these loops will be successively generated in the natural lexicographical order of chains. For each simple loop α, we will generate a representative $\alpha' = \sigma^i(\alpha)$. We will choose i in such a way that α' is strongly minimal. Since it is very easy to test minimality of first order, for a given candidate $\sigma^i(\alpha)$ we first test whether it is minimal of first order. In the affirmative, strong minimality is then tested by using the previously described function **is_strongly_minimal**.

First we define a "first object" which in fact is not a loop, by setting all the components to zero:

```
void first_object(unsigned int length,unsigned int *loop)
{
 unsigned int i;
```

```
 for(i=1;i<=length;i++) loop[i]=0;
} /* first_object */
```

Next, an external program calls iteratively the function **next_loop**, which produces the following (in the natural lexicographic order of chains) strongly minimal loop from a given one. If such a loop does not exist then the function returns 0. Observe that, incidentally, when we call **next_loop** and the parameter `loop` is the "first object", this function produces the lowest (in the lexicographic order) strongly minimal loop.

```
int next_loop(unsigned int length,unsigned int *loop,
              unsigned int **matrix,unsigned int size)
{
 unsigned int v,s,p;

 s=belongs(loop[length],matrix[loop[length-1]]);
 for(;;) {
  while(matrix[loop[length-1]][0] &&
  s<matrix[loop[length-1]][0]) {
   v=matrix[loop[length-1]][++s];
   if(belongs(loop[1],matrix[v])) {
    loop[length]=v;
    if(is_strongly_minimal(length,loop)) return 1;
   }
  } /* end of "while" */
  p=length-2;

  BACKWARD:;

  while((s=belongs(loop[p+1],matrix[loop[p]]))==
  matrix[loop[p]][0]) {
   if(p==1) {
    if(loop[1]==size) return 0;
    loop[1]++; goto FORWARD;
   }
   p--;
  }
  loop[p+1]=matrix[loop[p]][++s]; p++;

  FORWARD:;

  while(p<length-1) {
```

```
   v=matrix[loop[p]][matrix[loop[p]][0]];
   if(v<loop[1]) {
    if(p>1) {p--; goto BACKWARD;}
    if(loop[1]==size) return 0;
    loop[1]++; continue;
   }
   s=1;
   while((loop[p+1]=matrix[loop[p]][s])<loop[1]) s++;
   p++;
  } /* end of "while (p<length-1)" */

  s=0;
 } /* end of "for" */
} /* next_loop */
```

3.4 Calculus of the Markov transition matrix

This section is devoted to describe the function **Markov**, which calculates the Markov transition matrix and the reduced Markov matrix (the definition of both matrices can be found at page 161) of a Markov model $(T, A \cup V(T), g)$ such that (T, A, f) is a canonical model and $g|_A = f|_A$. Several variables are assumed to be externally memory-allocated:

```
/* INHERITED FROM THE PROGRAM "TREES": */
extern struct arbre MZ;
extern unsigned int *im;

/* NEW VARIABLES: */
extern int **markov,*Vertex,*Edge;
extern unsigned int nn,**markov2,*inv;
```

Note that `MZ` and `*im` are variables inherited from the program "TREES" and they codify the canonical model (T, A, f) according to the conventions of Section 3.2. The g-images of all points of $A \cup V(T)$ are stored on the vector `*im`. Since $g|_A = f|_A$, the positions `im[i]` for $1 \leq i \leq |A|$ have been calculated by the program "TREES". The rest of the vector may have been filled by another external program. The Markov transition matrix and the reduced Markov matrix will be stored, respectively, on `**markov` and `**markov2`.

Note that `**markov` is an array of integers rather than unsigned integers. The sign of the elements of `**markov` allows us to codify the behavior of the

$(A \cup V(T))$-monotone map g according to the following conventions (for the rest of this section, *edge* will stand for "$(A \cup V(T))$-basic interval":

1. We assume an (arbitrary) orientation for every edge n. Since the program "TREES" have stored the edges as ordered pairs of integers {`n.e1`,`n.e2`}, we take an orientation $<$ such that `n.e1<n.e2`.
2. Let n and m be edges such that n g-covers m. Let p and q be points of n such that $q < p$ (with respect to the orientation of n) and $g(p)$ and $g(q)$ belong to m. Then we say that g is *increasing* from n to m if $g(q) < g(p)$ (with respect the orientation of m), and *decreasing* otherwise.
3. Let `n`= {`n.e1`,`n.e2`} be an edge. Since g is $(A \cup V(T))$-monotone, it makes sense to sort consecutively the set of edges g-covered by n: the first edge being the one containing the point $g(n.e1)$, the last one containing the point $g(n.e2)$.

In order to codify (2), we multiply by -1 the element `markov[n][m]` if and only if g is decreasing from `n` to `markov[n][m]`. Moreover, we sort the elements of `markov[n]` with respect to the order defined by (3).

Note that (2) is equivalent to associate a sign to each arrow of the Markov graph of g. This will allow us to associate a sign to each generated Markov loop.

The function constructs the reduced matrix `**markov2` by discarding the edges which cannot belong to any Markov loop because either they are not g-covered by any edge or they do not g-cover any edge, and then proceeding iteratively by discarding edges which either g-cover or are g-covered by previously discarded edges. Recall that, by definition, the edges of the reduced matrix `**markov2` are unsigned and each row is sorted in increasing order (see page 161). The number of rows of the matrix `**markov2`, which equals the amount of relevant edges, is stored in the variable `nn`. This reduced matrix is the one which is used by the functions of Section 3.3 in order to generate simple loops.

The names of the edges in the reduced matrix are not the original ones. The re-labeling is codified using the vector `*inv`: `inv[m]`=`n` if and only if the n-th original edge is the m-th one after the reduction process.

```
void Markov(void) {

 unsigned int i,j,k;

 for(i=1;i<=MZ.nedges;i++) { markov[i][0]=0; markov2[i][0]=0;}
 for(i=1;i<=MZ.nedges;i++) { unsigned int vector[3],point,edg;
  vector[0]=2;
```

```
 vector[1]=im[MZ.edges[i].e1]; vector[2]=im[MZ.edges[i].e2];
 if(vector[1]==vector[2]) continue;
 convex_hull(MZ,Vertex,Edge,vector);
 point=vector[1];
 edg=0;
 do {
  if(++edg>MZ.nedges) edg=1;
  if(!Edge[edg]) continue;
  if(point==MZ.edges[edg].e1) {
   markov[i][++markov[i][0]]=edg;
   Edge[edg]=0; point=MZ.edges[edg].e2;
   continue;
  }
  if(point==MZ.edges[edg].e2) {
   markov[i][++markov[i][0]]=-edg;
   Edge[edg]=0; point=MZ.edges[edg].e1;
   continue;
  }
 } while(punt!=vector[2]);
}  /* Markov transition matrix (**markov) computed. */

/* Now we compute the reduced matrix **markov2 : */

/* Deleting irrelevant edges (we temporary use
   *Edge as edge-renumbering vector: */

for(i=1;i<=MZ.nedges;i++) Edge[i]=1;
do { k=0;
 for(i=1;i<=MZ.nedges;i++) {
  if(!Edge[i]) continue;
  for(j=1;j<=markov[i][0];j++)
   if(Edge[abs(markov[i][j])]) goto MORE;
  Edge[i]=0; k=1; goto NEWI; /* i-th edge doesn't
                                cover anything */
  MORE:;
  for(j=1;j<=MZ.nedges;j++) {
   if(!Edge[j]) continue;
   if(pos(i,markov[j])) goto NEWI;
  }
  Edge[i]=0; k=1; /* i-th edge is not covered
                     by anything */
  NEWI:;
 }
} while(k);
```

```
 /* change of names (direct and inverse), and
    nn = amount of relevant edges: */

 nn=0;
 for(i=1;i<=MZ.nedges;i++) {
  if(!Edge[i]) continue;
  Edge[i]=(++nn); inv[nn]=i;
 }

 /* translating into **markov2: */

 for(i=1;i<=MZ.nedges;i++) {
  if(!Edge[i]) continue;
  for(j=1;j<=markov[i][0];j++) { int l;
   if(!(l=Edge[abs(markov[i][j])])) continue;
   markov2[Edge[i]][++markov2[Edge[i]][0]]=l;
  }
 }

 /* sorting each row: */

 for(i=1;i<=nn;i++) if(markov2[i][0]>1)
  Sort1(markov2[i][0],markov2[i]);
} /* Markov */
```

3.5 Tests of period-forcing

As we explained at the beginning of the chapter, we are mainly interested in testing the following statement:

Conjecture 3.5.1. *Let $\mathcal{P}$ be a pattern. Let (T, A, f) be a canonical model of $\mathcal{P}$. If g is a tree map exhibiting $\mathcal{P}$, then* $\mathrm{Per}(f) \subset \mathrm{Per}(g)$.

A simple example shows that Conjecture 3.5.1 is false. Consider a pattern $([T, A], [\theta])$ with $A = \{x_1, x_2, x_3, x_4, y\}$, $\theta(x_i) = x_{i+1 \bmod 4}$ for $1 \le i \le 4$ and $\theta(y) = y$, such that (T, A) consists of two discrete components $\{x_1, x_3, y\}$ and $\{x_2, x_4, y\}$. On the left side of Figure 3.2 the linear canonical model (T, A, f) of this pattern is shown. On the right side, a linear Markov model (T, A, g) exhibiting the same pattern with $g(v) = g(v') = y$ is shown. Observe that $\mathrm{Per}(f) = \{1, 2, 4\}$, while $\mathrm{Per}(g) = \{1, 4\}$. Hence these maps provide a counterexample to Conjecture 3.5.1. However, it is not difficult to see that

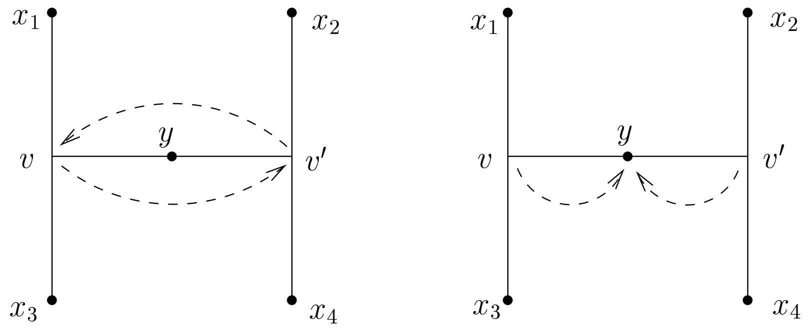

Figure 3.2: Two models which exhibit a pattern $([T, A], [\theta])$ with $A = \{x_1, x_2, x_3, x_4, y\}$, $\theta(x_i) = x_{i+1 \bmod 4}$ for $1 \leq i \leq 4$ and $\theta(y) = y$. On the left figure, the canonical model (T, A, f), for which $\{v, v'\}$ is a 2-periodic orbit. On the right figure, a Markov model (T, A, g) with $g(v) = g(v') = y$.

all 2-periodic points of f are not A-significant. In fact, there is only one f-monotone equivalence class, and each periodic point of f belongs to this class.

This example shows us that we have to restrict our attention to significant periodic points, and suggests the following

Conjecture 3.5.2. *Let $\mathcal{P}$ be a pattern. Let (T, A, f) be a canonical model of $\mathcal{P}$. If there is an A-significant n-periodic point of f then $n \in \mathrm{Per}(g)$ holds for each tree map g exhibiting $\mathcal{P}$.*

In order to simplify the computer search we can consider also several slightly weaker versions:

Conjecture 3.5.3. *Let $\mathcal{P}$ be a pattern. Let (T, A, f) be a canonical model of $\mathcal{P}$. If there is an A-significant n-periodic point of f then $n \in \mathrm{Per}(g)$ holds for each linear Markov model $(S, B \cup V(S), g)$ exhibiting $\mathcal{P}$ over B.*

Conjecture 3.5.4. *Let $\mathcal{P}$ be a pattern. Let (T, A, f) be a canonical model of $\mathcal{P}$. If there is an A-significant n-periodic point of f then $n \in \mathrm{Per}(g)$ holds for each linear Markov model $(T, A \cup V(T), g)$ exhibiting $\mathcal{P}$ over A.*

If "$\Rightarrow$" stands for "implies" then obviously Conjecture 3.5.2 $\Rightarrow$ Conjecture 3.5.3 $\Rightarrow$ Conjecture 3.5.4. On the other hand, it is not difficult to see that Conjecture 3.5.3 $\Rightarrow$ Conjecture 3.5.2. For us, it is an open problem whether Conjecture 3.5.4 implies Conjecture 3.5.3.

Next we discuss some possible strategies of testing those conjectures from a numerical point of view. Let $\mathcal{P}$ be a pattern. Let (T, A, f) be a canonical model of $\mathcal{P}$. Let $(T, A \cup V(T), g)$ be a Markov model exhibiting $\mathcal{P}$ over A.

In particular, $g|_A = f|_A$ holds. Consequently, the Markov transition matrix (and thus the set of periods) of g depends only on the images of g over the (finite) set $V(T) \setminus A$. On the other hand, since $(T, A \cup V(T), g)$ is a Markov model it follows that $g(V(T)) \subset A \cup V(T)$. Therefore, if we set $k = |V(T) \setminus A|$ and $m = |A \cup V(T)|$, for a fixed pattern $\mathcal{P}$ there are k^m-many essentially different Markov models to be tested in order to find a potential counterexample to Conjecture 3.5.4. In other words, Conjecture 3.5.4 is appropriate for computer exploration. Furthermore, a counterexample to Conjecture 3.5.4 is also a counterexample to Conjectures 3.5.3 and 3.5.2.

On the other hand, consider a Markov model $(S, B \cup V(S), g)$ exhibiting $\mathcal{P}$ over B. In particular, $|A| = |B|$ and the sets of discrete components of (T, A) and (S, B) are exactly the same. Consider a discrete component K. Since there are only finitely many trees with $|\operatorname{En}(\langle K \rangle)|$ endpoints (up to homeomorphisms), for a fixed pattern $\mathcal{P}$ there are also finitely many essentially different Markov models to be tested in order to find a potential counterexample to Conjecture 3.5.3. Hence Conjecture 3.5.3 is also appropriate for computer exploration.

In a first step, we have chosen to test the validity of Conjecture 3.5.4. To do it, we have constructed a program which uses almost all the functions described in this chapter. In particular, we need a function to calculate the set of periods of a given linear Markov model. This function will be based on the following result, whose proof we only outline since the involved ideas are folk knowledge for interval maps and can be easily extended to tree maps.

Proposition 3.5.5. *Let (T, Q, g) be a linear Markov model. Let x be a periodic point of g. Then, either $\operatorname{Orb}(x) \subset Q$ or $\operatorname{Orb}(x) \cap Q = \emptyset$ and in this case there is a unique simple loop α in the Markov g-graph of Q satisfying that x and α are associated and $|\alpha| = |\operatorname{Orb}(x)|$.*

Proof. Assume that $\operatorname{Orb}(x) \cap Q = \emptyset$. Set $n = |\operatorname{Orb}(x)|$. By Lemma 1.3.2 there exists a unique loop $I_0 \to I_1 \to \dots \to I_{n-1} \to I_0$, which we denote by α, such that x and α are associated. We have to see that α is simple. By Lemma 1.3.3 and Remark 1.3.4, there exists a closed interval $K \subset I_0$ containing x such that g^n is monotone on K and $g^n(K) = I_0$. If α is not a *thin* loop (α is said to be *thin* if I_{i+1} is the only Q-basic interval g-covered by I_i for $0 \le i < n-1$ and I_0 is the only Q-basic interval g-covered by I_{n-1}), the fact that α is simple follows analogously as in the proof of Stage 4, Case 2 of Theorem 2.5.1. On the other hand, when α is thin the fact that α is simple easily follows from Corollary 2.2.7 and Proposition 2.2.8 of [34]. ■

Let $(T, A \cup V(T), g)$ be a linear Markov model. We have used Proposition 3.5.5 to construct a function `isperiod` which admits an integer n and a map g and tests whether $n \in \operatorname{Per}(g)$. The function returns an integer according to the following convention:

1. `isperiod(n,g)`=0 if $n \notin \operatorname{Per}(g)$.
2. `isperiod(n,g)`=1 if there is some simple positive loop of length n in the Markov g-graph of $A \cup V(T)$.
3. `isperiod(n,g)`=2 if all simple loops of length n in the Markov g-graph of $A \cup V(T)$ are negative.
4. `isperiod(n,g)`=3 if there are no simple loops of length n in the Markov g-graph of $A \cup V(T)$ but there is an n-periodic orbit of g contained in $A \cup V(T)$.

Besides all the loop-managing routines of Sections 3.3 and 3.4, `isperiod` needs another function, which we have called `OrbitOfVertices`, in order to test whether $A \cup V(T)$ contains an n-periodic orbit of g. Next we list its source code. `OrbitOfVertices` assumes all the conventions and global variables of this chapter. In particular, the g-images of the points of $V(T) \setminus A$ are stored on the vector `*im`, and `endpat` equals $|A|+1$. The function returns 1 if $A \cup V(T)$ contains an n-periodic orbit, 0 otherwise.

```
int OrbitOfVertices(unsigned int n) {
 unsigned int i,j,k;

 /* Deleting eventually periodic points:
    (vector pupe externally memory-allocated) */

 for(i=endpat;i<=MZ.nedges+1;i++) if(im[i]<endpat) pupe[i]=0;
  else pupe[i]=1;
 do {
  k=0;
  for(i=endpat;i<=MZ.nedges+1;i++) {
   /* Can we delete the i-th point ? */
   if(!pupe[i]) continue;  /* point already deleted */
   for(j=endpat;j<=MZ.nedges+1;j++)
    if(pupe[j] && im[j]==i) goto DONOTDELETE;
   k=1; pupe[i]=0;
   DONOTDELETE:;
  }    /* k=0 if no point has been deleted */
 } while(k);

 /* Each non-deleted point is periodic. Now we go through
    its images and compute the orbit size: */

 for(i=endpat;i<=MZ.nedges+1;i++) {
```

```
  if(!pupe[i]) continue;
  j=0; k=i; pupe[i]=0;
  do pupe[(k=im[k])]=0; while(++j<n && k!=i);
  if(j<n) continue;   /* period < n */
  if(k!=i) continue;   /* period > n */
  return 1;
 }
 return 0;
}
```

Finally, next we state a result which has been used to optimize the period-forcing exploration. The proof of (a) is not difficult, and (b) and (c) follow easily from the ideas of the proof of Proposition 2.7.6.

Proposition 3.5.6. *Let $\mathcal{P}$ be a pattern and let (T, A, f) be a canonical model of $\mathcal{P}$ having an A-significant n-periodic point x. Let β be a simple loop $\pi_0 \to \pi_1 \to \ldots \to \pi_0$ in the $\mathcal{P}$-graph path such that x and β are associated, and let α be a simple loop in the Markov $(A \cup V(T))$-graph of f such that x and α are associated. Then the following statements hold:*

(a) α and β have the same sign.

(b) If α (or β) is positive then $n \in \mathrm{Per}(g)$ for each tree map g which exhibits $\mathcal{P}$.

(c) If $\mathrm{Int}(\langle \pi_0 \rangle) \cap V(T) = \emptyset$ then $n \in \mathrm{Per}(g)$ for each tree map g which exhibits $\mathcal{P}$.

A period-forcing explorer program has been designed with the aim of finding counterexamples to Conjecture 3.5.4. Next we describe its underlying algorithm (we do not list the source code since it is simply composed of calls to some known functions and, in consequence, it lacks interest).

Period-forcing explorer algorithm:

1. Consider a fixed pattern $\mathcal{P}$.
2. Compute a canonical model (T, A, f) of $\mathcal{P}$.
3. Compute the Markov f-graph of $A \cup V(T)$.
4. Consider a fixed integer $k > 1$ ("period-exploring depth").
5. For each $2 \le n \le k$, now we *test* the period n:
   (a) If `isperiod(n,f)`$=0$ then $n \notin \mathrm{Per}(f)$ and we stop the test for n.
   (b) If `isperiod(n,f)`$=1$ then $n \in \mathrm{Per}(f)$ and there is some simple positive loop of length n in the Markov f-graph of $A \cup V(T)$. If the n-periodic point associated to this loop is A-significant, then

by Proposition 3.5.6 (b) we have that $n \in \mathrm{Per}(g)$ for each tree map g exhibiting $\mathcal{P}$. This is not necessarily true when the point is not A-significant. In any case, we stop the test for n (below we discuss which periods we possibly overlook by proceeding in this way).

(c) If `isperiod(n,f)`$=2$ then $n \in \mathrm{Per}(f)$. If there exists some simple loop $\pi \to \ldots \to \pi$ of length n in the $\mathcal{P}$-path graph such that $\mathrm{Int}(\langle\pi\rangle) \cap V(T) = \emptyset$, then we stop the test for n (Proposition 3.5.6 (c)). Otherwise, for each Markov model $(T, A \cup V(T), g)$ exhibiting $\mathcal{P}$ over A we test whether `isperiod(n,g)`$\neq 0$. If `isperiod(n,g)`$= 0$ for some g then we have found a counterexample to Conjecture 3.5.4.

(d) If `isperiod(n,f)`$=3$ then $n \in \mathrm{Per}(f)$ but the n-periodic orbits of f are contained in $A \cup V(T)$. In this case we stop the test for n.

Let $\mathcal{P}$ be a pattern and let (T, A, f) be a canonical model of $\mathcal{P}$. We say that a positive integer n is *special* for $\mathcal{P}$ if $n \in \mathrm{Per}(f)$ and each n-periodic point x of f satisfies the following conditions:

1. there is a simple positive Markov loop associated to x
2. x is not A-significant.

We say that a pattern $\mathcal{P}$ is *special* if there exists some $n \in \mathbb{N}$ such that n is special for $\mathcal{P}$.

If a period n is special for a pattern $\mathcal{P}$, our algorithm overlooks the period n (the test for n stops because `isperiod(n,f)`$= 1$), while we cannot assure that $n \in \mathrm{Per}(g)$ for each Markov map g exhibiting $\mathcal{P}$. In order to avoid this circumstance, we can execute the algorithm only over non-special patterns. Alternatively, it is not difficult to manually check the existence of special periods for a given pattern, and test these periods apart.

The program has been linked using the standard `gcc` compiler on a UNIX Workstation. It has been running for several months on an Origin-2000 machine of Silicon Graphics with 8 processors. A wide variety of patterns have been tested. The results *confirm* Conjecture 3.5.4 in the affirmative: no counterexample has been found. Next we display a list of some tested patterns. In any case, (T, A, f) is a periodic model, with $A = \{x_i\}_{i=1}^{|A|}$ and $f(x_i) = x_{i+1 \bmod |A|}$ for $1 \le i \le |A|$. For each pattern we show the set of discrete components and the periods n for which `isperiod(n,f)`$=2$ (tested periods):

- $\{x_1, x_3, x_{10}, x_{11}\}$, $\{x_1, x_2, x_4, x_7, x_8\}$, $\{x_1, x_5, x_6, x_9\}$, period 4.
- $\{x_1, x_4, x_6, x_{10}\}$, $\{x_1, x_2, x_3, x_7\}$, $\{x_1, x_5, x_8, x_9\}$, period 6.
- $\{x_1, x_5, x_6, x_7\}$, $\{x_1, x_2, x_3, x_4\}$, period 8.

- $\{x_1, x_2, x_4, x_5, x_7, x_8\}$, $\{x_5, x_6\}$, $\{x_3, x_6, x_9\}$, period 4.
- $\{x_1, x_2, x_3, x_4\}$, $\{x_1, x_5, x_8\}$, $\{x_3, x_6, x_9\}$, $\{x_4, x_7, x_{10}\}$, $\{x_1, x_{11}, x_{12}\}$, period 8.
- $\{x_1, x_3, x_6\}$, $\{x_1, x_2, x_4\}$, $\{x_2, x_5, x_7\}$, period 4.
- $\{x_1, x_3, x_6, x_9\}$, $\{x_1, x_2, x_4, x_7\}$, $\{x_2, x_5, x_8, x_{10}\}$, period 4.
- $\{x_{20}, x_{21}, x_{22}\}$, $\{x_2, x_{11}, x_{20}\}$, $\{x_2, x_5, x_8\}$, $\{x_5, x_{14}, x_{23}\}$, $\{x_8, x_{17}, x_{26}\}$, $\{x_1, x_{10}, x_{22}\}$, $\{x_1, x_4, x_7\}$, $\{x_4, x_{13}, x_{25}\}$, $\{x_7, x_{16}, x_{19}\}$, $\{x_3, x_{12}, x_{21}\}$, $\{x_3, x_6, x_9\}$, $\{x_6, x_{15}, x_{24}\}$, $\{x_9, x_{18}, x_{27}\}$, period 24.
- $\{x_1, x_2, x_3\}$, $\{x_1, x_4, x_8\}$, $\{x_2, x_5, x_9\}$, $\{x_3, x_6, x_7\}$, periods 6 and 8.
- $\{x_1, x_2, x_3\}$, $\{x_1, x_4, x_5\}$, $\{x_2, x_6, x_8\}$, $\{x_3, x_7, x_9\}$, period 8.
- $\{x_1, x_2, x_3, x_7\}$, $\{x_1, x_4, x_5, x_6\}$, period 6.

Chapter 4

A note on the periodic orbits and topological entropy of graph maps

4.1 Introduction

The notion of topological entropy appeared early in the sixties (see [1]). It is defined for continuous maps on compact metric spaces and is a quantitative measure of the dynamical complexity of the map. It is an important topological invariant.

There are some properties of the dynamical behavior of the maps which are controlled by the topological entropy. For instance, it measures the exponential growth rate, when n tends to infinity, of the number of different orbits of length n if we use certain precision to distinguish two orbits (see [25]). For a piecewise monotone map f of the interval, it measures also the exponential rate of increase with n of the number of maximal intervals of monotonicity of f^n (see [39]).

We are interested in relating periodic orbits and topological entropy. For continuous maps on the interval, to every periodic orbit P of f we can associate a number $h(P)$ which is the topological entropy of the "connect-the-dots" map corresponding to P or the "linearization" of P. In fact, this entropy corresponds to the infimum of the entropies of all maps exhibiting orbits with the same combinatorics as P (see Corollary 4.4.7 of [8]).

In the interval case it is possible to show that the entropy of any map f is the supremum of the values $h(P)$ corresponding to all the periodic orbits P of f. Furthermore, for each n, we can take this supremum only over the orbits of period $k > n$. This result was stated by Takahashi [43] and proved with the assumption that f is piecewise monotone. In the general case it also was proved in an independent way by Block and Coven [20] and Misiurewicz and Nitecki [38].

Since the topological entropy is usually considered as a measure of the degree of chaos, a natural problem is developing algorithms for calculating it (see [29], [40] or [17]). These algorithms are based on different properties of the entropy and some of them take into account the existence of periodic orbits (see, for instance, [20]) and, in particular, properties like the Takahashi's result.

In this chapter we show that an analogous relation between periodic behavior and topological entropy is satisfied for continuous maps on graphs (see [4]). Our work has been motivated by a question posed by S. Kolyada and N. Snoha to Ll. Alsedà. To this end we introduce some basic notation and state in detail the main result of the chapter. We recall that the basic definitions on graphs has been introduced in subsection 2.8.1.

Let G be a graph. Given a point $x \in V(G)$, the number of edges containing x (with the edges homeomorphic to a circle counted twice) will be called the *valence of* x. Since any graph can be embedded in $\mathbb{R}^3$, in what follows we will consider each graph endowed with the topology induced by the topology of $\mathbb{R}^3$.

Now we extend the notion of *interval* introduced in subsection 2.8.1. We shall call a set $J \subset G$ an *interval* if there is a homeomorphism $\phi : I \longrightarrow J$, where I is $[0,1]$, $(0,1]$, $[0,1)$ or $(0,1)$, and there are no vertices in J except perhaps $\phi(0)$ and $\phi(1)$. The set $\phi((0,1))$ will be called the *interior of* J and will be denoted $\mathrm{Int}(J)$. If $I = [0,1]$, the interval J will be called *closed* and if $I = (0,1)$, the interval J will be called *open.* Notice that it may happen that $\mathrm{Int}(J) \neq J$ for an interval J being an open set in the topology of G. For example, let G be a graph with two vertices and one edge. Then G is an interval and an open set as a topological space but $\mathrm{Int}(G)$ does not contain the vertices. As usual, a *subinterval* of an interval J will be an interval contained in J.

Now we define an equivalence relation among graph models as follows: we say that (G, A, f) and (G', A', f') are *equivalent* if there exists a homeomorphism $\phi : G \longrightarrow G'$ with $\phi(A) = A'$ such that f and $\phi^{-1} \circ f' \circ \phi$ are homotopic relative to A. Notice that then $\phi^{-1} \circ f' \circ \phi|_A = f|_A$. Each equivalence class of this relation, denoted by $[G, A, f]$, will be called an *action.* Note that if two graph models are equivalent then they have the same graph pattern.

We define the *entropy of* $[G, A, f]$, denoted $h([G, A, f])$, as

$$h([G, A, f]) = \inf\{h(f') : (G', A', f') \in [G, A, f]\}.$$

Given an action $[G, A, f]$, from [11] it follows that there exists a representative

(G, A, g) of $[G, A, f]$ that gives the entropy of the action, i.e., such that $h(g) = h([G, A, f])$. We shall use this fact to obtain lower bounds of the topological entropy of a graph map. The main result of this chapter is the following.

Theorem E. *Let $f : G \longrightarrow G$ be a graph map. For each nonnegative integer m we have*

$$h(f) = \sup\{h([G, P, f]) \ : \ P \text{ periodic orbit of } f \text{ and } |P| > m\}.$$

Our proof of this result is based in the main ideas used in the proof of the analogous result for interval maps (see, for instance, theorem 4.4.10 of [8]) and in some properties of graph maps pointed out in [36].

4.2 Proof of Theorem E

We shall need a simple property which is well known for interval and circle maps (see, for instance, [8]).

Lemma 4.2.1. *Let $f : G \longrightarrow G$ be a graph map. Let $\{J_i \subset G : i = 1, 2, \dots, n\}$ be a family of closed intervals such that $f(J_i) \supset J_{i+1}$ for $i = 1, 2, \dots, n-1$, and $f(J_n) \supset J_1$. Then there exists a point $x = f^n(x)$ such that $f^i(x) \in J_{i+1}$ for all $0 \leq i \leq n-1$.*

Proof. Since $f(J_n) \supset J_1$ and there are no vertices in the interior of an interval, there exists a closed subinterval $K_n \subset J_n$ with $f(K_n) = J_1$. Analogously, since $f(J_{n-1}) \supset J_n$, there is a closed subinterval $K_{n-1} \subset J_{n-1}$ with $f(K_{n-1}) \subset J_n$ and $f^2(K_{n-1}) = J_1$.

Inductively, there is a closed subinterval $K_1 \subset J_1$ with $f^i(K_1) \subset J_{i+1}$, for $i = 1, 2, \dots, n-1$, and $f^n(K_1) = J_1$. Then, since f^n is a continuous map and there are no vertices in $\text{Int}(J_1)$, the intermediate value theorem ensures the existence of a point $x \in K_1$ such that $f^n(x) = x$. By the election of K_1 it follows that $f^i(x) \in J_{i+1}$ for all $0 \leq i \leq n-1$ and the lemma follows. ■

As for interval and circle maps, an important notion for obtaining minimal models of an action is the notion of local monotonicity. We recall that this notion has been introduced in subsection 2.8.1.

Let (G, A, f) be a representative of an action. We say that (G, A, f) is *monotone* if f restricted to any interval I without points of $A \cup V(G)$ in its interior is monotone. If in addition $f(V(G)) \subset A \cup V(G)$ then (G, A, f) is called *simplicial*. Given an action, as we noticed above, there is a representative such that its entropy coincides with the topological entropy of the action. Moreover, in [11] it is shown that this representative can be taken simplicial. We shall use this fact in order to prove that the existence of a horseshoe gives a lower bound of the topological entropy of the map. To this end we introduce the notion of horseshoe.

Let $s \geq 2$. An *s-horseshoe for f* is a closed interval $I \subset G$ and closed subintervals $J_1, J_2, \ldots, J_s$ of I with pairwise disjoint interiors, such that $f(J_i) = I$ for $j = 1, 2, \ldots, s$. An s-horseshoe is *strong* if in addition the intervals $J_1, J_2, \ldots, J_s$ are contained in $\mathrm{Int}(I)$ and are pairwise disjoint.

Proposition 4.2.2. *Let $f : G \longrightarrow G$ be a graph map. Assume that f^k has a strong s-horseshoe for some $k \geq 1$ and $s \geq 3$. Then there is a periodic orbit P of f with period $|P| \geq 2(s-2)$ such that $h([G, P, f]) \geq \frac{1}{k}\log(s-2)$.*

Proof. From the definition of a strong horseshoe there exist a closed interval $I \subset G$ and pairwise disjoint subintervals $J_1, J_2, \ldots, J_s$ contained in the interior of I, such that $f^k(J_i) = I$ for $i = 1, 2, \ldots, s$. Thus we have $f^k(J_1) \supset J_i$, $f^k(J_i) \supset J_1$, $f^k(J_s) \supset J_i$ and $f^k(J_i) \supset J_s$ for $i = 2, 3, \ldots, s-1$.

Then we consider the sequence of intervals $\{I_i : i = 1, 2, \ldots, 4(s-2)\}$ defined by

$$I_i = \begin{cases} J_{j+1} & \text{if } i = 2j-1 \quad \text{for } j = 1, 2, \ldots, s-2 \\ J_1 & \text{if } i = 2j \quad \text{for } j = 1, 2, \ldots, s-2 \\ J_{j+1-(s-2)} & \text{if } i = 2j-1 \quad \text{for } j = s-1, s, \ldots, 2(s-2) \\ J_s & \text{if } i = 2j \quad \text{for } j = s-1, s, \ldots, 2(s-2). \end{cases}$$

Actually, this sequence is

$$J_2, J_1, J_3, J_1, \ldots, J_1, J_{s-1}, J_1, J_2, J_s, J_3, J_s, \ldots, J_s, J_{s-1}, J_s\,.$$

It satisfies the hypothesis of Lemma 4.2.1 for f^k. Thus, there exists a periodic orbit $Q = \{x, f^k(x), f^{2k}(x), \ldots\}$ of f^k with $f^{ik}(x) \in I_{i+1}$. Let P be the periodic orbit of f containing Q. Obviously $|P| \geq |Q|$. Furthermore $|Q| \geq 2(s-2)$. Indeed, if $|Q| < 2(s-2)$ then there exists $i \in \{1, 3, 4, \ldots, s-1\}$ with $x \in J_i \cap J_2$, which gives a contradiction.

Given $(\overline{G}, \overline{P}, \overline{f}) \in [G, P, f]$, if ϕ is the homeomorphism given by the equivalence between $(\overline{G}, \overline{P}, \overline{f})$ and (G, P, f), then the subset $\phi(Q)$ of $\overline{P}$ satisfies $(\overline{G}, \phi(Q), \overline{f}^k) \in [G, Q, f^k]$. Since $\frac{1}{k}h(\overline{f}^k) = h(\overline{f})$ we have

$$\begin{aligned} h([G, P, f]) &= \inf\{h(\overline{f}) : (\overline{G}, \overline{P}, \overline{f}) \in [G, P, f]\} \\ &= \inf\{\frac{1}{k}h(\overline{f}^k) : (\overline{G}, \overline{P}, \overline{f}) \in [G, P, f]\} \\ &\geq \inf\{\frac{1}{k}h(\overline{f}) : (\overline{G}, \overline{Q}, \overline{f}) \in [G, Q, f^k]\} = \frac{1}{k}h([G, Q, f^k]). \end{aligned}$$

Now we prove that $h([G, Q, f^k]) \geq \log(s-2)$ and we are done. From [11] there exists a simplicial representative (G, Q, g) of $[G, Q, f^k]$ such that $f^k|_Q = g|_Q$ and $h([G, Q, f^k]) = h(g)$. Notice that g has a strong $(s-2)$-horseshoe. Indeed, for each $i \in \{2, 3, \ldots, s-1\}$ there is an element from $Q \cap J_i$ mapped to J_1 and an element from $Q \cap J_i$ mapped to J_s. Thus, since g is Q-monotone and $J_i \subset I$, every $g(J_i)$ contains all the intervals $J_2, J_3, \ldots, J_{s-1}$. Thus g also has a strong $(s-2)$-horseshoe as we claimed because $J_1, J_2, \ldots, J_s$ is a strong s-horseshoe of f^k.

From this fact and Lemma 3.4 of [36] it follows that $h(g) \geq \log(s-2)$ and

$$h([G, P, f]) \geq \frac{1}{k} h([G, Q, f^k]) = \frac{1}{k} h(g) \geq \frac{1}{k} \log(s-2).$$

This ends the proof. ∎

Lastly we prove Theorem E.

Proof of Theorem E. From the definition of $h([G, P, f])$ it follows that

$$h(f) \geq \sup\{h([G, P, f]) \ : \ P \text{ periodic orbit of } f \ \text{ and } |P| > m\}.$$

So, we shall prove the other inequality.

If $h(f) = 0$ we are done. So we assume that $h(f) > 0$. From Theorem B of [36], there are sequences of natural numbers $(k_n)_{n=1}^{\infty}$ and $(s_n)_{n=1}^{\infty}$ such that f^{k_n} has an s_n-horseshoe for each $n \geq 1$ and $h(f) = \limsup_{n \to \infty} \frac{1}{k_n} \log s_n$. Furthermore, from Lemma 3.3 of [36], it follows that f^{k_n} has a strong $(s_n - 2)$-horseshoe for each $n \geq 1$.

Now we distinguish two cases.

Case 1. $h(f) = \infty$.

Then the natural numbers s_n take infinitely many different values. So we can choose them in such a way that $(s_n)_{n=1}^{\infty}$ is an increasing sequence.

From Proposition 4.2.2 (with $s = s_n - 2$) it follows that for each $n \in \mathbb{N}$ there exists a periodic orbit P_n of f such that $|P_n| \geq 2(s_n - 4)$ and $h([G, P_n, f]) \geq \frac{1}{k_n} \log(s_n - 4)$. Given any $M > 0$, since $\limsup_{n \to \infty} \frac{1}{k_n} \log s_n = \infty$, we can choose an n such that $\frac{1}{k_n} \log(s_n - 4) > M$ and $2(s_n - 4) > m$. Then we have $|P_n| > m$ and

$$h([G, P_n, f]) \geq \frac{1}{k_n} \log(s_n - 4) > M.$$

Since M is arbitrary, it follows that

$$h(f) = \infty = \sup\{h([G, P, f]) : P \text{ periodic orbit of } f \ \text{ and } |P| > m\},$$

as we claimed.

Case 2. $0 < h(f) < \infty$.

Since $(k_n)_{n=1}^{\infty}$ and $(s_n)_{n=1}^{\infty}$ are sequences of natural numbers and $0 < \limsup_{n\to\infty} \frac{1}{k_n} \log s_n < \infty$, then either both sequences $(k_n)_{n=1}^{\infty}$ and $(s_n)_{n=1}^{\infty}$ take infinitely many different values or both sequences take finitely many values.

If $(k_n)_{n=1}^{\infty}$ and $(s_n)_{n=1}^{\infty}$ take infinitely many different values, we can choose the sequence $(s_n)_{n=1}^{\infty}$ in such a way that it is an increasing sequence. As in Case 1, for each $n \in \mathbb{N}$ there exists a periodic orbit P_n of f such that $|P_n| \geq 2(s_n - 4)$ and $h([G, P_n, f]) \geq \frac{1}{k_n} \log(s_n - 4)$. Now $\limsup_{n\to\infty} \frac{1}{k_n} \log(s_n - 4) = h(f)$. Thus, given any $\epsilon > 0$, there exists an n such that $\frac{1}{k_n} \log(s_n - 4) > h(f) - \epsilon$ and $2(s_n - 4) > m$. Then we have $|P_n| > m$ and

$$h([G, P_n, f]) \geq \frac{1}{k_n} \log(s_n - 4) > h(f) - \epsilon.$$

Since ϵ is arbitrary, it follows that $\sup\{h([G, P, f]) : |P| > m\} \geq h(f)$, as we claimed.

If $(k_n)_{n=1}^{\infty}$ and $(s_n)_{n=1}^{\infty}$ take finitely many different values, then there exist natural numbers k and s such that $h(f) = \frac{1}{k} \log s$ and f^k has an s-horseshoe. Then we can take $k_n = nk$ and $s_n = s^n$ for all $n \geq 1$ and we have $\limsup_{n\to\infty} \frac{1}{k_n} \log s_n = h(f)$. Also, from Lemmas 3.2 and 3.3 of [36], it follows that f^{nk} has an s^n-horseshoe and consequently a strong $(s^n - 2)$-horseshoe, for each $n \geq 1$. The rest of the proof follows as above. ∎

Appendix: Source code of some general purpose functions

A.1 Dynamic memory management

Here we list the source code of some functions which allow us to allocate and release memory space for vectors and matrices of types `unsigned int` and `struct pair`:

```
int reserve_matrix(unsigned int ***mat,unsigned int f,
                   unsigned int c)
{
  unsigned int i;
  int reserve_vector(unsigned int **,unsigned int);
  void free_matrix(unsigned int ***,unsigned int);

  *mat = (unsigned int **) malloc(f*sizeof(*(*mat)));
  if(!(*mat)) return 0;

  if(!c) {for(i=0;i<f;i++){ (*mat)[i] = NULL; } return 1;}

  for(i=0;i<f;i++){
    if(!reserve_vector(&((*mat)[i]),c)) {
      free_matrix(mat,i); return 0;
    }
  }
  return 1;
}

void free_matrix(unsigned int ***mat,unsigned int f)
{ unsigned int i; void free_vector(unsigned int **);
  for(i=0;i<f;i++){free_vector(&((*mat)[i]));} free(*mat);}
```

The function **reserve_matrix** reserves memory for an array `**mat` of `f` rows and `c` columns. If the parameter `c` is given the value 0, then memory is reserved for `f`-many non initialized pointers for which a value `NULL` is given (that is, `f`-many vectors of undetermined size). For each of these vectors, further calls to the function **reserve_vector** allows variable row-size for the matrix `**mat`.

We remark that in all functions performing dynamic memory management the variable for which we want to reserve memory appears with an extra level of reference (we have, for instance, `***mat` instead of `**mat`). Therefore, it is necessary to send as a parameter the address of the variable which we want to allocate, instead of its value. The function modifies this address to be the starting address of the reserved memory space.

The rest of functions of dynamic memory management work analogously as the function **reserve_matrix**:

```
int reserve_vector(unsigned int **v,unsigned int n)
{ *v = (unsigned int *) malloc(n*sizeof(unsigned int));
  if(*v) return 1; else return 0;}


void free_vector(unsigned int **v) {if(*v) free(*v);}
```

The next functions are just the “huge” version of the previous ones.

```
int reserve_matrix_H(unsigned int huge *(huge *(*mat)),
                        unsigned int f,unsigned int c)
{
  unsigned int i;
  int reserve_vector_HH(unsigned int huge *(huge *),
                           unsigned int);
  void free_matrix_H(unsigned int huge *(huge *(*)),
                         unsigned int);

  *mat=(unsigned int huge *(huge *)) farmalloc(f*sizeof(*(*mat)));
  if(!(*mat)) return 0;

  if(!c) {for(i=0;i<f;i++){ (*mat)[i] = NULL; } return 1;}

  for(i=0;i<f;i++){
    if(!reserve_vector_HH(&((*mat)[i]),c)){
      free_matrix_H(mat,i); return 0;
    }
  }
```

```
  return 1;
}


void free_matrix_H(unsigned int huge *(huge *(*mat)),
                       unsigned int f)
{ int i; void free_vector_HH(unsigned int huge *(huge *));
  for(i=0;i<f;i++) free_vector_HH(&((*mat)[i]));
  farfree((unsigned int far *) (*mat));
}


int reserve_vector_HH(unsigned int huge *(huge *v),
                         unsigned int n)
{ *v=(unsigned int huge *) farmalloc(n*sizeof(unsigned int));
  if(*v) return 1; else return 0;}


void free_vector_HH(unsigned int huge *(huge *v))
{if(*v) farfree((unsigned int far *) (*v));}

int reserve_vector_H(unsigned int huge *(*v),
                        unsigned int n)
{ *v=(unsigned int huge *) farmalloc(n*sizeof(unsigned int));
  if(*v) return 1; else return 0;}


void free_vector_H(unsigned int huge *(*v))
{if(*v) farfree((unsigned int far *) (*v));}


int reserve_pointers_sp(struct pair ***mat,
                          unsigned int f)
{
  unsigned int i;

  *mat = (struct pair **) malloc(f*sizeof(*(*mat)));
  if(!(*mat)) return 0;
  for(i=0;i<f;i++) (*mat)[i] = NULL;
  return 1;
}


int reserve_vector_sp(struct pair **v,unsigned int n)
```

```
{ *v = (struct pair *) malloc(n*sizeof(struct pair));
  if(*v) return 1; else return 0;}
```

A.2 Calculus of the path transition matrix

The aim of the function **fillpaths** is the calculus of the path transition matrix associated to a pattern $\mathcal{P}$. Recall that the algebraic representation of $\mathcal{P}$ described in Subsection 3.2.3 is stored using `**c`, `npunts` and `**orb`. In addition, some new external variables are assumed to be memory-allocated:

```
extern unsigned int **c,npoints,*asbp,ndc,
                    **paths2,*inv_paths,nnpaths;
extern int **paths; extern struct parell **orb,*bp;
```

The path transition matrix is stored at `**paths`, and the *reduced* path transition matrix obtained by erasing non-relevant paths (see Subsection 3.4) is stored at `**paths2`. The amount of relevant basic paths is stored at `nnpaths`. The amount of different basic paths contained in the discrete components from 1 to n, for $1 \leq n \leq$`ndc`, is stored at `asbp[n]` ("accumulated sum of basic paths"). Thus the total amount of basic paths is `asbp[ndc]`.

This function uses a little function, **nbp**, ("number of basic path"), which has three unsigned integers `com`, `p1`, `p2` as parameters, and returns the numbering which corresponds to the basic path defined by points `p1` and `p2` of the `com`-th discrete component.

The conventions and procedures are very similar to that of **Markov** function. In particular, note that each element of the matrix `**paths` has a sign. As in Subsection 3.4, this is equivalent to associate a sign to each arrow in the $\mathcal{P}$-path graph. This will allow us to associate a sign to each loop of the $\mathcal{P}$-path graph.

```
void fillpaths(void) {

 unsigned int i,j,k,r,s;

 for(i=0;i<=asbp[ndc];i++) paths[i][0]=paths2[i][0]=0;

  asbp[0]=0;
  for(i=1;i<=ndc;i++) {
   for(j=asbp[i-1]+1;j<=asbp[i];j++) {
    unsigned int patologic=0,comp,point;
    r=orb[i][bp[j].e1].e1;s=orb[i][bp[j].e2].e1;
```

```
/* We cannot call REDUCTION with exactly one component
   marked as unerasable. Thus now we detect whether the
   images of the two points of the j-th basic path belong
   to the same discrete component. */

/* case 1: There is 1 \leq a \leq ndc such that the images
   of the two points of the j-th basic path have binary
   representations {a,b} and {a,c} respectively. */
if(r==s) {
 paths[j][0]=1;
 paths[j][1]=nbp(r,orb[i][bp[j].e1].e2,orb[i][bp[j].e2].e2);
 continue;
}
/* case 2: case 1 does not hold:  */
/* case 2.1 */
for(k=1;k<=ndc;k++) {
 if((k==r)||(k==s)) continue;
 if((c[r][k]==orb[i][bp[j].e1].e2)&&
 (c[s][k]==orb[i][bp[j].e2].e2)) {
  patologic=1;
  break;
 }
}
if(patologic){
 paths[j][0]=1; paths[j][1]=nbp(k,c[k][r],c[k][s]);
 continue;
}

/* case 2.2: */
if(c[r][s]==orb[i][bp[j].e1].e2) {
 paths[j][0]=1; paths[j][1]=nbp(s,c[s][r],orb[i][bp[j].e2].e2);
 continue;
}
if(c[s][r]==orb[i][bp[j].e2].e2) {
 paths[j][0]=1; paths[j][1]=nbp(r,orb[i][bp[j].e1].e2,c[r][s]);
 continue;
}

/* The convex hull of the images of the two points of the
j-th basic path intersects more than one component. Thus we
can call REDUCTION. */

for(k=1;k<=ndc;k++) c[k][k]=1;
c[r][r]=c[s][s]=2;
```

```
   reduction();

   comp=r;
   point=orb[i][bp[j].e1].e2;
   /* is r irrelevant?: */
   for(k=1;k<=ndc;k++) {
    if((k==r)||(!c[k][k])) continue;
    if(c[r][k]==point) {
     c[r][r]=0;
     comp=k;
     point=c[k][r];
     break;
    }
   }
   do {
    for(k=1;k<=ndc;k++) if((k!=comp)&&(c[k][k])&&
    (c[comp][k])) break;
    paths[j][++paths[j][0]]=nbp(comp,point,c[comp][k]);
    c[comp][comp]=0;
    point=c[k][comp];
    comp=k;
   } while(c[comp][comp]!=2);
   if(point!=orb[i][bp[j].e2].e2) {
    paths[j][++paths[j][0]]=nbp(comp,point,orb[i][bp[j].e2].e2);
   }
  }
 }

/* Now we compute the reduced matrix **paths2. */
/* Deleting unnecessary paths: */

for(i=1;i<=asbp[ndc];i++) paths[0][i]=1;
do {
 k=0;
 for(i=1;i<=asbp[ndc];i++) {
  if(!paths[0][i]) continue;
  for(j=1;j<=paths[i][0];j++)
  if(paths[0][abs(paths[i][j])]) goto MORE;
  paths[0][i]=0; k=1; goto NEWI;
    /* i-th edge doesn't cover anything  */
  MORE:;
  for(j=1;j<=asbp[ndc];j++) {
   if(!paths[0][j]) continue;
   if(pos(i,paths[j])) goto NEWI;
```

```
   }
   paths[0][i]=0; k=1;   /* i-th edge non-covered by anything */
   NEWI:;
  }
 } while(k);

 /* change of names (direct and inverse), and nnpaths =
    amount of significant paths: */

 nnpaths=0;
 for(i=1;i<=asbp[ndc];i++) {
  if(!paths[0][i]) continue;
  paths[0][i]=(++nnpaths); inv_paths[nnpaths]=i;
 }


 /* translate into **paths2: */

 for(i=1;i<=asbp[ndc];i++) {
  if(!paths[0][i]) continue;
  for(j=1;j<=paths[i][0];j++) { int l;
   if(!(l=paths[0][abs(paths[i][j])])) continue;
   paths2[paths[0][i]][++paths2[paths[0][i]][0]]=l;
  }
 }

 /* sort each row: */


 for(i=1;i<=nnpaths;i++) if(paths2[i][0]>1)
  Sort1(paths2[i][0],paths2[i]);
 return;
}


int nbp(unsigned int com,unsigned int p1,unsigned int p2) {
 unsigned int a,b,x,y,z;
 if((p1==p2)||(!p1)||(!p2)) error(5,NULL);
 if((p1>c[0][com])||(p2>c[0][com])) error(5,NULL);
 x=p1;y=p2;if(p2<p1) {x=p2;y=p1;};
 z=asbp[com-1]+c[0][com]*(x-1)-x*(x-1)/2+y-x;

 if (p1<p2) return z; else return -z;
}
```

A.3 Sorting functions

We have used two functions for sorting vectors of integers. The algorithms have been extracted from [41]. The function **Sort1** sorts a vector of positive integers `*ra` between positions `ra[1]` and `ra[n]` in increasing order.

```
void Sort1(unsigned n, unsigned int *ra)
{
   int l,j,ir,i;
   unsigned int rra;

   l=(n >> 1)+1;
   ir=n;
   for (;;) {
      if (l > 1)
         rra=ra[--l];
      else {
         rra=ra[ir];
         ra[ir]=ra[1];
         if (--ir == 1) {
            ra[1]=rra;
            return;
         }
      }
      i=l;
      j=l << 1;
      while (j <= ir) {
         if (j < ir && ra[j] < ra[j+1]) ++j;
         if (rra < ra[j]) {
            ra[i]=ra[j];
            j += (i=j);
         }
         else j=ir+1;
      }
      ra[i]=rra;
   }
}
```

The function **Sort2** sorts a vector of positive integers `*ra` between `ra[1]` and `ra[n]` according to the values of a vector `*rb` of positive integers. That is, positions 1 to n of of `*rb` are sorted in increasing order, carrying the changes to the vector `*ra`.

```
void Sort2(unsigned int n, unsigned int *ra, unsigned int *rb)
```

```
{
   int l,j,ir,i;
   unsigned int rrb,rra;

   l=(n >> 1)+1;
   ir=n;
   for (;;) {
      if (l > 1) {
         rra=ra[--l];
         rrb=rb[l];
      } else {
         rra=ra[ir];
         rrb=rb[ir];
         ra[ir]=ra[1];
         rb[ir]=rb[1];
         if (--ir == 1) {
            ra[1]=rra;
            rb[1]=rrb;
            return;
         }
      }
      i=l;
      j=l << 1;
      while (j <= ir)   {
         if (j < ir && ra[j] < ra[j+1]) ++j;
         if (rra < ra[j]) {
            ra[i]=ra[j];
            rb[i]=rb[j];
            j += (i=j);
         }
         else j=ir+1;
      }
      ra[i]=rra;
      rb[i]=rrb;
   }
}
```

A.4 Data input and output

Here we list the source code of some functions which either get data from the keyboard or read/write data from/to disk files:

```
int getnum(int min, int max)
```

```
{
 /* This function waits for an input from the keyboard. The
    input must be an integer inside the rang [min,max]. */

  int n; int x,y;
  x = wherex(); y = wherey();
  while (!scanf("%u",&n) || n < min || n > max){
    putch(7); gotoxy(x,y); clreol(); fflush(stdin); }
  return n;
}

char *err_pat[]={
  "opening the pattern write file",
  "of pattern writing",
  "the pattern file does not exist",
  "of pattern reading"
  };

void WritePattern(char *fitxer)
{
  int l,n;
  FILE *fp;
  if(!access(fitxer,0) &&
     (getanswer("YN","This file already exists. Continue? (Y/N)")
     != 'Y')) exit(0);
  if((fp=fopen(fitxer,"wb")) == NULL) error(1,err_pat);
  n=ndc+1;
  if(!fwrite(&ndc,sizeof(unsigned int),1,fp) ||
     fwrite(c[0],sizeof(unsigned int),n,fp) != n)
     error(2,err_pat);
  for(l=1;l<=ndc;l++){
    if(fwrite(c[l],sizeof(unsigned int),n,fp) != n ||
       fwrite(&(orb[l][1]),sizeof(struct pair),
       c[0][l],fp) != c[0][l]) error(2,err_pat);
  }
  fclose(fp);
}

void ReadPattern(char *fitxer)
{
  int l,n;
  FILE *fp;
  if((fp=fopen(fitxer,"rb")) == NULL) error(3,err_pat);
  if(!fread(&ndc,sizeof(unsigned int),1,fp)) error(4,err_pat);
```

```
  n=ndc+1;
  if(!reserve_matrix(&c,n,n)
  || !reserve_pointers_sp(&orb,n)) error(3,NULL);
  if(fread(c[0],sizeof(unsigned int),n,fp) != n)
   error(4,err_pat);
  for(l=1;l<=ndc;l++){
    if(fread(c[l],sizeof(unsigned int),n,fp) != n)
     error(4,err_pat);
    if(!reserve_vector_sp(&(orb[l]),c[0][l]+1))
     error(3,NULL);
    if(fread(&(orb[l][1]),sizeof(struct pair),c[0][l],fp)
     != c[0][l]) error(4,err_pat);
  }
  fclose(fp);
}

#include "ctype.h"
char getanswer(char *answers, char *prompt)
/* Prints a prompt and gets an allowed answer
   pressed.  */
{ char a; int x,y;
 printf("\n%s: ",prompt); x=wherex(); y=wherey();
 do { gotoxy(x,y); clreol(); a = toupper(getchar());
 } while (strchr(answers,a) == NULL);
 return(a);
} /* getanswer */
```

A.5 Other functions

The following function **error**, of type `void`, is declared in the head file:

```
void error(int,char**);
```

A vector of strings containing 4 error messages is declared before any function declaration. These messages can occur during the execution of any function. If one sends `NULL` as the value of the parameter `*buff[]`, then **error** prints out the general error message number `num` ($1 \leq$ `num` ≤ 4). Otherwise, **error** assumes that `*buff[]` is a vector of local error messages and then `num`-th message is printed out.

```
char *err_gen[]={
  "General of memory",
  "Non allocated vector",
```

```
  "Non allocated matrix",
  "Impossible structural error"
  };

void error(int num, char *buff[])
{ if(buff) printf("\nERROR: %s.\n\n",buff[num-1]);
  else printf("\ngeneral ERROR: %s.\n\n",err_gen[num-1]);
  exit(1);}
```

The following functions are self-commented:

```
unsigned int belongs(unsigned int m,unsigned int *n)
{
 /* If m belongs to the positive integer vector n,
    returns its lowest occurrence position. Otherwise
    returns 0. */

 unsigned int i;
 for(i=1;i<=n[0];i++) { /* n[0] = length of n. */
  if(m==n[i]) return i; /* If n[0]=0 then 'for' does not act */
 }                      /* and 0 is returned. */
 return 0;
}


int pos(int m,int *n)
{
 /* If the positive integer m coincides with the absolute value
    of some component of the vector n then returns the lowest
    occurrence position. Otherwise returns 0. */

 int i;
 for(i=1;i<=n[0];i++) {     /* n[0] = length of n. */
  if(m==abs(n[i])) return i;/* If n[0]=0 then 'for' does not act */
 }                          /* and 0 is returned. */
 return 0;
}


int sgn(int n)
{
 /* Returns the sign of n. */

 if(n>0) return 1;
```

```
 if(n<0) return -1;
 return 0;
}
```

Bibliography

[1] Adler, R., Konheim, A., McAndrew, M. [1965] "Topological entropy", *Trans. Am. Math. Soc.* **114**, 309–319.

[2] Alsedà, Ll., Gautero, F., Guaschi, J., Los, J., Mañosas, F., Mumbrú, P. [2002] "Patterns and minimal dynamics for graph maps", Prepublicacions UAB.

[3] Alsedà, Ll., Guaschi, J., Los, J., Mañosas, F., Mumbrú, P. [1997] "Canonical representatives for patterns of tree maps", *Topology* **36**, 1123–1153.

[4] Alsedà, Ll., Juher, D., Mumbrú, P. [2001] "A note on the periodic orbits and topological entropy of graph maps", *Proc. Amer. Math. Soc.* **129**, no. 10, 2941–2946.

[5] Alsedà, Ll., Juher, D., Mumbrú, P. [2003] "Sets of periods of piecewise monotone tree maps", to appear in *Int. J. of Bif. and Chaos*

[6] Alsedà, Ll., Juher, D., Mumbrú, P. [2003] "On the minimal models for graph maps", to appear in *Int. J. of Bif. and Chaos*

[7] Alsedà, Ll., Llibre, J., Misiurewicz, M. [1989] "Periodic orbits of maps of Y", *Trans. Amer. Math. Soc.* **313**, 475–538.

[8] Alsedà, Ll., Llibre, J., Misiurewicz, M. [2000] *Combinatorial dynamics and entropy in dimension one*, Advanced Series in Nonlinear Dynamics **5**, World Scientific, second edition.

[9] Alsedà, Ll., Llibre, J., Misiurewicz, M., Simó, C. [1985] *Twist periodic orbits and topological entropy for continuous maps of the circle of degree one which have a fixed point*, Ergodic Theory Dynam. Systems **5**, 501–517.

[10] Alsedà, Ll., Llibre, J., Serra, R. [1984] "Minimal periodic orbits for continuous maps of the interval", *Trans. Am. Math. Soc.* **286**, 595–627.

[11] Alsedà, Ll., Mañosas, F., Mumbrú, P. [2000] "Minimizing topological entropy for continuous maps on graphs", *Ergod. Th. & Dynam. Sys.* **20**, no. 6, 1559–1576.

[12] Alsedà, Ll., Ye, X. [1993] "Division for star maps with the branching point fixed", *Acta Math. Univ. Comenianae* **LXII**, 237–248.

[13] Alsedà, Ll., Ye, X. [1994] "Minimal sets of maps of Y", *J. Math. Anal. Appl.* **187**, 324–338.

[14] Alsedà, Ll., Ye, X. [1995] "No division and the set of periods for tree maps", *Ergod. Th. & Dynam. Sys.* **15**, 221–237.

[15] Baldwin, S. [1987] "Generalizations of a theorem of Sharkovskii on orbits of continuous real-valued functions", *Discrete Math.* **67**, 111–127.

[16] Baldwin, S. [1991] "An extension of Sharkovskii's Theorem to the n-od", *Ergod. Th. & Dynam. Sys.* **11**, 249–271.

[17] Baldwin, S., Slaminka, E. [1997] "Calculating topological entropy", *J. Stat. Phys.* **89**, 1017–1033.

[18] Block, L. [1979] "Simple periodic orbits of mappings of the interval", *Trans. Amer. Math. Soc.*, **254**, 391–398.

[19] Block, L. [1981] "Periods of periodic points of maps of the circle which have a fixed point", *Proc. Amer. Math. Soc.* **82**, 481–486.

[20] Block, L., Coven, E. [1989] "Approximating entropy of maps of the interval", *Proceedings of the semester on Ergodic Theory and Dynamical Systems*, 237–241, Banach Center Pub. **23**, PWN, Warsaw.

[21] Block, L., Guckenheimer, J., Misiurewicz, M., Young, L.S. [1980] "Periodic points and topological entropy of one-dimensional maps", *Global theory of dynamical systems*, pp. 18–34, SLNM **819**, Springer, Berlin.

[22] Blokh, A.M. [1991] "On some properties of graph maps: spectral decomposition, Misiurewicz conjecture and abstract sets of periods", preprint, *Max Planck Institut für Mathematik*, Bonn.

[23] Blokh, A.M. [1992] "Periods implying almost all periods for tree maps", *Nonlinearity* **5**, 1375–1382.

[24] Blokh, A.M. [1994] "Trees with snowflakes and zero entropy maps", *Topology* **33**, 379–396.

[25] Bowen, R. [1973] "Entropy for group endomorphisms and homogeneous spaces", *Trans. Am. Math. Soc.* **153**, 401–414; erratum: *Trans. Am. Math. Soc.* **181**, 509–510.

[26] Coppel, W.A. [1983] "Sharkovskii-minimal orbits", *Math. Proc. Cambr. Philos. Soc.* **93**, 397–408.

[27] Denker, M., Grillenberger, C., Sigmund, K. [1976] *Ergodic theory on compact spaces*, SLNM **527**, Springer, Berlin.

[28] Efremova, L.S. [1978] "Periodic orbits and a degree of a continuous map of a circle" (in Russian), *Diff. and Integr. Equations* (Gor'kii) **2**, 109–115.

[29] Góra, P., Boyarsky, A. [1991] "Computing the topological entropy of general one-dimensional maps", *Trans. Am. Math. Soc.* **323**, 39–49.

[30] Ho, C.-W. [1984] "On Block's condition for simple periodic orbits of functions on an interval", *Trans. Am. Math. Soc.* **281**, 827–832.

[31] Imrich, W., Kalinowski, R. [1985 a] "Periodic points of small periods of continuous mappings of trees", *Ann. Discrete Math.* **27**, 443–446.

[32] Imrich, W., Kalinowski, R. [1985 b] "Periodic points of continuous mappings of trees", *Ann. Discrete Math.* **27**, 447–460.

[33] Jiang, B. [1983] *Lectures on Nielsen fixed point theory*, American Mathematical Society, Providence, R.I.

[34] Juher, D. [1998] *Estudi del conjunt de períodes en models canònics d'aplicacions d'arbres*, Treball de Recerca, Universitat Autònoma de Barcelona.

[35] Li, T.-Y., Misiurewicz, M., Pianigiani, G., Yorke, J.A [1982] "No division implies chaos", *Trans. Amer. Math. Soc.* **273** 191–199.

[36] Llibre, J., Misiurewicz, M. [1993] "Horseshoes, entropy and periods for graph maps", *Topology* **32**, 649–664.

[37] Misiurewicz, M [1982] "Periodic points of maps of degree one of a circle", *Ergod. Th. & Dynam. Sys.* **2**, 221–227.

[38] Misiurewicz, M., Nitecki, Z. [1991] "Combinatorial patterns for maps of the interval", *Mem. Amer. Math. Soc.* **94**, no. 456, vi+112 pp.

[39] Misiurewicz, M., Szlenk, W. [1980] "Entropy of piecewise monotone mappings", *Studia Math.* **67**, 45–63.

[40] Newhouse, S., Pignataro, T. [1993] "On the estimation of topological entropy", *J. Stat. Phys.* **72**, 1331–1351.

[41] Press, W.H. et al. *Numerical Recipes in C: the art of scientific computing*, Cambridge University Press, 1988

[42] Sharkovskii, A.N. [1964] "Co–existence of the cycles of a continuous mapping of the line into itself" (in russian), *Ukrain. Math. Zh.* **16 (1)**, 61–71. English translation in Proceedings of the Conference "Thirty Years after Sharkovskii's Theorem: New Perspectives" (Murcia, 1994), Internat. J. Bifur. Chaos Appl. Sci. Engrg. **5** (1995), 1263–1273.

[43] Takahashi, Y. [1980] "A formula for the topological entropy of one-dimensional dynamics", *Sci. Papers College Gen. Ed. Univ. Tokyo* **30**, 11–22.